AfIA

Association française
pour l'Intelligence Artificielle

TALN-RECITAL

Conférence sur le Traitement Automatique des Langues Naturelles

PFIA 2019

Table des matières

Emmanuel Morin, Sophie Rosset et Pierre Zweigenbaum (TALN)
Anne-Laure Ligozat et Sahar Ghannay (RECITAL).
Éditorial 7

.
Comités 8

Volume I : Articles longs

Syrielle Montariol et Alexandre Allauzen.
Apprentissage de plongements de mots dynamiques avec régularisation de la dérive 13

Victor Connes et Nicolas Dugué.
Apprentissage de plongements lexicaux par une approche réseaux complexes 27

Ludovic Tanguy, Pauline Brunet et Olivier Ferret.
Comparaison qualitative et extrinsèque d'analyseurs syntaxiques du français : confrontation de modèles distributionnels sur un corpus spécialisé 39

Loïc Vial, Benjamin Lecouteux et Didier Schwab.
Compression de vocabulaire de sens grâce aux relations sémantiques pour la désambiguïsation lexicale 55

Natalia Grabar, Cyril Grouin, Thierry Hamon et Vincent Claveau.
Corpus annoté de cas cliniques en français 71

Antoine Caubrière, Natalia Tomashenko, Yannick Estève, Antoine Laurent et Emmanuel Morin.
Curriculum d'apprentissage : reconnaissance d'entités nommées pour l'extraction de concepts sémantiques 85

Anissa Hamza et Delphine Bernhard.
Détection des ellipses dans des corpus de sous-titres en anglais 99

Tim Van de Cruys.
La génération automatique de poésie en français 113

Marco Dinarelli et Loïc Grobol.
Modèles neuronaux hybrides pour la modélisation de séquences : le meilleur de trois mondes 127

Amalia Todirascu, Marion Cargill et Thomas Francois.
PolylexFLE : une base de données d'expressions polylexicales pour le FLE 143

Volume II : Articles courts

Kate Thompson, Nicholas Asher, Philippe Muller et Jeremy Auguste.
Analyse faiblement supervisée de conversation en actes de dialogue 159

Salima Mdhaffar, Yannick Estève, Nicolas Hernandez, Antoine Laurent et Solen Quiniou.
Apport de l'adaptation automatique des modèles de langage pour la reconnaissance de la parole : évaluation qualitative extrinsèque dans un contexte de traitement de cours magistraux 167

Sonia Badene, Kate Thompson, Jean-Pierre Lorré et Nicholas Asher.
Apprentissage faiblement supervisé de la structure discursive 175

Frédéric Béchet, Cindy Aloui, Delphine Charlet, Géraldine Damnati, Johannes Heinecke, Alexis Nasr et Frédéric Herlédan.
CALOR-QUEST : un corpus d'entraînement et d'évaluation pour la compréhension automatique de textes 185

Iris Eshkol-Taravella, Mariame Maarouf, Marie Skrovec et Flora Badin.
Chunker différents types de discours oraux : défis pour l'apprentissage automatique 195

Yuming Zhai, Gabriel Illouz et Anne Vilnat.

Classification automatique des procédés de traduction 205

Guillaume Wisniewski.
Combien d'exemples de tests sont-ils nécessaires à une évaluation fiable ? Quelques observations sur l'évaluation de l'analyse morpho-syntaxique du français. 215

Tsanta Randriatsitohaina et Thierry Hamon.
De l'extraction des interactions médicament-médicament vers les interactions aliment-médicament à partir de textes biomédicaux : Adaptation de domaine 223

Fiammetta Namer, Lucie Barque, Olivier Bonami, Pauline Haas, Nabil Hathout et Delphine Tribout.
Demonette2 - Une base de données dérivationnelles du français à grande échelle : premiers résultats 233

Elise Bigeard et Natalia Grabar.
Détecter la non-adhérence médicamenteuse dans les forums de discussion avec les méthodes de recherche d'information 245

Rémi Cardon et Natalia Grabar.
Détection automatique de phrases parallèles dans un corpus biomédical comparable technique/simplifié 255

Benoît Sagot.
Développement d'un lexique morphologique et syntaxique de l'ancien français 265

Adrien Bardet, Fethi Bougares et Loïc Barrault.
Étude de l'apprentissage par transfert de systèmes de traduction automatique neuronaux 275

Antoine Perquin, Gwénolé Lecorvé, Damien Lolive et Laurent Amsaleg.
Évaluation objective de plongements pour la synthèse de parole guidée par réseaux de neurones 285

Sara Meftah, Nasredine Semmar, Youssef Tamaazousti, Hassane Essafi et Fatiha Sadat.
Exploration de l'apprentissage par transfert pour l'analyse de textes des réseaux sociaux 293

Syrielle Montariol, Aina Garí Soler et Alexandre Allauzen.
Exploring sentence informativeness 303

Fréjus A. A. Laleye, Antonia Blanié, Antoine Brouquet, Dan Benhamou et Gaël de Chalendar.
Hybridation d'un agent conversationnel avec des plongements lexicaux pour la formation au diagnostic médical 313

Nadia Bebeshina-Clairet et Mathieu Lafourcade.
Inférence des relations sémantiques dans un réseau lexico-sémantique multilingue 323

Jean-Yves Antoine, Marion Crochetet, Céline Arbizu, Emmanuelle Lopez, Samuel Pouplin, Amélie Besnier et Mathieu Thebaud.
Ma copie adore le vélo : analyse des besoins réels en correction orthographique sur un corpus de dictées d'enfants 333

Olga Seminck, Vincent Segonne et Pascal Amsili.
Modèles de langue appliqués aux schémas Winograd français 343

Patricia Chiril, Farah Benamara, Véronique Moriceau, Marlène Coulomb-Gully et Abhishek Kumar.
Multilingual and Multitarget Hate Speech Detection in Tweets 351

Iris Eshkol-Taravella et Hyun Jung Kang.
Observation de l'expérience client dans les restaurants 361

Laurent Kevers, Florian Guéniot, A. Ghjacumina Tognotti et Stella Retali-Medori.
Outiller une langue peu dotée grâce au TALN : l'exemple du corse et de la BDLC 371

Amira Barhoumi, Nathalie Camelin, Chafik Aloulou, Yannick Estève et Lamia Hadrich Belguith.
Plongements lexicaux spécifiques à la langue arabe : application à l'analyse d'opinions 381

Saoussen Mathlouthi Bouzid et Chiraz Ben Othmane Zribi.
Q-learning pour la résolution des anaphores pronominales en langue arabe 391

Tom Bourgeade et Philippe Muller.
Représentation sémantique distributionnelle et alignement de conversations par chat 399

Quentin Gliosca et Pascal Amsili.
Résolution des coréférences neuronale : une approche basée sur les têtes 409

Amir Hazem, Béatrice Daille, Dominique Stutzmann, Jacob Currie et Christine Jacquin.
Réutilisation de textes dans les manuscrits anciens 417

Aleksandra Miletić, Delphine Bernhard, Myriam Bras, Anne-Laure Ligozat et Marianne Vergez-Couret.
Transformation d'annotations en parties du discours et lemmes vers le format Universal Dependencies : étude de cas pour l'alsacien et l'occitan 427

Yoann Dupont.
Un corpus libre, évolutif et versionné en entités nommées du français 437

Filipo Studzinski Perotto, Fadila Taleb, Eric Trupin, Youssouf Saidali, Maryvonne Holzem, Jacques Labiche et Laurent Vercouter.
Une approche hybride pour la segmentation automatique de documents juridiques 447

Volume III : RECITAL

Mathilde Regnault.
Adaptation d'une métagrammaire du français contemporain au français médiéval 459

Mérième Bouhandi.
Apport des termes complexes pour enrichir l'analyse distributionnelle en domaine spécialisé 473

Jessica López Espejel.
Automatic summarization of medical conversations, a review 487

Bruno Oberle.
Détection automatique de chaînes de coréférence pour le français écrit : règles et ressources adaptées au repérage de phénomènes linguistiques spécifiques 499

Ygor Gallina.
Etat de l'art des méthodes d'apprentissage profond pour l'extraction automatique de termes-clés 513

Emmanuelle Kelodjoue.
Extraction d'opinions pour l'analyse multicritère à partir de corpus oraux transcrits : État de l'art 525

Léon-Paul Schaub et Cyndel Vaudapiviz.
Les systèmes de dialogue orientés-but : état de l'art et perspectives d'amélioration 541

Mathilde Veron.
Lifelong learning et systèmes de dialogue : définition et perspectives 563

Manon Scholivet.
Méthodes de représentation de la langue pour l'analyse syntaxique multilingue 577

Dusica Terzic.
Parsing des textes journalistiques en serbe à l'aide du logiciel Talismane 591

Sandra Bellato.
Vers la traduction automatique d'adverbiaux temporels du français en langue des signes française 605

Volume IV : Démonstrations

Didier Schwab, Pauline Trial, Céline Vaschalde, Loïc Vial et Benjamin Lecouteux.
Apporter des connaissances sémantiques à un jeu de pictogrammes destiné à des personnes en situation de handicap : un ensemble de liens entre WordNet et Arasaac, Arasaac-WN 619

Guillaume Dubuisson Duplessis, Sofiane Kerroua, Ludivine Kuznik et Anne-Laure Guénet.
Cameli @ : analyses automatiques d'e-mails pour améliorer la relation client 623

Marine Schmitt, Élise Moreau, Mathieu Constant et Agata Savary.
Démonstrateur en-ligne du projet ANR PARSEME-FR sur les expressions polylexicales 627

Olivier Hamon, Kévin Espasa et Sara Quispe.
SylNews, un agréfilter multilingue 631

Ioan Calapodescu, Caroline Brun, Vassilina Nikoulina et Salah Aït-Mokhtar.
"Sentiment Aware Map" : exploration cartographique de points d'intérêt via l'analyse de sentiments au niveau des aspects 635

Alexandre Arnold, Gérard Dupont, Catherine Kobus, François Lancelot et Pooja Narayan.
Interprétation et visualisation contextuelle de NOTAMs (messages aux navigants aériens) . . . 639

Éditorial

La 26e édition de la conférence TALN et la 21e édition de la session jeunes chercheuses et chercheurs RECITAL se déroulent cette année à Toulouse au sein de la Plateforme française d'intelligence artificielle (PFIA). TALN a une longue tradition de tenue conjointe avec des conférences de domaines proches. Cette pratique a été initiée avec les Journées d'étude sur la parole (JEP) en 2002 à Nancy puis depuis 2008 tous les quatre ans (2008 : Avignon, 2012 : Grenoble, 2016 : Paris). Elle s'est diversifiée avec la Conférence de recherche d'information et applications (CORIA) en 2018 à Rennes. Elle innove cette année avec un hébergement à Toulouse au sein de PFIA. Ces événements sont l'occasion de rencontres enrichissantes pour tous. Cette année, ce ne sont pas moins de huit conférences, sans compter les ateliers associés, aux sessions desquelles les participants à TALN-RECITAL pourront se mêler : APIA (5e Conférence sur les Applications Pratiques de l'Intelligence Artificielle), CAp (21e Conférence sur l'Apprentissage Automatique), IC (30es Journées Francophones Ingénierie des Connaissances), JFPDA (14es Journées Planification, Décision et Apprentissage), JFSMA (27es Journées Francophones sur les Systèmes Multi-Agents), JIAF (13es Journées d'Intelligence Artificielle Fondamentale), RJCIA (17e Rencontre des Jeunes Chercheurs en Intelligence Artificielle), ainsi que CNIA (22e Conférence Nationale en Intelligence Artificielle), qui regroupe les thématiques de l'intelligence artificielle non couvertes par les conférences précédentes.

Les conférences invitées plénières, les sessions de présentations affichées et de démonstrations, les déjeuners et pauses café, les dîners de la conférence sont autant de moments programmés pour que se retrouvent les participants de toutes les conférences. Nous tenons à saluer la qualité de la planification et du suivi du comité scientifique de la plateforme ainsi que le grand travail du comité d'organisation, le tout visant à assurer que l'ensemble des conférences se tienne dans les meilleures conditions et au meilleur coût.

Pour la deuxième année consécutive, les modalités de soumission à TALN se faisaient avec un appel unique et un seul format de soumission en article court pouvant être étendu en article long sur proposition du comité de programme (et demande préalable des auteurs). Nous avons ainsi reçu soixante cinq articles courts et le comité de programme a proposé à dix articles le passage en format long (15 %) et a retenu trente et un articles en format court (48 %). Chaque article a été relu par trois membres du comité de lecture en s'appuyant le cas échant sur des relecteurs additionnels. Le comité de programme s'est appuyé sur ces relectures pour sélectionner lors d'une réunion pleinière les articles composant le programme. C'est un fonctionnement auquel nous sommes profondément attachés pour assurer une diversité dans les thématiques abordées. L'ensemble des évaluations ont été réalisées en double aveugle. Nous remercions les membres des comités de programme et de lecture (à parité femme – homme) pour leur contribution indispensable à ce processus. Le programme de la conférence est complété par quatre démonstrations sélectionnées par le comité de programme. Les titres des sessions donnent une idée des thématiques abordées par la conférence. Ils comprennent des paliers et tâches habituels du TAL (Morphologie et Syntaxe, Syntaxe, Résolution d'anaphores, Multilinguisme), reflètent la place prise par l'apprentissage (Apprentissage par transfert et modèles de langue, Plongements de mots), l'importance fondamentale que continuent à jouer les corpus et bases de données lexicales (Ressources), et l'intérêt du TAL pour des domaines particuliers (Langues spécialisées, Traitement de la langue biomédicale). Comme chaque année, l'ATALA a décerné un prix de thèse dont la récipiendaire présentera son travail en session plénière. La conférence a invité la présentation d'instruments récents du CNRS par leurs coordinatrices : d'une part le pré-GDR TAL (INS2I / informatique), qui adopte une vision inclusive du traitement de la langue (écrite, orale, signée), couvrant les communautés du traitement automatique des langues, du traitement automatique du langage parlé et de la recherche d'information ; d'autre part le GDR LIFT (INSHS / sciences du langage) sur la linguistique informatique, formelle et de terrain.

Cette année, dix-sept articles ont été soumis à RECITAL. Après avoir été chacun évalué par deux membres du comité de programme, quatre articles ont été retenus pour une présentation orale (soit un taux de sélection pour présentation orale de 24 %), et sept autres ont été retenus pour une présentation sous forme de poster (taux de sélection global de 65 %). Nous avons ainsi pu donner l'opportunité à douze jeunes chercheuses et chercheurs, en grande majorité en début de thèse, de présenter leurs travaux à la communauté. Nous remercions le comité de programme (également à parité femme – homme) pour leur minutieux travail de relecture.

Nous souhaitons pour finir au public de ces conférences une semaine riche en découvertes scientifiques et en rencontres de nouveaux collègues, dans une ambiance assurément chaude pour toute la semaine.

Emmanuel Morin, Sophie Rosset et Pierre Zweigenbaum (TALN)
Anne-Laure Ligozat et Sahar Ghannay (RECITAL)

Comités

Présidents de TALN

— Emmanuel Morin (LS2N, Université de Nantes)
— Sophie Rosset (LIMSI, CNRS, Université Paris-Saclay)
— Pierre Zweigenbaum (LIMSI, CNRS, Université Paris-Saclay)

Membres du CP de TALN

— Delphine Bernard (LiLPa, Université de Strasbourg)
— Chloé Braud (LORIA, CNRS)
— Nathalie Camelin (LIUM, Le Mans Université)
— Peggy Cellier (IRISA, INSA Rennes)
— Benoît Crabbé (LLF, Université Paris Diderot)
— Iris Eshkol-Taravella (MoDyCo, Université Paris Nanterre)
— Cécile Fabre (CLLE-ERSS, Université Toulouse - Jean Jaurès)
— Núria Gala (LPL, Aix Marseille Université)
— Thierry Hamon (LIMSI, Université Paris Nord)
— Philippe Langlais (RALI/DIRO, Université de Montréal)
— Gwénolé Lecorvé (IRISA, Université de Rennes 1)
— Aurélie Névéol (LIMSI, CNRS, Université Paris-Saclay)
— Damien Nouvel (ERTIM, INaLCO)
— Didier Schwab (LIG, Université Grenoble Alpes)
— Xavier Tannier (LIMICS, Université Pierre et Marie Curie)

Comité de lecture de TALN

— Gilles Adda (LIMSI, CNRS, Université Paris-Saclay)
— Salah Ait-Mokhtar (Naver Labs Europe)
— Alexandre Allauzen (LIMSI, CNRS, Université Paris-Saclay)
— Maxime Amblard (LORIA, Université de Lorraine)
— Jean-Yves Antoine (LIFAT, Université de Tours)
— Loïc Barrault (LIUM, Le Mans Université)
— Denis Béchet (LS2N, Université de Nantes)
— Frederic Béchet (LIS, Aix-Marseille Université)
— Patrice Bellot (LIS, Aix-Marseille Université)
— Asma Ben Abacha (Lister Hill Center, National Library of Medicine)
— Laurent Besacier (LIG, Université Grenoble Alpes)
— Yves Bestgen (ILC, Université catholique de Louvain)
— Philippe Blache (LPL, CNRS, Aix-Marseille Université)
— Fethi Bougares (LIUM, Le Mans Université)
— Thierry Charnois (LIPN, Université Paris 13)
— Vincent Claveau (IRISA, CNRS)
— Chloé Clavel (LTCI, Télécom ParisTech)
— Kevin Bretonnel Cohen (University of Colorado School of Medicine)
— Béatrice Daille (LS2N, Université de Nantes)
— Géraldine Damnati (Orange Labs)
— Gaël Dias (GREYC, Normandie Université)
— Marco Dinarelli (LIG, CNRS)
— Patrick Drouin (OLST, Université de Montréal)
— Dominique Estival (MARCS, Western Sydney University)
— Yannick Estève (LIUM, Le Mans Université)
— Olivier Ferret (CEA LIST)
— Karën Fort (STIH, Sorbonne Université)
— Thomas Francois (CENTAL, Université catholique de Louvain)
— Éric Gaussier (LIG, Université Grenoble Alpes)
— Jérôme Goulian (LIG, Université Grenoble Alpes)

— Natalia Grabar (STL, CNRS)
— Cyril Grouin (LIMSI, CNRS, Université Paris-Saclay)
— Olivier Hamon (Syllabs)
— Nabil Hathout (CLLE-ERSS, CNRS)
— Amir Hazem (LS2N, Université de Nantes)
— Nicolas Hernandez (LS2N, Université de Nantes)
— Stéphane Huet (LIA, Université d'Avignon et des Pays de Vaucluse)
— Christine Jacquin (LS2N, Université de Nantes)
— Sylvain Kahane (Modyco, Université Paris Nanterre)
— Olivier Kraif (LIDILEM, Université Grenoble Alpes)
— Mathieu Lafourcade (LIRMM, Université de Montpellier)
— David Langlois (LORIA, Université de Lorraine)
— Eric Laporte (LIGM, Université Paris-Est Marne-la-Vallée)
— Thomas Lavergne (LIMSI, Université Paris Sud, Université Paris-Saclay)
— Joseph Le Roux (LIPN, Université Paris 13)
— Benjamin Lecouteux (LIG, Université Grenoble Alpes)
— Yves Lepage (Waseda University)
— Denis Maurel (LIFAT, Université de Tours)
— Richard Moot (LIRMM, CNRS)
— Véronique Moriceau (IRIT, Université Paul Sabatier)
— Philippe Muller (IRIT, Université Paul Sabatier)
— Alexis Nasr (LIS, Aix Marseille Université)
— Adeline Nazarenko (LIPN, Université Paris 13)
— Luka Nerima (Université de Genève)
— Jian-Yun Nie (RALI/DIRO, Université de Montréal)
— Yannick Parmentier (LORIA, Université de Lorraine)
— Sebastian Peña Saldarriaga (Dictanova)
— Thierry Poibeau (Lattice, CNRS)
— Alain Polguère (ATILF, Université de Lorraine)
— Jean-Philippe Prost (LIRMM, Université de Montpellier)
— Solen Quiniou (LS2N, Université de Nantes)
— Christian Raymond (IRISA, INSA Rennes)
— Christian Retoré (LIRMM, Université de Montpellier)
— Djamé Seddah (ALMAnaCH, Paris Sorbonne Université)
— Gilles Serasset (LIG, Université Grenoble Alpes)
— Michel Simard (NRC, Canada)
— Kamel Smaili (LORIA, Université de Lorraine)
— Pascale Sébillot (IRISA, INSA Rennes)
— Ludovic Tanguy (CLLE-ERSS, Université Toulouse - Jean Jaurès)
— Juan-Manuel Torres-Moreno (LIA, Université d'Avignon et des Pays de Vaucluse)
— Guillaume Wisniewski (LIMSI, Université Paris-Sud, Université Paris-Saclay)
— François Yvon (LIMSI, CNRS, Université Paris-Saclay)

Relecteurs additionnels de TALN

— Jingshu Liu (Dictanova)
— Emile Chapuis (LTCI, Télécom ParisTech)
— Caroline Langlet (LTCI, Paris Sorbonne Université)
— Joseph Lark (Dictanova)
— Alexandre Garcia (LTCI, Télécom ParisTech)

Présidentes de RECITAL

— Anne-Laure Ligozat (LIMSI, CNRS, Université Paris-Saclay)
— Sahar Ghannay (LIMSI, CNRS, Université Paris-Saclay)

Membres du CP de RECITAL

— Jean-Yves Antoine (LIFAT, Université de Tours)

— Ismail Badache (ESPE / LIS, Aix-Marseille Université)
— Amira Barhoumi (LIUM, Université du Maine - MIRACL Sfax)
— Rachel Bawden (University of Edinburgh)
— Aurélien Bossard (LIASD, Université Paris 8)
— Chloé Braud (LORIA, CNRS)
— Nathalie Camelin (LIUM, Université du Maine)
— Rémi Cardon (STL, Lille)
— Peggy Cellier (IRISA, INSA Rennes)
— Antoine Doucet (L3i, Université de la Rochelle)
— Maha Elbayad, LIG/ Inria
— Arnaud Ferré (LIMSI-CNRS/MaIAGE-INRA, Université Paris-Saclay)
— Amel Fraisse (Gériico, Lille)
— Thomas François (CENTAL, Université catholique de Louvain)
— Nicolas Hernandez (LS2N, Université de Nantes)
— Yann Mathet (Greyc, Université de Caen)
— Alice Millour (STIH, Université Paris-Sorbonne)
— Anne-Lyse Minard (LLL, Orléans)
— Jose Moreno (IRIT, UPS)
— Tsanta Randriatsitohaina (LIMSI, Université Paris-Sud, Université Paris-Saclay)
— Loïc Vial (LIG, Université Grenoble Alpes)

Volume III : RECITAL

Adaptation d'une métagrammaire du français contemporain au français médiéval

Mathilde Regnault[1, 2]
(1) Lattice
(2) Inria
mathilde.regnault@sorbonne-nouvelle.fr

RÉSUMÉ
Le français médiéval se caractérise par une importante variabilité langagière. Nous cherchons à étendre un corpus d'ancien français annoté en syntaxe de dépendance avec de nouveaux textes de cette période et de moyen français. Pour cela, nous voulons adapter des outils existants et non entraîner un *parser* avec des données annotées. Dans cet article, nous présentons un état de l'art pour ce projet et notre démarche : adapter FRMG (*French Metagrammar*) à des états de langue antérieurs.

ABSTRACT
Adapting an existing metagrammar for Contemporary French to Medieval French

Medieval French is characterized by strong language variation. Our purpose is to extend a corpus of Old French annotated with dependency syntax with new texts of this period and add texts of Middle French. In order to achieve this, we want to adapt existing tools instead of training a parser with annotated data. In this article, we present a state of the art for this project and our solution : adapting the French Metagrammar (FRMG) to former states of language.

MOTS-CLÉS : métagrammaire, analyse syntaxique automatique, français médiéval, données hétérogènes.

KEYWORDS: metagrammar, syntactical parsing, Medieval French, heterogeneous data.

1 Introduction

Dans le cadre du projet ANR Profiterole, nous développons des outils d'annotation syntaxique pour traiter le français médiéval, caractérisé par une importante variabilité langagière. Cela représente une difficulté pour le *parsing* (Guibon *et al.*, 2014, 2015). Pour gérer les différentes variations (diachroniques, dialectales...), nous avons choisi de ne pas nous servir d'un *parser* statistique, mais de développer une métagrammaire du français médiéval, que nous n'avons pas encore évaluée. Un système symbolique permet de mettre à profit les grammaires existantes (Hasenohr & Raynaud de Lage, 1993; Buridant, 2000; Marchello-Nizia, 1979). Une métagrammaire est une description modulaire et hiérarchisée des phénomènes syntaxiques, divisée en classes, qui nous permet de mettre en place une analyse fine de la langue et d'intégrer des contraintes activées ou non selon le contexte ou les métadonnées. Ce formalisme présente aussi une économie dans l'écriture de la grammaire d'une langue car ses classes permettent de factoriser les descriptions de phénomènes. Nous pensons qu'il est possible d'adapter une métagrammaire du français contemporain, FRMG (Villemonte de la Clergerie, 2005), à des états de langue antérieurs et hétérogènes. Nous pourrons ainsi étendre

le SRCMF (Prévost & Stein, 2013), un corpus d'ancien français (9e-13e s.) annoté en syntaxe de dépendance, pour en faire une ressource conséquente (env. un million de mots), incluant des textes en moyen français (14e-15e s.).

Dans cet article, nous présenterons d'abord les objectifs de nos travaux et les défis qu'ils soulèvent. Puis nous exposerons les problématiques liées à l'adaptation d'un outil du français contemporain au français médiéval, et, dans une dernière partie, nos solutions et travaux en cours.

2 Objectifs et défis de l'analyse syntaxique automatique du français médiéval

2.1 Objectifs du projet

Nous souhaitons multiplier par quatre la taille du SRCMF, un corpus arboré de 250 000 mots, en y ajoutant des textes annotés automatiquement. L'annotation manuelle est trop coûteuse pour envisager une nouvelle campagne de cette ampleur. Les nouveaux textes sont en partie choisis pour rendre la représentation de la langue médiévale plus complète. Six nouveaux dialectes feront ainsi leur apparition (pour un nouveau total de douze), comme le bourguignon, l'orléanais et le wallon. Les domaines seront mieux représentés, en particulier les textes historiques, dont on ne trouve qu'un exemplaire dans le SRCMF. Le changement le plus remarquable est l'ajout du moyen français à ce corpus, qui présente des différences notables avec l'ancien français.

2.2 Travaux antérieurs

Dans leurs premiers travaux sur le SRCMF, Guibon *et al.* (2014) ont fait des expériences d'étiquetage morpho-syntaxique et syntaxique avec des CRF et un *parser* statistique. Cependant, les résultats sont moins bons que pour une langue contemporaine en partie à cause de l'hétérogénéité des données, due notamment aux divers dialectes et aux différentes époques des textes. Ces techniques reposent essentiellement sur les données d'apprentissage, idéalement proches des ensembles de test. L'ancien français se caractérise par une grande variabilité, excluant une telle proximité. Nous voulons aussi ajouter des textes de moyen français, plus éloignés encore des textes du SRCMF. Il semble donc plus prometteur d'utiliser un système symbolique ou hybride pour analyser automatiquement de nouveaux textes.

Pour annoter automatiquement des textes de portugais médiéval (12e-13e s.), Rocio *et al.* (2003) ont quant à eux fait le choix d'adapter non seulement leur *parser*, mais aussi leur grammaire du portugais contemporain. Ils ont fait l'hypothèse qu'il était possible d'annoter une nouvelle langue avec des outils et ressources robustes d'une langue similaire. Nous avons choisi de procéder de manière semblable en adaptant un système basé sur une grammaire (voir section 4). Cependant, nous souhaitons couvrir une période plus vaste (9e-15e s.), dont les états de langue sont à la fois hétérogènes et éloignés du français contemporain.

3 Problématiques liées au français médiéval et différences avec le français contemporain

3.1 Syntaxe du français médiéval

Nous ne présenterons ici que quelques particularités du français médiéval qui ont un impact sur l'identification des constituants principaux. Leur ordre est libre, mais contrairement au latin, la déclinaison ne permet pas de déterminer la fonction syntaxique des syntagmes nominaux car elle est assez peu respectée et devient inefficace. D'autres critères permettent d'analyser les phrases.

ex. ordre SOV : "Tierce fiede Deu Samuel apela" (*Quatre Livres des rois*), trad. *Dieu appela Samuel trois fois.*

Tierce	fiede	Deu	Samuel	apela
Trois	fois	Dieu	Samuel	appela

"Deu" est ici sujet. L'analyse OSV (avec Samuel comme sujet) est exclue du fait que, jusqu'au 13e siècle, l'ordre OSV n'est possible qu'avec un sujet pronominal (Schøsler, 1984). L'analyse grammaticale est confirmée par le texte latin dont *Quatre Livres des rois* est la traduction :
"Et adjecit Dominus, et vocavit adhuc Samuelem tertio" [1], où "Samuelem" est un accusatif, objet de "vocavit".
Dans d'autres cas, aucune "règle" ne permet de lever l'ambiguïté :
ex. "Dolant et pansif *Lancelot* / **Vit** *la dame de la meison*" (*Le Chevalier de la charrete* de Chrétien de Troyes, tel que cité par Buridant (2000)), où *Lancelot* et *la dame de la meison* sont tous deux candidats à la fonction de sujet. C'est la suite du texte qui permet d'analyser cette phrase :
"Sel **mist** a consoil **a reison** : / Sire, por Deu et por vostre ame...", trad. de Buridant (2000) : *La dame du manoir le vit [vit Lancelot] triste et troublé. Aussi lui parla-t-elle en secret de la sorte : Sire, pour Dieu et sur votre âme...*
Ce n'est que le tour de parole suivant qui indique que le sujet des verbes "vit" et "mist a reison" est "la dame de la meison".

Il est habituel qu'un verbe transitif n'ait pas d'objet en français contemporain, mais très rare que le sujet fasse défaut. Le français médiéval est lui considéré comme une langue à sujet partiellement nul. La réalisation du sujet n'est donc pas prioritaire. L'analyse d'une phrase n'est pour autant pas nécessairement ambiguë :
ex. "Et quant il furent revenu si **fisent** savoir as barons qu'il avoient fait" (*Conquête de Constantinople* de Robert de Clari, tel que cité par T. Rainsford *et al.* (2012)), trad. *Et quand ils furent revenus,* ***ils firent*** *savoir au baron qu'ils l'avaient fait.*

Et quant il furent revenu	si	fisent	savoir	as	barons	qu'	il	avoient	fait
Et quand ils furent revenus		*ils* firent	savoir	au	baron	qu'	ils	*l'*avaient	fait

Dans cet exemple, aucun syntagme nominal ou pronom ne peut être sélectionné pour sujet de "fisent", car "barons" est précédé de "as", ce qui en fait l'objet second du verbe.

De nombreuses différences existent entre français contemporain et français médiéval. Par exemple, la graphie des mots n'était pas fixe et la variation dialectale était plus forte. Cependant, nous souhaitons

1. *Les Quatre livres des rois, traduits en français du XIIIe siècle, suivis d'un fragment de moralités sur Job et d'un choix de sermons de Saint Bernard*, de Lincy, L.R. (1841). Collection de documents inédits sur l'histoire de France, disponible à cette adresse : `https://books.google.fr/books?id=uW49AQAAMAAJ`.

tirer parti du lien entre les différents états de langue du français médiéval d'une part et le français contemporain d'autre part pour développer un *parser*.

3.2 La langue comme *continuum*

Les changements qui se sont produits du 9e au 15e siècle préfigurent la langue moderne. L'ordre des constituants principaux en est un exemple remarquable. Majoritaire en latin tardif, l'ordre SOV ne l'est plus dès le 12e siècle. C'est SVO, déjà présent dans les premiers textes d'ancien français, qui devient alors le plus fréquent. Cela a aussi un impact sur la place du complément de nom, qui a de plus en plus tendance à apparaître à droite de leur gouverneur syntaxique. Prenons l'exemple du complément déterminatif, tel que décrit par Buridant (2000). L'ancien français se passe parfois de préposition (ex. *a, de*) pour attacher le complément au nom qui le régit. Habituellement on observe cet ordre : "N1 (déterminé) + N2 (déterminatif)", mais on trouve parfois l'inverse, et plus particulièrement avec les mots *Dieu, Damedieu* ("Dieu"), *Dé hé* (malédiction), *Jesus*, ou *autrui*.
ex. "Seignors fait el por ***Deu*** merci Saintes reliques voi ici" (*Tristan* de Béroul, dans le SRCMF), trad. *Seigneurs, dit-elle, par la grâce* ***de Dieu****, je vois ici les saintes reliques.* Cependant, même avec ces mots, cet ordre n'est pas obligatoire. L'analyse N1 + N2 reste autorisée, et l'ambiguïté introduite ne peut être résolue que dans un second temps.

C'est également pendant cette période que les déterminants, apparus en latin tardif, commencent à se répandre. Le français contemporain a largement hérité ses structures du français médiéval, même si certaines ont évolué de façon conséquente. Ce sont surtout les fréquences d'apparition des phénomènes qui ont changé.

3.3 Problématiques liées à l'adaptation

Le français médiéval et le français contemporain semblent suffisamment similaires pour que nous puissions garder une certaine continuité entre le *parser* pour le français contemporain et celui pour le français médiéval. Adapter un système originellement développé pour un état de langue à des états bien antérieurs demande cependant de mettre au point une méthodologie pour le traitement des données hétérogènes.

Nous nous sommes demandé si une seule grammaire pouvait suffire à couvrir une période de sept siècles et si le système résultant ne serait pas inutilement complexe. Néanmoins, nous ne pouvons pas développer plusieurs grammaires couvrant chacune une période ou un dialecte. La langue ne saurait en effet être ainsi "découpée" car elle forme un *continuum*. De plus, les phénomènes syntaxiques ne connaissent pas nécessairement une évolution linéaire. Par exemple, l'ordre OSV (avec objet nominal) est très rare en ancien français, mais il connaît une hausse ponctuelle aux 14e-15e siècles. Il n'est donc pas possible de définir des étapes strictes que l'on décrirait dans différentes grammaires. La fréquence d'apparition de phénomènes est différente selon la date du texte, son dialecte, son domaine, son auteur et s'il est écrit en vers ou en prose. De nouveaux textes pourraient donc déjouer les prévisions, ce qui mènerait leur analyse à l'échec. Nous devons donc trouver un moyen de représenter à la fois la continuité et les variations possibles de la syntaxe à travers cette période.

Pour cela, nous avons envisagé plusieurs méthodes. La *Grammar Matrix* (Bender *et al.*, 2002) a inspiré une approche par librairies. Les descriptions de phénomènes syntaxiques seraient séparées, et chaque état de langue pourrait sélectionner les ressources dont il a besoin. Cependant, nous n'avons

pas l'ambition de développer une librairie universelle de grammaires, et nous risquerions de nous heurter aux problèmes définis précédemment (définition impossible de la syntaxe de chaque état de langue et incertitude face aux nouveaux textes). Nous avons choisi un formalisme plus modulaire : les métagrammaires.

4 Adaptation de FRMG

Les différences notables entre l'ancien et le moyen français d'une part et le français moderne d'autre part pouvent suggérer de développer un nouveau système pour traiter ces états de langue. Plutôt que développer une nouvelle métagrammaire spécifiquement conçue pour l'ancien et le moyen français, nous avons choisi d'adapter FRMG, une métagrammaire du français contemporain, qui représente une base intéressante pour décrire des états de langue anciens et hétérogènes.

4.1 Une métagrammaire pour le français médiéval

Les métagrammaires peuvent être écrites pour générer plusieurs types de grammaires. Elles ne sont pas intrinsèquement liées au formalisme des grammaires d'arbres adjoints, mais ce sont ces dernières qui nous intéressent et que nous allons présenter car FRMG en génère une.

4.1.1 Les grammaires d'arbres adjoints

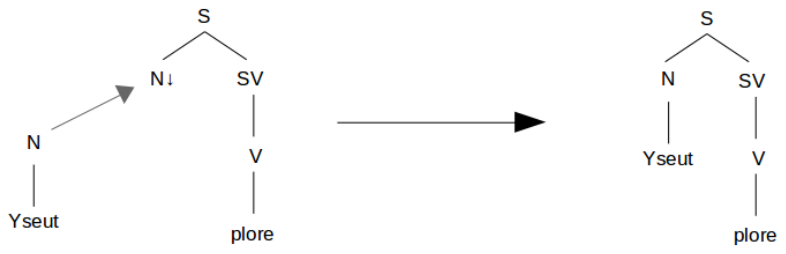
Plusieurs métagrammaires (Candito, 1999) sont basées sur les grammaires d'arbres adjoints (ou *Tree Adjoining Grammars*, désormais TAG) (Joshi *et al.*, 1975), un formalisme mathématique adapté aux langages naturels. Il est décrit formellement par le quintuple <N, T, I, A, S> où :

- N est un ensemble de non terminaux correspondant à des catégories syntagmatiques
- T est un ensemble de terminaux correspondant à des éléments lexicaux
- I est un ensemble d'arbres initiaux
- A est un ensemble d'arbres auxiliaires, qui ont chacun un noeud feuille appelé "noeud pied", symbolisé par une étoile (*) et étiqueté par un non terminal de même catégorie que le noeud racine de cet arbre
- $S \in N$ est l'axiome

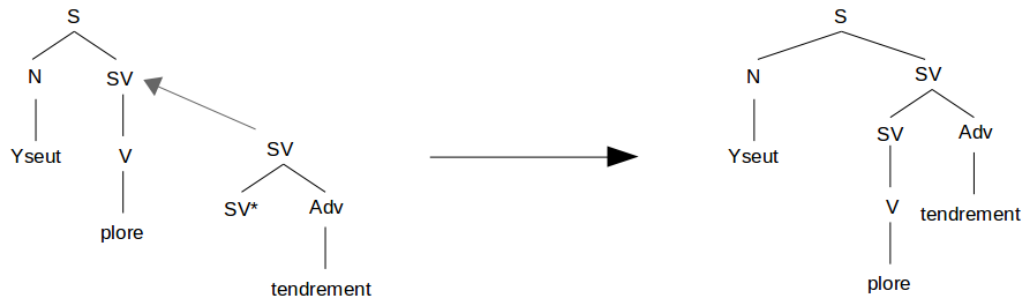
Une grammaire d'arbres adjoints est un ensemble d'arbres élémentaires. Les noeuds feuilles de ces arbres sont étiquetés par un terminal ou par un non terminal, et les noeuds non feuilles sont étiquetés par des non terminaux. Ce sont des noeuds de substitution (↓) ou des noeuds pieds.

Les deux opérations disponibles pour combiner des arbres sont :

- la substitution, obligatoire et unique, qui consiste à insérer un arbre initial ou dérivé sur un noeud à substituer dans un arbre élémentaire ou dérivé (ici N↓) :



— et l'adjonction, facultative et récursive, qui consiste à insérer un arbre auxiliaire ou son dérivé à un noeud de même catégorie (ici SV) :



Le noeud pied et la racine de l'arbre adjoint doivent aussi avoir la même catégorie (ici SV). L'adjonction peut être rendue obligatoire ou interdite, mais elle ne peut avoir lieu sur un noeud de substitution ou un noeud pied.

Lors de l'analyse d'une phrase, ses éléments terminaux sont intégrés à un arbre dérivé en tant que noeuds feuilles par les opérations décrites ci-dessus. La dérivation est complète une fois que tous les noeuds feuilles sont remplacés par des terminaux. Les TAG appartiennent à la classe des grammaires contextuelles (Joshi *et al.*, 1990) et sont plus puissantes que des grammaires hors contexte, tout en restant analysables en temps polynomial (Joshi, 1985).

Vijay-Shanker (1988) présente l'ajout de structures de traits, qui sont des couples attribut-valeur qui peuvent être attribués aux noeuds des arbres. Ils permettent d'introduire des contraintes sur leur unification. Un attribut peut recevoir une valeur absolue ou une variable pour être identique à un autre attribut. Les noeuds d'un arbre ont des structures de traits amont et aval qui doivent s'unifier lors de la dérivation, à l'exception des noeuds à substituer et des noeuds pieds. L'unification de deux structures de traits aboutit à une nouvelle structure, ou échoue, et ne permet pas l'opération de dérivation.

Sans contraintes supplémentaires, motivées linguistiquement, les TAG pourraient utiliser toutes sortes d'arbres qui ne répondraient pas aux réalisations syntaxiques d'une langue. Quatre principes ont permis de les adapter aux langages naturels (Kroch & Joshi, 1985; Abeillé, 1993) :

— principe d'ancrage lexical (ou lexicalisation) : un arbre élémentaire doit avoir au moins une ancre lexicale non vide
— principe de cooccurrence prédicat-arguments : chaque prédicat intègre ses arguments à sa structure élémentaire
— principe de consistance sémantique : un arbre élémentaire ne peut être sémantiquement "vide", les éléments fonctionnels n'étant pas perçus comme autonomes
— principe de non compositionnalité : un arbre élémentaire correspond à une seule unité sémantique

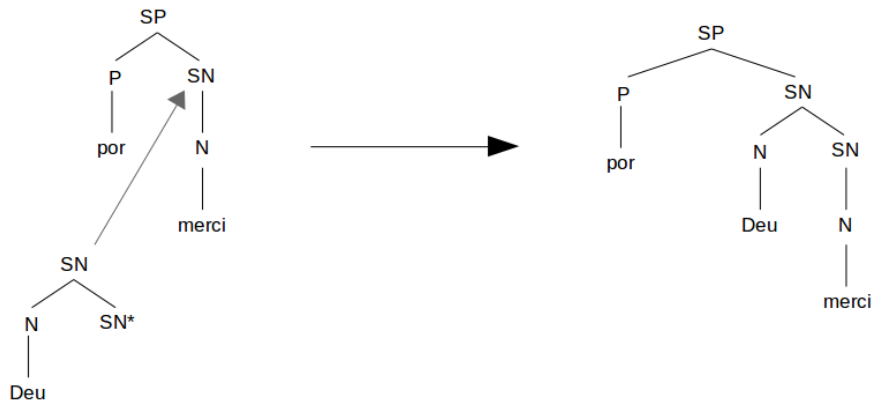


FIGURE 1 – Analyse N2 (déterminatif) + N1 (déterminé)

Il s'agit d'une convention habituellement acceptée par les linguistes. L. Kallmeyer (2010) remarque cependant que seuls les deux premiers principes sont généralement observés, le dernier demandant une prise de position claire sur ce qu'est une unité sémantique. La lexicalisation présente un intérêt pour le *parsing* car elle permet d'effectuer un tri avant l'analyse avec les mots de la phrase, qui ont des sous-catégorisations différentes.

Les structures de traits sont également mises au profit de la syntaxe, notamment pour faire peser des contraintes de tête, de pied et d'accord (en genre, nombre et cas selon la langue) sur l'unification de noeuds.

4.1.2 Avantages des TAG pour le français médiéval

Les TAG semblent avoir un pouvoir expressif suffisant pour traiter les états de langue du français médiéval, qui se caractérisent notamment par un ordre des mots plus libre qu'en français contemporain, ce qui pourrait poser problème. Cette hypothèse reste cependant à confirmer. Néanmoins, le domaine de localité étendu des TAG est un atout certain pour traiter les langues naturelles. Il permet une description locale de phénomènes, comme l'attachement d'arguments à leur prédicat, car les arbres élémentaires de la grammaire peuvent avoir une profondeur supérieure à 1, contrairement aux systèmes à base de règles. Il est ensuite possible d'insérer un nombre arbitraire de modifieurs entre ces éléments au moyen de l'adjonction, sans défaire l'arbre de départ. Le verbe peut par exemple se trouver éloigné de son objet, mais cela n'a pas de conséquence sur l'analyse.

Il est possible de lier des arbres élémentaires à certains mots du lexique, et ainsi de créer des arbres très spécifiques, n'étant utilisés qu'avec ces mots, comme dans le cas du complément déterminatif précédant son gouverneur syntaxique, qui n'est réalisé qu'avec certains mots (cf. section 3.2 et fig. 1).

Les grammaires d'arbres adjoints semblent donc tout indiquées pour l'analyse de langues naturelles, permettant à la fois la description générale de phénomènes syntaxiques et la gestion de cas particuliers. Une grammaire à large couverture est en revanche un objet complexe et difficile à maintenir à cause du grand nombre d'arbres qui la composent. Ceux-ci ont habituellement de nombreux points communs puisqu'ils représentent les différentes déclinaisons possibles de phénomènes. Il serait donc possible de factoriser ces arbres, grâce notamment au domaine étendu de localité. Un formalisme plus abstrait permet de générer ces arbres factorisés à partir d'une description modulaire de la syntaxe : les métagrammaires.

4.1.3 Modularité d'une métagrammaire

Une métagrammaire (Candito, 1999) fournit une description modulaire et hiérarchique d'une langue, organisée en classes. Celles-ci expriment des contraintes sur les constituants possibles des arbres de la grammaire et sur leurs relations (ex. accord, dominance...). Elle est destinée à être compilée en une grammaire d'arbres adjoints.

Pour ce faire, elle doit exprimer explicitement les trois "dimensions" (Candito, 1999) qui contrôlent les arbres. Selon le principe d'ancrage lexical, la description syntaxique d'un énoncé dépend du lexème qui le compose. Un lexique donne accès à des informations, comme la valence et la catégorie grammaticale, organisées en structures de traits appelées *hypertags* (Kinyon, 2000).

La catégorie grammaticale du lemme principal d'un arbre (son ancre) a ainsi une sous-catégorisation initiale, qui constitue la première dimension. Par exemple, le verbe "chasser", en ancien français comme en français contemporain, attend deux arguments (dans une diathèse (ou "voix") active, qui sert de cadre canonique) : un sujet et potentiellement un objet.
ex. "et là assortirent grant nombre d'artillerie, **qui** d'entrée *chassa* **tous les gens du duc de Calabre** hors du villaige de Charenton" (*Mémoires I* de Philippe de Commynes [2], p.61)
La valence de ce lemme peut être redistribuée, le sortant ainsi de son emploi "canonique", qui est décrit comme standard par la grammaire. Cette redistribution, la deuxième dimension, change la sous-catégorisation du lemme et potentiellement le nombre d'arguments représentés.
ex. avec nombre d'arguments conservé : "car **il** en *estoit chassé* **par le conte de Warvic**" (*Mémoires II*, p.139), où on trouve un complément d'agent qui serait sujet à la voix active
ex. avec modification du nombre d'arguments (sans complément d'agent) : "car **Pierre de Medicis** *fut chassé* ce jour" (*Mémoires VIII*, p.11)
Enfin, les fonctions syntaxiques peuvent être réalisées de différentes manières, selon l'hypothèse distributionnelle, ce qui constitue le dernier paramètre contrôlant la construction des arbres.

Ces informations doivent être représentées dans la métagrammaire, ainsi que l'architecture des arbres initiaux et adjoints. Par souci d'efficacité et pour décrire au mieux la langue, il est important d'être le plus général possible. En factorisant l'information linguistique, on peut répartir les contours généraux ou communs à plusieurs catégories dans des classes mères dont des classes plus spécifiques héritent. Par exemple, les modaux, auxiliaires et autres verbes du lexique, partagent des caractéristiques comme l'accord avec le sujet et une partie de leur structure d'arbre. Néanmoins, ils n'ont pas la même sous-catégorisation et n'ancreront donc pas les mêmes arbres. Diviser ainsi l'information facilite la compréhension et la maintenance de la métagrammaire. Les différents arbres produits sont ensuite combinés par les opérations d'adjonction et de substitution pour former une TAG.

4.2 *French Metagrammar* (FRMG)

FRMG [3] est un ensemble de 451 classes produisant 381 arbres. Cette métagrammaire a été développée parallèlement au lexique *Lefff* (Sagot, 2010), avec lequel elle partage des types et la notion d'*hypertag*, rendant possible l'ancrage des arbres de la grammaire par des mots du lexique.

2. Cet exemple est issu de la *Base de Français Médiéval* (BFM - *Base de Français Médiéval* [En ligne]. Lyon : ENS de Lyon, Laboratoire IHRIM, 2016, <txm.bfm-corpus.org>).
3. FRMG est disponible à cette adresse : `http://alpage.inria.fr/frmgwiki/`.

4.2.1 Expressivité de la métagrammaire

FRMG est une grammaire à large couverture. Depuis 2004, les campagnes d'évaluation successives ont permis de la confronter à de nouvelles données, comme des écrits journalistiques ou scientifiques. L'entraînement du *parser* et l'apprentissage de poids de désambiguïsation sur le *French Treebank* l'ont rendue plus robuste (Villemonte de la Clergerie, 2013). Actuellement, les *tweets* constituent un défi parce qu'ils présentent une ressource non standard, difficile à traiter. Ils introduisent de fortes variations langagières, de nouveaux phénomènes syntaxiques, et un usage différent de la ponctuation. La métagrammaire a été enrichie pour intégrer ces nouveautés.

Les métagrammaires ont été longtemps mises en avant pour traiter la valence des verbes et les structures verbales standard. Elles proposent en effet un cadre de description simple grâce à leur modularité et au domaine étendu de localité (si elles sont fondées sur le formalisme des TAG). Cependant, E. Villemonte de la Clergerie (2012) souligne aussi leur apport pour l'analyse des modifieurs, dont la multiplicité dépasse celle des structures verbales. Ces dernières sont décrites par quelques arbres dans FRMG, tandis que les modifieurs, qui constituent un enjeu majeur, restent peu explorés. Leur variété et leur nombre en font des phénomènes difficiles à capturer, en particulier s'ils portent sur la phrase. Il est possible de les intégrer à l'arbre d'analyse à plusieurs noeuds différents. Les modifieurs peuvent prendre diverses formes, certaines étant difficiles à analyser, comme des syntagmes nominaux ou des locutions plus ou moins figées. Ce sont généralement des contraintes sémantiques qui permettront d'identifier de tels groupes. Ces cas spécifiques sont décrits au terme d'un long recensement. Les arbres de modifieurs sont très nombreux dans FRMG, mais il s'agit de descriptions simples. Ils pourraient être factorisés, à l'instar des arbres de structures verbales.

4.2.2 Expressivité du langage SMG

La métagrammaire FRMG s'appuie sur le langage de représentation *Simple Metagrammar* (SMG) et son compilateur, *mgcomp*, qui offrent de nombreuses possibilités d'expression pour décrire une langue. Tout d'abord, des contraintes comme l'égalité, la dominance ou la précédence entre noeuds permettent de décrire des arbres élémentaires.

Lors de la compilation de la grammaire, les arbres sont factorisés au moyen d'opérateurs (Villemonte de la Clergerie, 2010) qui sont des extensions du formalisme TAG. La disjonction permet d'introduire des alternatives dans la dérivation. Les gardes (Villemonte de la Clergerie, 2010) permettent la gestion des noeuds optionnels. La répétition est possible grâce à l'étoile de Kleene. Elle est surtout utilisée pour la coordination. Enfin, l'entrelacement permet un ordre libre entre noeuds frères. Il est notable que ces opérations n'ont pas d'impact sur l'expressivité ou la complexité des TAG, mais ils permettent de réduire la taille de la grammaire, rendant son usage plus efficace.

La description de la langue dans une métagrammaire repose en partie sur les *hypertags*, des structures de traits associées à chaque mot du lexique, contenant des informations syntaxiques comme la valence et les réalisations possibles. Ces éléments sont générés en même temps que la grammaire et permettent l'ancrage du lexique dans les arbres.

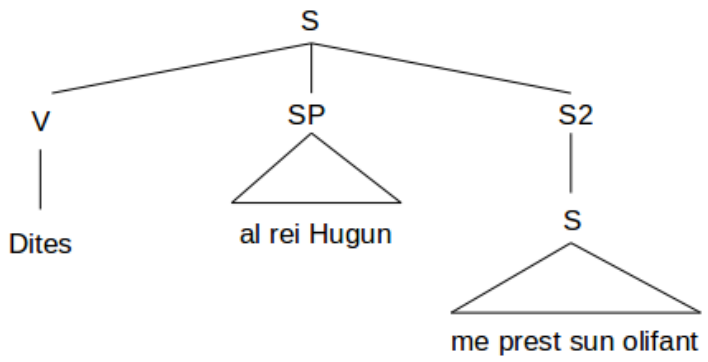


FIGURE 2 – Analyse d'une complétive sans "que"

4.3 Motivations et limites de la démarche

Plutôt que créer entièrement une nouvelle métagrammaire, nous avons choisi d'adapter FRMG au français médiéval parce que les états de langue qui le composent présentent un nombre conséquent de similarités par rapport au français contemporain. La langue moderne est aussi soumise au changement. Par exemple, pour traiter la disparition du "que" devant une complétive (Boutin, 2007), FRMG l'a rendu optionnel [4]. Son absence n'empêche plus l'analyse de la phrase. L'origine de cette tendance n'a a priori pas de lien avec le français médiéval, qui présente toutefois la même caractéristique.
ex. "Dites al rei Hugun me prest sun olifant" (*Le Voyage de Charlemagne à Jérusalem et à Constantinople*, 471, cf. fig.2), trad. par Buridant (2000) : *Dites au roi Hugon qu'il me prête son olifan.*
La conjonction peut être élidée car le subjonctif dans la subordonnée est un indice suffisant. Il semble donc que les différents états de langue présentent une syntaxe similaire et des cas de variation. Ceux-ci peuvent être inclus à la métagrammaire, qui est aisément modifiable.

En revanche, il n'est pas aisé de déterminer la probabilité d'une analyse. Lors de la désambiguïsation, des analyses sont préférées à d'autres en fonction du poids qui leur est attribué, obtenu grâce à un apprentissage sur *treebank* (Villemonte de la Clergerie, 2013). Ces poids seront appris sur le SRCMF, et seront très différents de ceux appris pour FRMG à partir du *French Treebank*. Cependant, le formalisme des *treebanks* ne correspond pas à celui des métagrammaires, ce qui limite l'apport d'une ressource pour déterminer les poids des analyses.

4.4 Leviers d'action pour adapter la métagrammaire

4.4.1 Relâchement de contraintes

Dans un premier temps, nous pouvons considérer la tâche d'adaptation de FRMG comme le simple relâchement de nombreuses contraintes. Une description plus générale, moins contrainte en amont, permet à la grammaire de traiter ces données hétérogènes. Le français médiéval peut être vu comme une succession d'états de langue préfigurant le français contemporain, avec plus de variations, une graphie non fixe et des règles d'accord moins respectées. Encore en moyen français, C. Marchello-Nizia (1979) relève des cas d'absence d'accord verbal avec le sujet, qu'il soit en position préverbale

4. Une garde porte sur la conjonction. Si celle-ci est absente, l'analyse de la subordonnée peut être faite, mais elle est pénalisée lors de la désambiguïsation car cette tournure reste rare en français contemporain.

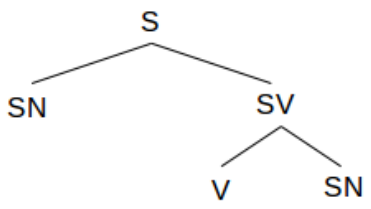


FIGURE 3 – Représentation traditionnelle des constituants principaux

ou postverbale.
ex. "si luy **avoit** *jeunesse et crainte* les yeulz si bandez que en rien il ne s'appercevoit du bien qu'on luy vouloit" (*Cent nouvelles nouvelles*, tel que cité par C. Marchello-Nizia), trad. *La jeunesse et la crainte lui **avaient** bandé les yeux, de sorte qu'il ne voyait pas qu'on lui voulait du bien.*
Il est plus fréquent, et ce dès l'ancien français, de trouver un substantif singulier à valeur collective (*gent, peuple...*) comme sujet d'une verbe au pluriel (Hasenohr & Raynaud de Lage, 1993; Buridant, 2000). Inversement, il est possible de trouver des verbes au singulier alors qu'ils ont plusieurs sujets exprimés (voir l'exemple de *Cent nouvelles nouvelles* ci-dessus). L'accord du participe passé varie beaucoup plus qu'en français contemporain. Employé avec *avoir*, il peut s'accorder avec l'objet placé avant ou après l'auxiliaire, ou même avec un substantif proche, rattaché à l'objet. Employé avec l'auxiliaire *être* (*estre*), il s'accorde avec le sujet, sauf, parfois, si celui-ci est postposé. Rendre plus souple la contrainte d'accord verbal semble donc être une mesure importante pour le traitement du français médiéval.

Des études ont été faites pour confirmer et quantifier l'impact des métadonnées sur le *parsing* (Guibon *et al.*, 2014, 2015). Les auteurs ont évalué des modèles appris sur différents ensembles d'apprentissage, et ont observé que le dialecte et la date sont les métadonnées les plus discriminantes (même si le domaine du texte et sa forme sont aussi une cause importante de variation). Dans leur article, T. Rainsford *et al.* (2012) tirent la même conclusion : "à des caractéristiques externes différentes peuvent être associées des propriétés langagières différentes". Leur étude de la zone préverbale en ancien français met en lumières des différences dues à la forme des textes. Sur l'ensemble du corpus étudié, la zone préverbale n'accueille habituellement qu'un élément, l'ancien français étant une langue majoritairement à verbe second (V2). Cependant, en prose les exceptions sont assez régulières. Les phrases à deux éléments préverbaux ou plus ont un circonstant (souvent *si*), ou une subordonnée précédant le sujet, ou un circonstant à valeur énonciative. Les textes en vers présentent plus de variété. On trouve par exemple plus souvent l'antéposition de deux satellites. Il semble donc qu'il vaut mieux privilégier une description très générale des phénomène syntaxiques pour rendre acceptables les multiples variations existantes.

4.4.2 Contraintes différentes

L'ordre libre des mots est une des principales caractéristiques qui différencie le français médiéval de la langue contemporaine. FRMG a une représentation traditionnelle des constituants principaux (cf. fig.3) : le sujet et le syntagme verbal (SV) sont des noeuds frères, tandis que l'objet est sous SV (Thomasset & Villemonte de la Clergerie, 2005).

Pour autoriser un ordre libre de ces constituants, nous avons tiré profit de l'entrelacement (symbolisé par #), et avons donc remonté l'objet au même niveau que les deux autres (cf. fig.4). La grammaire

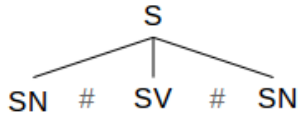


FIGURE 4 – Représentation des constituants principaux dans notre métagrammaire

peut ainsi reconnaître les six ordres possibles du français médiéval : SOV, SVO, OSV, OVS, VSO, VOS.

Pour cette métagrammaire du français médiéval, nous ne prenons pas en compte la ponctuation. Elle n'est pas présente dans le SRCMF, qui sera utilisé pour évaluer la métagrammaire et entraîner les modèles de désambiguïsation. Ce corpus est cependant segmenté en phrases. Les descriptions de phénomènes habituellement accompagnés de marques de délimitation, comme l'incise, ne peuvent donc pas reposer sur la présence de celles-ci. Nous n'excluons cependant pas d'ajouter la ponctuation à notre métagrammaire dans un second temps, si sa présence est souhaitable.

4.4.3 Facettes

Nous prévoyons d'implémenter un nouveau mécanisme pour filtrer l'analyse syntaxique selon les métadonnées des textes. Certaines contraintes ne s'appliquent qu'à des états de langue identifiés, et nous voulons restreindre leur activation à ces cas particuliers. Par exemple, les formes atones des pronoms objet sont placées avant le verbe, mais elles peuvent être postposées en picard, et même être élidées (Buridant, 2000). En nous appuyant sur des informations comme la date, le dialecte, la forme ou le domaine, nous voulons pouvoir permettre ou interdire certaines analyses, ou leur donner un poids plus ou moins important. Nous ne savons pas encore si nous souhaitons produire une grammaire par ensemble de facettes ou s'il est préférable d'appliquer des gardes à une seule grammaire et de les activer dynamiquement lors de l'analyse.

5 Conclusion et perspectives

L'adaptation de FRMG au français médiéval semble prometteuse, même si ces états de langue ancienne posent de nombreux défis pour l'analyse automatique. Des changements sont indispensables dans la métagrammaire. Il peut s'agir du simple relâchement de certaines contraintes pour ne décrire que ce qui est universel dans la grammaire, ou des changements de description. Nous souhaitons développer une métagrammaire assez générale pour traiter tous les phénomènes, quelle que soit leur fréquence dans les textes à annoter, mais en réduisant au mieux le nombre d'analyses possibles pour limiter la complexité.

De nombreuses problématiques restent à aborder, dont la description de phénomènes et l'inclusion d'un mécanisme de facettes pour filtrer l'application de ces contraintes. La fouille d'erreurs en sortie du *parser* permettra d'accélérer la mise au point de la grammaire et du lexique. A terme, nous voulons développer une méthodologie réutilisable pour traiter des données hétérogènes, incluant des arbres de décision en amont pour modifier le modèle ou la grammaire elle-même en fonction des métadonnées.

Références

ABEILLÉ A. (1993). *Les Nouvelles Syntaxes*. Armand Colin.

ABEILLÉ A. (2002). *Une grammaire électronique du français*. CNRS Editions.

BENDER E. M., FLICKINGER D. & OEPEN S. (2002). The grammar matrix : An open-source starter-kit for the rapid development of cross-linguistically consistent broad-coverage precision grammars. *COLING-02 on Grammar engineering and evaluation*.

BOUTIN A. B. (2007). *De et que subordonnants, et variation en français*. *Linx* [En ligne], 57, mis en ligne le 15 février 2011. http ://journals.openedition.org/linx/280.

BURIDANT C. (2000). *Grammaire nouvelle de l'ancien français*. Sedes.

CANDITO M.-H. (1999). *Organisation modulaire et paramétrable de grammaires électroniques lexicalisées, application au français et à l'italien*. PhD thesis, Université Paris 7.

CRABBÉ B. (2005). *Représentation informatique de grammaires fortement lexicalisées, Application à la grammaire d'arbres adjoints*. PhD thesis, Université Nancy 2.

GUIBON G., TELLIER I., CONSTANT M., PRÉVOST S. & GERDES K. (2014). Parsing poorly standardized language dependency on old french. In V. HENRICH, E. HINRICHS, D. DE KOK, P. OSENOVA & A. PRZEPIÓRKOWSKI, Eds., *Proceedings of the Thirteenth International Workshop on Treebanks and Linguistic Theories (TLT13)*, p. 51–61, Tübingen, Allemagne.

GUIBON G., TELLIER I., PRÉVOST S., CONSTANT M. & GERDES K. (2015). Searching for discriminative metadata of heterogenous corpora. In M. DICKINSON, E. HINRICHS, A. PATEJUK & A. PRZEPIÓRKOWSKI, Eds., *Proceedings of the Fourteenth International Workshop on Treebanks and Linguistic Theories (TLT14)*, p. 72–82, Varsovie, Pologne.

GUILLOT-BARBANCE C., HEIDEN S. & LAVRENTIEV A. (2017). Base de français médiéval : une base de référence de sources médiévales ouverte et libre au service de la communauté scientifique. *Diachroniques n.7*, p. 168–184.

HASENOHR G. & RAYNAUD DE LAGE G. (1993). *Introduction à l'ancien français de Guy Raynaud de Lage*. Sedes.

JOSHI A. K. (1985). Tree adjoining grammars : How much context-sensitivity is required to provide reasonable structural descriptions ? *Natural Language Parsing*.

JOSHI A. K., LEVY L. S. & TAKAHASHI M. (1975). Tree adjunct grammars. *Journal of Computer and System Sciences*.

JOSHI A. K., VIJAY-SHANKER K. & WEIR D. (1990). The convergence of mildly context-sensitive grammar formalisms. *University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-90-01*.

KALLMEYER L. (2010). *Parsing beyond context-free grammars*. Springer.

KINYON A. (2000). Hypertags : Beyond pos tagging. In D. N. CHRISTODOULAKIS, Ed., *Lecture Notes in Computer Science*, Berlin, Heidelberg : Springer.

KROCH A. S. & JOSHI A. K. (1985). The linguistic relevance of tree adjoining grammar. *University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-85-16*.

MARCHELLO-NIZIA C. (1979). *Histoire de la langue français aux XIVe et XVe siècles*. Bordas.

PRÉVOST S. & STEIN A. (2013). *Syntactic Reference Corpus of Medieval French (*SRCMF*)*. [version 0.92]. ENS de Lyon/ILR Stuttgart.

RAINSFORD T., GUILLOT-BARBANCE C., LAVRENTIEV A. & PRÉVOST S. (2012). La zone préverbale en ancien français : apport des corpus annotés. *SHS Web of Conferences*.

ROCIO V., ALVES M. A., LOPES J. G., XAVIER M. F. & VICENTE G. (2003). Automated creation of a medieval portuguese partial treebank. In A. ABEILLÉ, Ed., *Treebanks : Building and Using Parsed Corpora*.

SAGOT B. (2010). The lefff, a freely available and large-coverage morphological and syntactic lexicon for french. *7th international conference on Language Resources and Evaluation (LREC 2010)*.

SCHØSLER L. (1984). *La déclinaison bicasuelle de l'ancien français : son rôle dans la syntaxe de la phrase, les causes de sa disparition*. Odense University Press.

THOMASSET F. & VILLEMONTE DE LA CLERGERIE E. (2005). Comment obtenir plus des méta-grammaires. In *Actes de TALN 2005 (Traitement automatique des langues naturelles)*, Dourdan : ATALA.

VIJAY-SHANKER K. (1988). *A Study of Tree Adjoining Grammars*. PhD thesis, University of Pennsylvania.

VILLEMONTE DE LA CLERGERIE E. (2005). From metagrammars to factorized tag/tig parsers. In H. BUNT, R. MALOUF & A. LAVIE, Eds., *Proceedings of the Ninth International Workshop on Parsing Technology*, p. 190–191, Vancouver, Canada : Association for Computational Linguistics.

VILLEMONTE DE LA CLERGERIE E. (2010). Building factorized tags with meta-grammars. *The 10th International Conference on Tree Adjoining Grammars and Related Formalisms - TAG+10*.

VILLEMONTE DE LA CLERGERIE E. (2012). Etude du traitement de certains compléments de phrase dans le cadre d'une méta-grammaire. In J. RADIMSKY, Ed., *LGC'12 - 31ème 30ème Colloque international sur le Lexique et la Grammaire*, Nové Hrady, République tchèque.

VILLEMONTE DE LA CLERGERIE E. (2013). Improving a symbolic parser through partially supervised learning. In *The 13th International Conference on Parsing Technologies (IWPT)*, Naria, Japan : Springer.

Apport des termes complexes pour enrichir l'analyse distributionnelle en domaine de spécialité

Mérième Bouhandi
LS2N, Université de Nantes
merieme.bouhandi@ls2n.fr

RÉSUMÉ

L'essor et les performances des modèles de sémantique distributionnelle sont principalement dus à l'accroissement de la quantité de données textuelles disponibles ainsi qu'à la généralisation des méthodes neuronales pour la construction de ces modèles. La qualité des représentations distribuées est souvent corrélée à la quantité de données disponibles et les corpus spécialisés, généralement d'une taille modeste, se trouvent de ce fait pénalisés. Alors que la plupart des modèles de sémantique distributionnelle traitent de mots isolés, nous partons de l'hypothèse que l'exploitation des termes, notamment complexes, est essentielle notamment en langue de spécialité car ils sont porteurs d'une dimension sémantique supplémentaire. Ainsi, nous évaluons une méthode de généralisation des contextes distributionnels par un mécanisme d'inclusion lexicale reposant sur les termes complexes. Nos différentes représentations distributionnelles sont ensuite confrontées à une tâche d'extraction de concepts médicaux à partir des rapports médicaux proposée par l'édition 2010 du challenge i2b2.

ABSTRACT

Multi-words terms impact in improving domain-specific distributed representations

The success of distributional semantic models (DSM) is mainly due to the increase in the amount of available textual data and the democratisation of neural methods for building these models. The quality of distributed representations is often correlated with the quantity of data available, and domain-specific corpora are generally very small in size. While most DSMs deal with words, the contribution of terms, especially multi-word terms, is essential in domain-specific language because they carry an additional semantic dimension. However, they generally remain poorly represented in the context vectors and are often excluded when measuring similarity. Thus, we evaluate a method of generalising distributional contexts by performing lexical inclusion, using relations acquired in the corpora. The different distributional representations of our terms are then tested on a task of medical concepts classification from clinical reports proposed by the 2010 edition of the i2b2/VA Challenge.

MOTS-CLÉS : sémantique distributionnelle, inclusion lexicale, plongements de mots, extraction d'information.

KEYWORDS: distributional semantics, lexical inclusion, word embeddings, information extraction.

1 Introduction

L'analyse distributionnelle est un domaine renouvelé depuis la généralisation des approches neuronales qui sont venues concurrencer les approches traditionnelles à base de sacs de mots. Cette analyse repose sur l'hypothèse que les mots se trouvant dans des contextes d'apparition similaires tendent à

avoir un sens proche. Selon l'hypothèse harrissienne, le degré de recouvrement des contextes de deux mots permet de déterminer leur proximité sémantique. L'idée sous-jacente aux méthodes d'analyse distributionnelle est donc de regrouper les mots supposés sémantiquement proches.

Ces approches, qui se sont largement démocratisées notamment grâce à l'accroissement de la quantité de données textuelles disponibles sur le net, permettent de faire reposer le calcul du sens sur l'analyse statistique des contextes des mots dans un corpus et de ne pas avoir besoin de recourir à des sources extérieures de connaissance. Ces ressources, constituées, par exemple, d'archives de journaux, de publications glanées sur les réseaux sociaux, recommandations de films ou de restaurants ou de textes généraux issus du Web a permis la généralisation des méthodes distributionnelles et l'application de ces méthodes à grande échelle et sur une multitude de tâches. Les performances de ces méthodes se sont également nettement améliorées avec l'utilisation des méthodes neuronales pour la construction des modèles (Mikolov *et al.*, 2013; Bojanowski *et al.*, 2016).

Deux approches principales se côtoient. La première, dite approche explicite, est une approche par sac de mots. Celle-ci consiste en la création d'une matrice pondérée de mots cibles et des contextes distributionnels qui leur sont associés suivie ou non d'une méthode de réduction de dimension. La seconde, dite approche prédictive, consiste en l'utilisation de méthodes neuronales pour créer des représentations denses de faible dimension, en apprenant à prédire de façon optimale les contextes distributionnels autour d'un mot cible. Cependant, la qualité des représentations distribuées obtenues à travers ces méthodes est souvent corrélée au volume de données disponibles et des corpus généralistes de très grande taille sont utilisés pour générer ces matrices.

Or, dans le cas d'un domaine de spécialité, les corpus sont généralement de taille très modeste et ces méthodes se retrouvent dès lors moins efficaces. De plus, les corpus en langue de spécialité, qui véhiculent un savoir propre au domaine en question, présentent des différences fondamentales avec les domaines généralistes, autant d'un point de vue structurel que dans leur élaboration linguistique, étant donné l'importance des termes pour dénoter les notions du domaine. Des adaptations des méthodes disponibles sont donc alors nécessaires.

Ainsi, nous partons de l'hypothèse que l'exploitation des termes, notamment complexes, est essentielle notamment en langue de spécialité (Morin, 2007). Alors que la plupart des modèles de sémantique distributionnelle traitent de mots isolés, ces unités terminologiques sont porteuses d'une dimension sémantique supplémentaire et les relations classiques qu'elles ont entre elles peuvent généralement être trouvées dans une ressource terminologique propre au domaine. Les termes complexes se classent dans deux principales catégories (Périnet, 2015) : des multi-termes (polylexicaux) tel que *"muscle ischio-jambier"* ou des termes morphologiquement complexes, tel *"cardiovasculaire"*. Les constituants de ces termes sont liés entre eux par une variété de relations : modification (ex. *"muscle ischio-jambier"*), complémentation (ex. *"perforation intestinale"*), composition morphologique (ex. *"cardiovasculaire"*) ou encore affixation (ex. *"hyperthyroïdie"*) ou suffixation (ex. *"parasitose"*).

L'article se présente comme suit. Dans un premier temps (section 2), nous allons voir en quoi consistent ces méthodes distributionnelles à base de sacs de mots et neuronales et les différentes manières de les mettre en œuvre, en prenant en compte les spécificités des corpus en langue de spécialité. Dans un second temps (section 3), nous présenterons quelques expériences exploratoires que nous avons effectuées dans le but d'étudier l'effet sur les représentations distribuées de la prise en compte des termes complexes ainsi que de l'inclusion lexicale dans les vecteurs de contexte. Finalement (section 4), nous discutons des perspectives de travaux futurs.

2 Méthodes distributionnelles

Les méthodes d'analyse distributionnelle s'appuient sur l'hypothèse distributionnelle selon laquelle les mots se trouvant dans des contextes d'apparition similaires tendent à être sémantiquement proches. Le contexte, qui peut être *"une proposition, une phrase, un paragraphe, un document est souvent réduit aux mots qui le constituent"* (Zweigenbaum & Habert, 2006).

Il existe plusieurs manière de définir cette notion de contexte dans le cadre de l'analyse distributionnelle. Tout d'abord, deux type de modèles existent. Certains sont basés sur le document comme l'analyse sémantique latente (LSA, *Latent Semantic Analysis*) (Deerwester *et al.*, 1990). Ces méthodes partent du principe que le document entier fait partie du contexte d'un mot étudié m, par observation des occurrences du dit mot dans le contexte du document. Les autres méthodes sont basées sur les mots : le contexte d'un mot m est alors représenté par les mots se trouvant dans son voisinage immédiat. Lequel peut être de nature graphique (les contextes sont récupérés à l'aide d'une fenêtre d'une taille donnée, centrée sur le mot m) (Manning *et al.*, 2008; Périnet, 2015; Jurafsky & Martin, 2018) ou syntaxique (les relations de dépendance syntaxique déterminent le contexte de m) (Fabre *et al.*, 2014). Des expériences ont montré que les modèles à base de dépendances syntaxiques permettaient souvent d'acquérir des relations paradigmatiques alors que les modèles graphiques à base de co-occurrents tendent à extraire des concepts associés (Fabre *et al.*, 2014; Périnet, 2015). Combiner ces modèles pourrait alors permettre de tirer parti des avantages de chacun d'entre eux.

Il existe deux représentations computationnelles principales pour ces modèles distributionnels : les méthodes utilisant des graphes et les méthodes vectorielles. Les méthodes vectorielles reposent sur la représentation des mots dans un espace vectoriel en fonction de leurs propriétés distributionnelles. Une matrice pondérée de termes $\times$ termes ou de termes $\times$ contextes est générée et chaque vecteur représente alors à la fois les informations contextuelles et les données statistiques distributionnelles (Périnet, 2015). Ces représentations sont particulièrement prisées car elles ont l'avantage de permettre un calcul facile de la similarité sémantique entre deux termes : il suffit alors de calculer la distance entre leurs vecteurs de contexte.

2.1 Approches *classiques* par sacs de mots

Les méthodes distributionnelles classiques, dites approches en sacs de mots, reposent sur la création et l'exploitation d'une matrice de co-occurrences (Jurafsky & Martin, 2018; Manning *et al.*, 2008; Périnet, 2015). Cette matrice de *termes* $\times$ *contextes* est obtenue à travers l'acquisition dans le corpus des occurrences des mots cibles et de leurs contextes. Ces matrices ont l'avantage de permettre un calcul facile de la similarité sémantique entre deux termes. Celle-ci est calculée en prenant en compte la distance entre les deux vecteurs de contexte, en mesurant leur angle par exemple ou en comparant leur nombre de contextes communs par rapport à l'ensemble des contextes de ces mots.

Le simple calcul des occurrences des mots ne suffit généralement pas. Un mot avec une fréquence élevée peut ne pas être très discriminant en tant que contexte pour un mot cible, mais en prenant en compte uniquement son nombre d'occurrences une importante toute particulière lui sera accordée. La table 1 illustre la matrice de termes $\times$ contextes pour le texte *"The patient is an 81 year old female with a history of cerebrovascular accident , atrial fibrillation , hypothyroidism and dementia"*.

Différents moyens de pondérer cette matrice existent. L'un d'entre eux est l'information mutuelle ponctuelle ou PMI (Church & Hanks, 1990), qui permet de quantifier la probabilité que deux mots w (*"word"*) et c (*"context"*) apparaissent ensemble (distribution jointe) comparée à la probabilité qu'ils

	patient	year	old	female	history	cerebrovascular accident	atrial fibrillation	hypothyroidism	dementia
patient	1	1	1						
history			1	1	1	1	1		
cerebrovascular accident				1	1	1	1	1	
atrial fibrillation					1	1	1	1	1
hypothyroidism						1	1	1	1
dementia							1	1	1

TABLE 1: Matrice de co-occurences des mots cibles et de leurs contextes

soient tout à fait indépendants (distribution indépendante) :

$$PMI_{w,c} = log_2 \frac{P_{w,c}}{P_w P_c}$$

Les valeurs de la $PMI_{w,c}$ peuvent être positives ou négatives. Cependant, il est plus naturel de calculer la similarité des deux mots que leur "dissimilarité". La $PPMI_{w,c}$ ou information mutuelle ponctuelle positive permet de pallier ce problème en remplaçant les valeurs négatives par 0.

$$PPMI_{w,c} = \begin{cases} PMI_{w,c} & \text{si } PMI_{w,c} > 0 \\ 0 & \text{sinon} \end{cases}$$

Un autre problème se pose alors : les mots particulièrement rares tendent à avoir des valeurs de $PPMI_{w,c}$ particulièrement élevées. On peut réduire ce phénomène en remplaçant P_c par $P_{\alpha c}$ (Levy *et al.*, 2015). Généralement, la valeur de α est définie à $0,75$.

$$P_{\alpha c} = \frac{count(c)^\alpha}{\sum_c count(c)^\alpha} \qquad PMI_{\alpha w,c} = log_2 \frac{P_{w,c}}{P_w P_{\alpha c}}$$

$$PPMI_{\alpha w,c} = \begin{cases} PMI_{\alpha w,c} & \text{si } PMI_{\alpha w,c} > 0 \\ 0 & \text{sinon} \end{cases}$$

À l'issue de cette construction de matrice de co-occurrences pondérée, nous obtenons pour chacun de nos mots cibles un vecteur de contextes qui le représente.

Le principal problème rencontré avec l'utilisation de ces représentations vectorielles est ce que l'on appelle "*the curse of dimensionality*". En effet, et à plus forte raison dans les domaines de spécialité, peu de mots co-occurrents avec beaucoup de contextes et cela résulte en des matrices de très grande dimension et très éparses, avec beaucoup d'éléments à zéro. Au-delà des problèmes liés au temps de calcul de ces matrices, les matrices creuses sont moins efficaces que les matrices denses pour capturer certaines relations (la synonymie notamment).

Plusieurs méthodes ont été proposées pour pallier ce problème de matrice creuse dont la Décomposition en Valeurs Singulières (SVD, *Singular Value Decomposition*) (Jurafsky & Martin, 2018; Manning *et al.*, 2008). Cette méthode, initialement appliquée à des matrices de termes $\times$ documents

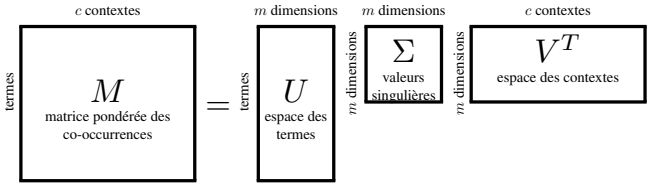


FIGURE 1: Décomposition de la matrice de termes × contextes en ses valeurs singulières

dans le cadre de la méthode LSA et dans un contexte de recherche d'information. Elle se présente comme suit : une matrice de termes × contextes M peut être décrite telle que $M = U\Sigma V^T$ (figure 1).

On peut ensuite ne conserver qu'un nombre k ($k \leq m$) de dimensions (figure 2). Ces k top dimensions correspondent alors aux k plus importantes valeurs singulières et un produit des matrices U, Σ et V^T renvoie une bonne approximation, dense de surcroît, de notre matrice creuse M de départ.

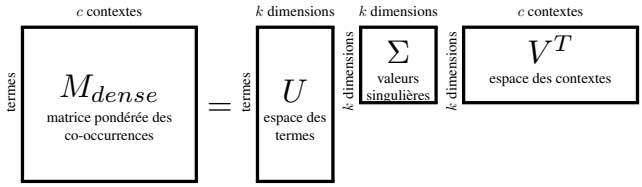


FIGURE 2: Décomposition de la matrice de termes × contextes en ses valeurs singulières en ne gardant les top k plus importantes valeurs singulières

2.2 De la sémantique distributionnelle *classique* aux réseaux de neurones : plongements de mots ou *word embeddings*

Une autre approche consiste à utiliser des réseaux de neurones. Bien qu'il existe de nombreuses sortes de plongements de mots (Bengio *et al.*, 2003; Collobert & Weston, 2008), nous nous ne concentrons ici que sur deux de ces méthodes. L'outil Word2vec (Mikolov *et al.*, 2013) permet de calculer des représentations vectorielles en utilisant deux architectures différentes : le modèle CBOW (*Continuous Bag of Words*) et le modèle Skip-Gram. CBOW entraîne le réseau de neurones pour prédire un mot w en fonction de son contexte dans la phrase. Pour la phrase *"the quick brown fox jumps over the lazy dog"* et une fenêtre de taille 2, le modèle va prédire le mot *"brown"* à partir de ses contextes w_{-1} (*"quick"*) et w_1 (*"fox"*). Skip-Gram prédit les contextes en fonction d'un mot w. Ainsi, pour la même phrase, le modèle essaye de prédire w_{-1} (*"quick"*) et w_1 (*"fox"*) à partir du mot cible w (*"brown"*).

FastText (Bojanowski *et al.*, 2016) permet de faire même chose en s'appuyant également sur les artchitectures Skip-Gram et CBOW mais en utilisant comme contexte l'information contenue au niveau des *n-grams* de caractères à la place des mots complets. Avec un découpage en $n = 6$, *"coronarography"* devient donc : *"<coron", "corona", "oronar", "onarog", "narogr", "narogr", "arogra", "rograp", "ograph", "graphy", "raphy>"*.

Cette méthode est particulièrement intéressante, et à plus forte raison dans les domaines de spécialité comme la médecine, où les mots sont souvent morphologiquement complexes ("*radio-*", "*cardio-*", "*-ite*", "*hyper-*", "*hemo-*", etc). Cela permet de rapprocher des mots ayant une composition similaire, *"coronarography"* et *"mammography"* par exemple, sur la base de leur composition (ici, les deux mots partagent le suffixe *"graphy"* qui fait référence à un mode de représentation écrite ou imagée).

2.3 Les plongements de mots ou *word embeddings* contextuels

Word2Vec et *FastText* ont pour inconvénient qu'ils n'apprennent qu'une représentation par mot – même si ce dernier est polysémique. De nouvelles approches sont apparues récemment pour construire des *word embeddings* contextuels. Nous présentons ici deux méthodes : *ELMo* et *BERT*.

ELMo (Peters *et al.*, 2018) (*Embeddings from Language Models*) est un modèle pré-entrainé dont l'architecture est basée sur des LSTM bidirectionnels. Les couches de niveau supérieur du modèle capturent les aspects contextuels des embeddings de mots tandis que les couches de niveau inférieur en capturent les aspects plutôt syntaxiques tout en prenant en compte la polysémie d'un mot compte-tenu de son contexte. De plus, le modèle apprend des représentations sur les caractères et permet, à l'instar de FastText, d'avoir un modèle robuste capable de gérer le problème des mots hors vocabulaire. Cependant, de par son architecture, à chaque étape de l'entraînement, *ELMo* se voit contraint de concaténer la "vue" arrière et la "vue avant".

BERT (Devlin *et al.*, 2018) (*Bidirectional Encoder Representations from Transformers*) est également un modèle de langue profond qui a rapidement fait office d'état de l'art dans de nombreuses tâches de traitement du language. Il s'appuie sur des couches de *transformers*. Il est donc directement bidirectionnel, ce qui lui donne un avantage par rapport à *ELMo*.

2.4 Représentation des termes complexes

Toutes ces approches sont généralement utilisées pour extraire les contextes de termes simples. Pour la représentation distribuée de représentation de termes complexes ou d'expression multi-mots, l'état de l'art propose d'additionner les représentations des mots qui les composent (Mitchell & Lapata, 2010). D'autres méthodes existent, notamment la fixation des termes complexes dans le corpus, avant entraînement, avec des résultats moins intéressants (Hazem & Daille, 2018).

2.5 Mesures de similarité

Après avoir extrait et pondéré les contextes pour chacun des termes, il convient alors de comparer les couples de mots cibles en calculant un score de similarité sémantique entre leurs vecteurs de contexte. Pour cela, il existe plusieurs mesures de similarités. Le choix d'une bonne mesure dépend de plusieurs facteurs, notamment le type de corpus utilisé, le type de relations visées et la dispersion des données. Deux mesures sont communément utilisées pour l'analyse distributionnelle (Périnet, 2015; Jurafsky & Martin, 2018) : l'indice de Jaccard et le cosinus.

L'indice de Jaccard compare le nombre de contextes communs de deux mots cibles m et n à l'ensemble des contextes de ces deux mots $C(m)$ et $C(n)$. Ainsi :

$$jaccard(m,n) = \frac{|C(m) \cap C(n)|}{|C(m) \cup C(n)|}$$

Le cosinus permet de déterminer la similarité sémantique de deux vecteurs en calculant leur angle. Cette méthode de calcul de similarité ne prend en compte ni la distance, ni la longueur des vecteurs et contrairement à l'indice de Jaccard, elle permet la prise en compte de la fréquence d'apparition des mots dans le vecteur de contexte :

$$cosine(m,n) = \frac{\sum_{i=1}^{N} m_i n_i}{\sqrt{\sum_{i=1}^{N} m_i^2}\sqrt{\sum_{i=1}^{N} n_i^2}}$$

2.6 Évaluation des représentations distribuées

Il est difficile d'évaluer ces représentations distribuées notamment parce qu'elles capturent une notion très large de proximité sémantique (Fabre *et al.*, 2014; Périnet, 2015). Les méthodes d'évaluation consistent souvent à comparer les relations obtenues entre le termes aux termes et relations extraites d'une ressource terminologique propre au domaine, à mettre à l'épreuve les représentations sur une tâche donnée où l'étude des relations entre les mots est nécessaire ou, par une très coûteuse évaluation intrinsèque, soumettre les relations qui en sont issues au jugement humain. Malgré cela, il y a un souvent un décalage entre les ressources prévues pour l'évaluation et les résultats fournis.

On sait cependant que la qualité des représentations distribuées est souvent corrélée à la quantité de données disponible ; en effet, cela permet aux modèles d'être moins affectés par le problème de dispersion des données. Or, à l'exception de quelques cas particuliers, les corpus de spécialité sont généralement de très petite taille – souvent de l'ordre du million de mots. De plus, nous partons de l'hypothèse que l'apport des termes complexes lors de la construction de nos vecteurs de contexte est important car ils sont porteurs d'une dimension sémantique supplémentaire (Morin, 2007). Cependant, selon Périnet (2015), les termes complexes sont parfois trop peu représentés dans les vecteurs de contexte. De ce fait, ils se retrouvent alors généralement écartés du calcul de similarité.

3 Approche poursuivie

Pour explorer cette problématique, nous partons de l'approche de Périnet (2015) qui consiste en la réduction de la dimensionnalité de la matrice de contexte par abstraction des contextes distributionnels. La méthode d'abstraction utilisée, l'inclusion lexicale, est une méthode de généralisation sémantique qui consiste à remplacer un terme complexe par sa tête. Dans le cas de Périnet (2015), cela se fait à partir de relations extraites automatiquement du corpus. L'inclusion lexicale permet de garder l'idée principale du terme complexe considéré en se débarrassant de ses spécificités. Par exemple *"examen médical"* devient *"examen"* et *"radiographie du col du fémur"* de vient *"radiographie"*. Ainsi, on peut dire que si un terme est inclus dans un autre, il existe une relation d'hyperonymie entre eux (Grabar & Zweigenbaum, 2004). Cela permet de réduire la dimensionnalité de la matrice tout en augmentant la fréquence d'apparition de certains contextes et donc d'augmenter la similarité entre les termes.

Nos représentations distribuées, construites sur un corpus spécialisé de petite taille, seront ensuite confrontées à une tâche de classification.

3.1 Données utilisées

L'évaluation de nos représentations distribuées a été réalisée sur la base du corpus proposé par l'édition 2010 du i2b2/VA *Natural Language Processing Challenges for Clinical Records*. Ce challenge comprend une tâche axée sur l'extraction de concepts médicaux à partir des rapports médicaux anonymisés. Initialement, 394 rapports annotés pour le corpus d'entraînement, 477 pour le test et 877 non annotés avaient été remis aux participants (Uzuner *et al.*, 2011). Cependant, une partie de ce corpus, fournie par *Partners HealthCare* et *Beth Israel Deaconess Medical Center*, n'est plus disponible. Nous avons donc utilisé une sous-partie de ce corpus : 170 rapports annotés pour le corpus d'entraînement et 256 pour le test, ainsi que 267 non annotés (392k mots, dont 34k mots uniques).

Cette édition du challenge proposait 3 tâches autour de l'extraction de concepts médicaux des corpus : i) classification des concepts extraits, ii) des assertions sur les concepts extraits et iii) des relations

entre les concepts extraits. Nous nous intéresserons à la tâche 1 dont voici un extrait de la référence :

```
c="concept text" offset||t="concept label"
c="prostate cancer" 48:3 48:4||t="problem"
c="chest x-ray" 155:1 155:2||t="test"
c="chemotherapy" 32:4 32:4||t="treatment"
```

Une partie du corpus de l'édition 2012 (1k documents) du i2b2/VA Natural Language Processing Challenges for Clinical Records a été rajoutée lors de l'apprentissage des représentations distribuées, dans le but de les enrichir et d'obtenir des représentations plus significatives.

L'utilisation des abréviations et acronymes médicaux ainsi que les fautes de saisie sont difficiles à corriger ou à normaliser sans avoir des connaissances préalables du domaine, ce qui rend difficile le prétraitement des rapports médicaux. Un premier nettoyage à été réalisé avant le passage des documents dans l'extracteur terminologique YaTea (Aubin & Hamon, 2006) pour la sélection des termes caractéristiques du domaine, notamment la normalisation de certains mots très fréquents et écrits sous diverses formes (table 2).

Terme	Dr.	M.d.	w/	w/o	w/out	or.
Normalisation	Doctor	Doctor	with	without	without	operation room

TABLE 2: Quelques exemples de normalisations effectuées

Les documents ont ensuite été segmentés en phrases, puis un étiquettage morpho-syntaxique a été effectué avec l'outil TreeTagger, suivi par la lemmatisation et l'extraction des termes simple ou complexes significatifs du document à l'aide de YaTea (la table 3 une comparaison entre concepts proposés par l'évaluation et concepts extraits avec l'outil YaTea). Ces termes seront ensuite utilisés lors de la construction de la matrice de contextes. Les mots vides (prépositions, articles, pronoms, etc.) ont été écartés pour ne conserver que les mots pleins et les nombres (tous normalisés à 0).

i2b2 (2010)	YaTea	i2b2 (2010) $\cap$ YaTea
19410	11806	1941

TABLE 3: Concepts proposés par l'évaluation et concepts extraits avec l'outil YaTea

3.2 Extration des relations entre les termes à l'aide de patrons lexicaux

Pour un terme donné, on peut tirer certaines de ses relations classiques d'une ressource terminologique. Ceci étant dit, construire des bases de relations manuellement est coûteux et à plus forte raison pour les domaines de spécialité, car cela demande une connaissance préalable du domaine. Les patrons lexicaux ou lexicaux-syntaxiques exploitent une analyse syntaxique préalable du corpus pour constituer automatiquement un réseau lexical sur la base de relations définies bien connues et caractéristiques de la langue ou du domaine.

Nous utilisons donc ici des patrons lexicaux pour la détection de l'inclusion lexicale. Les termes analysés sont les termes générés par YaTea (table 4).

3.3 Expériences et résultats

La première partie du travail a consisté en la construction de nos vecteurs de contexte en utilisant les méthodes PPMI et SVD présentées plus haut. Cette combinaison permet à la fois de pondérer les

Terme complet	community hospital emergency room	chest pressure	blood pressure	chest x-ray
Terme tronqué	hospital emergency room	pressure	pressure	x-ray

TABLE 4: Exemple de sortie du système d'extraction d'inclusion lexicale

vecteurs de contextes et de réduire la dimensionnalité des matrices de co-occurrences (Jurafsky & Martin, 2018). Nous procédons ensuite à l'application de l'inclusion lexicale sur notre corpus : Les termes complexes dans le texte sont remplacés par leurs têtes dans nos contextes grâce aux relations extraites du corpus. Nous comparons ainsi :

— plusieurs dimensions différentes : sans réduction de dimension, puis réduction à 50, 200, 300 dimensions
— corpus avec et sans abstraction de contextes par inclusion lexicale

La seconde partie du travail a consisté en la construction de nos vecteurs de contexte avec des méthodes neuronales Wordvec et FastText. Initialement prévu pour des corpus volumineux, nous avons testé plusieurs paramètres afin d'observer le comportement de nos petits corpus dans cet espace. L'inclusion lexicale est moins triviale sur les modèles neuronaux. En effet, on ne peut pas directement agir sur les contextes et appliquer sur ceux-ci l'inclusion lexicale. Pour contourner cette difficulté, nous avons tout simplement ajouté au corpus de base les phrases sur lesquelles l'inclusion lexicale a été effectuée. Par exemple, si l'on considère le corpus suivant : *"He was admitted to the **community hospital emergency room** where he got a **chest x-ray**"*, notre corpus devient : *"He was admitted to the **community hospital emergency room** where he got a **chest x-ray**. He was admitted to the **hospital emergency room** where he got a **x-ray**".*

Nous comparons ainsi :

— deux modèles d'embeddings différents (Word2Vec et FastText)
— deux architectures différentes : Skip-Gram et CBOW
— plusieurs dimensions différentes : 50, 200, 300
— corpus avec et sans abstraction de contextes par inclusion lexicale

Pour les deux approches (sac de mots et neuronale), nous utilisons comme autres paramètres une fenêtre restreinte de 5 mots et retirons du vocabulaire tout terme apparaissant moins de 4 fois.

3.4 Qualité des représentations obtenues

À l'issue de la construction des vecteurs de contexte et une fois la similarité entre les différents termes calculée, une liste de termes considérés par les modèles comme sémantiquement proches est alors générée. Plusieurs types de relations existent entre ces termes dont des *relations classiques* (Périnet, 2015) : l'hyperonymie (relation entre deux termes du plus général au plus spécifique : *organe/coeur*), la méronymie (un terme désigne une partie d'un second terme, comme : *bras/corps*), la synonymie (des termes similaire : *myocarde/muscle cardiaque* ou ayant un sens proche comme *triste/déprimé*), l'antonymie (des termes contraires : *présent/absent*, *chaud/froid*) ainsi que des liens morphologiques (des relations au niveau de la forme des mots : *changeable/buvable, opéraient/administraient, etc...*). On y trouve aussi des *relations non classiques*, telles que les relations propres au domaine ("*don d'organe*", "*transporteur*"), l'hyponymie ou co-hyponymie (relation entre deux termes du plus spécifique au plus général (*coeur/organe* ou deux termes ayant le même hyponyme *coeur/foie*), des relations nom-verbes (*"infection/se propager"*), des collocations (*"entraîner des conséquences"*), etc.

Pour examiner les représentations que nous avons obtenues, nous comparons un terme sélectionné *"drug abuse"* et ses 7 voisins les plus proches pour chacun des modèles.

PPMI et SVD Comme on peut le voir dans la table 5, les voisins du terme *"drug abuse"* sont assez pertinents : plusieurs relations propres au domaine avec *"prostitution"*, *"recreational"*, *"ivdu"* (*"Intravenous Drug Use"*), l'hyperonyme *"abuse"*, des co-hyponymes avec *"tobacco abuse"* et *"alcohol abuse"* voire même une certaine notion d'antonymie avec *"non-smoker"*.

Modèle	Plus proches voisins						
$PPMI_{\alpha}SVD - Full$	prostitution	smoke history	recrudescence	non-compliance	tobacco abuse	prisoner	tobacco
$PPMI_{\alpha}SVD - 50$	recreational	abuse	domestic	non-smoker	violence	ethanol	tobacco abuse
$PPMI_{\alpha}SVD - 200$	recreational	ethanol	prostitution	0-to-0-pack-year	etoh	abuse	tobacco
$PPMI_{\alpha}SVD - 300$	recreational	prostitution	ethanol	alcohol	0-to-0-pack-year	tobacco	alcohol abuse
$PPMI_{\alpha}SVD_{IL} - Full$	prostitution	smoke history	recrudescence	prisoner	non-compliance	alcohol abuse	aya ma den erinmarg hospital
$PPMI_{\alpha}SVD_{IL} - 50$	recreational	abuse	nonsensitization	non-smoker	ethanol	ivdu	0-to-0-pack-year
$PPMI_{\alpha}SVD_{IL} - 200$	ivdu	recreational	ethanol	prostitution	etoh	ethanol	alcohol
$PPMI_{\alpha}SVD_{IL} - 300$	prostitution	recreational	ivdu	ethanol	alcohol	illicit	etoh

TABLE 5: Mots les plus proches de "*drug abuse*", classés par ordre de proximité, en utilisant la PPMI, puis en combinant PPMI et SVD

Embeddings neuronaux Nous observons dans la table 6 deux tendances principales. Avec Word2Vec, et à plus forte raison avec CBOW, les termes les plus proches du mot cible sont principalement des termes monolexicaux, présentant des relations propres au domaine, comme *"crack"*, *"alcoholism"*, mophologiquement complexes, comme avec *"polysubstance"*. On retrouve aussi l'hyperonyme *"abuse"* et les co-hyponymes *"tobacco abuse"* et *"alcohol abuse"*. Deux liens intéressants, celui de l'overdose suicidaire, est établi entre le mot cible *"drug abuse"* et son voisin *"suicide attempt"*, et celui qui indique la portée récréative des drogues avec *"recreational"* qui apparaît à quelques reprises dans les voisins de notre mot cible.

Avec FastText, les termes sont plus souvent polylexicaux, notamment avec des termes ayant un fort lien morphologique avec notre mot cible, notamment des synonymes, *"substance abuse"* et *"polysubstance abuse"*, des co-hyponymes *"cocaine abuse"*, *"tobacco abuse"* et *"alcohol abuse"*.

Deux liens intéressants ici également, un qui touche à un aspect légal de la consommation de drogue *"illicit"* et ce lien qui lie parfois toxicomanie et SIDA avec *"hiv"*.

Pour l'ensemble des modèles et d'un point de vue purement qualitatif, le voisinage post-inclusion lexical permet principalement l'acquisition de co-hyponymes et de relations propres au domaine.

3.5 Classification des concepts médicaux

L'extraction terminologique terminée, nos vecteurs de mots prêts, nous disposons alors de notre ensemble pour la classification de nos concepts. 3 labels sont donnés par la campagne i2b2/VA 2010 pour cette tâche : *problem, test, treatment*. L'algorithme utilisé est le classifieur SVM. Les SVM sont efficaces dans des espaces multidimensionnels ainsi que sur des corpus de petites tailles. Nous pouvons observer en table 7 que les résultats de la classification pré et post inclusion lexicale avec l'approche en sacs de mots, pondérée avec une PPMI et avec et sans réduction de dimensionnalité. Les résultats nous montrent des résultats quasi égaux, avant ou après inclusion lexicale.

Pour les méthodes neuronales, la table 8 montre les résultats de cette classification avant inclusion lexicale. FastText dépasse à peine Word2Vec avec l'approche Skip-Gram. Les performances de l'approche CBOW sont assez clairement inférieures.

La table 9 montre que les meilleurs résultats après inclusion lexicale améliorent à peine les résultats que ce soit pour l'approche CBOW ou Skip-Gram (3 *runs* ont été effectués avec des résultats variables en moyenne de l'ordre de ± 0.1).

Modèle	Plus proches voisins						
$W2V-50-sg$	abuse	0-pack	tobacco abuse	suicide attempt	crack	widowed	alcoholism
$W2V-200-sg$	0-pack	abuse	suicide attempt	alcoholism	reflux disease	tobacco history	cig
$W2V-300-sg$	abuse	suicide attempt	0-pack-year	non-smoker	crack	reflux disease	tobacco history
$W2V_{IL}-50-sg$	recreational	suicide attempt	socially	polysubstance	crack	user	reflux disease
$W2V_{IL}-200-sg$	suicide attempt	crack	ex	user	bronchitis	recreational	tobacco history
$W2V_{IL}-300-sg$	recreational	suicide attempt	socially	ingestion	0-pack-year	alcohol history	tob
$W2V-50-cbow$	cigarette	depression	etoh	alcohol abuse	0-pack	migraine	alcohol
$W2V-200-cbow$	radiation exposure	colonic	depression	tb	abuse	sexually	social
$W2V-300-cbow$	tobacco	cigarette	alcoholism	social	alcohol	smoking	0-pack
$W2V_{IL}-50-cbow$	drug	polysubstance	etoh	seizure	abuse	suicide	illicit
$W2V_{IL}-200-cbow$	drug	etoh	illicit	contrast allergy	seizure	recreational	tobacco history
$W2V_{IL}-300-cbow$	contrast allergy	reflux disease	counselor	anxiety	alcohol history	ketoacidosis	alcohol abuse
$FT-50-sg$	tobacco abuse	abuse	alcohol abuse	illicit	tobacco user	ethanol abuse	cocaine abuse
$FT-200-sg$	abuse	tobacco	drug	illicit	etoh	alcohol	alcoholism
$FT-300-sg$	abuse	tobacco	drug	alcohol	illicit	etoh	smoker
$FT_{IL}-50-sg$	abuse	ethanol abuse	tobacco abuse	cocaine abuse	alcohol abuse	illicit	tobacco user
$FT_{IL}-200-sg$	abuse	ethanol abuse	tobacco abuse	cocaine abuse	alcohol abuse	polysubstance abuse	illicit
$FT_{IL}-300-sg$	abuse	tobacco abuse	ethanol abuse	cocaine abuse	illicit	alcohol abuse	tobacco user
$FT-50-cbow$	substance abuse	cocaine abuse	tobacco abuse	ethanol abuse	passage	quincy	hhnk
$FT-200-cbow$	cocaine abuse	tobacco abuse	ethanol abuse	pan	dyspesia	message	substance abuse
$FT-300-cbow$	substance abuse	page	tobacco abuse	onset dyspnea	passage	cocaine abuse	dyspepsia
$FT_{IL}-50-cbow$	ethanol abuse	tobacco abuse	cocaine abuse	abuse	alcohol abuse	tobacco	hiv
$FT_{IL}-200-cbow$	ethanol abuse	tobacco abuse	cocaine abuse	tobacco	etoh	dlco	tobacco user
$FT_{IL}-300-cbow$	ethanol abuse	tobacco abuse	cocaine abuse	abuse	hiv	hillpa	pamela

TABLE 6: Mots les plus proches de "*drug abuse*", classés par ordre de proximité, avec différentes configurations d'embeddings neuronaux

Modèle	$PPMI_\alpha$							
Spécificité	Sans IL				Avec IL			
Dimensions	Sans SVD	50	200	300	Sans SVD	50	200	300
Précision	0,75	0,69	0,64	0,66	**0,76**	0,70	0,66	0,68
Recall	**0,80**	0,72	0,66	0,67	**0,80**	0,72	0,67	0,68
F-mesure	0,76	0,69	0,62	0,66	**0,77**	0,71	0,66	0,68

TABLE 7: Résultat de classification des modèles obtenus en appliquant la $PPMI_\alpha$ sans SVD et avec SVD avec une dimension de 50, 200, 300, avec et sans abstraction des contextes et avec macro mesure

Modèle	Word2Vec						FastText					
Architecture	Skip-Gram			CBOW			Skip-Gram			CBOW		
Dimensions	50	200	300	50	200	300	50	200	300	50	200	300
Précision	0,79	0,77	0,79	0,73	0,73	0,76	**0,8**	0,79	0,79	0,76	0,77	0,77
Recall	0,78	0,78	0,79	0,72	0,73	0,77	**0,8**	0,78	0,78	0,75	0,76	0,76
F-mesure	0,79	0,78	0,79	0,72	0,73	0,76	**0,8**	0,79	0,78	0,76	0,76	0,77

TABLE 8: Résultat de classification pour les modèles Word2Vec et FastText sans inclusion lexicale, mais avec dédoublement du corpus avec fenêtre de 5 et avec macro mesure

Modèle	Word2Vec_IL						FastText_IL					
Architecture	Skip-Gram			CBOW			Skip-Gram			CBOW		
Dimensions	50	200	300	50	200	300	50	200	300	50	200	300
Précision	**0,79**	**0,79**	**0,79**	**0,79**	0,78	0,78	**0,79**	**0,79**	**0,79**	0,78	0,77	0,77
Recall	0,79	0,78	0,79	0,79	0,78	0,78	**0,8**	0,78	0,78	0,79	0,78	0,77
F-mesure	**0,79**	**0,79**	**0,79**	**0,79**	0,78	0,78	**0,79**	**0,79**	0,78	0,78	0,77	0,77

TABLE 9: Résultat de classification pour les modèles Word2Vec et FastText avec inclusion lexicale, avec fenêtre de 5 et avec macro mesure

4 Conclusion et perspectives

Les méthodes distributionnelles se sont largement généralisées avec la multiplication des données textuelles disponibles. Construire des bases de relations manuellement est très coûteux et les méthodes distributionnelles permettent de faire reposer le calcul du sens sur l'analyse statistique des contextes des mots dans un corpus sans avoir besoin de recourir à des sources extérieures de connaissance. Les approches neuronales sont venues renouveler les performances de ces méthodes.

Dans cet article, nous avons présenté les méthodes classiques et neuronales les plus communes pour la mise en oeuvre des modèles de sémantique distributionnelle. Nous avons listé les principales raisons pour lesquelles ses méthodes n'étaient pas adaptées corpus de spécialité et qu'il fallait les adapter pour qu'ils prennent en compte les particularités structurelles et linguistiques de ces derniers. Nous avons énoncé le problème de dispersion des données ainsi que d'importance des termes, notamment complexes, dans l'étude des textes en langue de spécialité. Nous avons ensuite présenté des ébauches de travaux sur lesquels nous avons commencé à nous pencher à savoir l'étude de l'impact de la généralisation des contextes, notamment ici de l'inclusion lexicale, sur la sélection de contextes significatifs pour nos mots cibles.

Nous avons montré que d'un point de vue qualitatif, les relations obtenus entre les unités terminologiques propre au domaine était plus homogènes et que cela augmentait très légèrement les résultats de classification. Globalement, les résultats obtenus par les modèles à base de sac de mots et ceux obtenus par approche neuronale avec l'architecture Skip-Gram ne diffèrent pas de façon significative. Cependant, l'inclusion harmonise et hisse légèrement les résultats des méthodes neuronales avec l'architecture CBOW au niveau des autres. Notre tâche consistant en l'amélioration endogène des contextes distributionnels en domaine de spécialité, nous avons pour le moment écarté tout recours vers l'utilisation de corpus, de ressources ou de modèles pré-entraînés (e.g. *ELMo* et *BERT*) venant enrichir nos modèles distributionnels.

Ces expériences préliminaires ouvrent des nombreuses perspectives. Premièrement, prendre en compte un plus grand nombre de relations (synonymie, etc.) et ne pas se contenter uniquement de remplacer un terme par sa tête (inclusion lexicale, hypéronymie). Deuxièmement, déterminer une manière de sélectionner les contextes les plus significatifs en amont de l'inclusion lexicale (avec par, exemple, la mesure du $Cf \times Itf$ proposée par Périnet (2015)). Troisièmement, nous avons vu qu'appliquer l'abstraction des contextes directement sur les outils Word2Vec et FastText n'étaient pas trivial et nous avons contourné le problème en gardant à la fois la phrase originale et celle sur laquelle a été effectuée une inclusion lexicale. Quatrièmement, il serait intéressant d'explorer d'autres façons d'acquérir les unités terminologiques propres au domaine sur lesquelles nous basons notre travail. Finalement, l'évaluation des représentations étant une tâche très ardue, il serait intéressant d'évaluer nos représentations sur une autre tâche comme l'alignement lexical bilingue, par exemple.

Remerciements

Ce travail a été réalisé dans le cadre du projet ANR ADDICTE (Analyse Distributionnelle en Domaine de Spécialité) soutenu par l'Agence Nationale de Recherche sous la référence ANR-17-CE23-0001.

Références

AUBIN S. & HAMON T. (2006). Improving term extraction with terminological resources. In T. SALAKOSKI, F. GINTER, S. PYYSALO & T. PAHIKKALA, Eds., *Advances in Natural Language Processing (5th International Conference on NLP, FinTAL 2006)*, number 4139 in LNAI, p. 380–387 : Springer.

BENGIO Y., DUCHARME R., VINCENT P. & JANVIN C. (2003). A neural probabilistic language model. *J. Mach. Learn. Res.*, **3**, 1137–1155.

BOJANOWSKI P., GRAVE E., JOULIN A. & MIKOLOV T. (2016). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, **abs/1607.04606**.

CHURCH K. W. & HANKS P. (1990). Word association norms, mutual information, and lexicography. *Comput. Linguist.*, **16**(1), 22–29.

COLLOBERT R. & WESTON J. (2008). A unified architecture for natural language processing : Deep neural networks with multitask learning. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, p. 160–167, New York, NY, USA : ACM.

DEERWESTER S., T. DUMAIS S., FURNAS G., LANDAUER T. & HARSHMAN R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, **41**, 391–407.

DEVLIN J., CHANG M., LEE K. & TOUTANOVA K. (2018). BERT : pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv :1810.04805*, **abs/1810.04805**.

FABRE C., HATHOUT N., SAJOUS F. & TANGUY L. (2014). Ajuster l'analyse distributionnelle à un corpus spécialisé de petite taille. In *21e Conférence sur le Traitement Automatique des Langues Naturelles (TALN 2014)*, p. 266–279, Marseille, France.

GRABAR N. & ZWEIGENBAUM P. (2004). Lexically-based terminology structuring. *Terminology*, **10**(1), 23–54.

HAZEM A. & DAILLE B. (2018). Word embedding approach for synonym extraction of multi-word terms. In *Proceedings of the 11th Language Resources and Evaluation Conference*, Miyazaki, Japan : European Language Resource Association.

JURAFSKY D. & MARTIN J. H. (2018). *Speech and Language Processing : An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Upper Saddle River, NJ, USA : Prentice Hall PTR, 1st edition.

LEVY O., GOLDBERG Y. & DAGAN I. (2015). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, **3**, 211–225.

MANNING C. D., RAGHAVAN P. & SCHÜTZE H. (2008). *Introduction to Information Retrieval*. Cambridge, UK : Cambridge University Press.

MIKOLOV T., CHEN K., CORRADO G. & DEAN J. (2013). Efficient estimation of word representations in vector space. *CoRR*, **abs/1301.3781**.

MITCHELL J. & LAPATA M. (2010). Composition in distributional models of semantics. *Cognitive Science*, **34**(8), 1388–1429.

MORIN E. (2007). Apport des termes complexes à l'acquisition lexicale multilingue à partir de corpus comparables spécialisés : entre intuition et réalité. In *7ème rencontres Terminologies et Intelligence Artificielle (TIA'07)*, p. 11–20, Sophia Antipolis, France.

PÉRINET A. (2015). *Distributional analysis applied to specialized corpora : reduction of data sparsity through context abstraction*. Theses, Université Paris 13 ; Laboratoire d'Informatique Médicale et d'Ingénieurie des Connaissances en e-Santé.

PETERS M. E., NEUMANN M., IYYER M., GARDNER M., CLARK C., LEE K. & ZETTLEMOYER L. (2018). Deep contextualized word representations. In M. A. WALKER, H. JI & A. STENT, Eds., *NAACL-HLT*, p. 2227–2237 : Association for Computational Linguistics.

UZUNER O., SOUTH B., SHEN S. & DUVALL S. (2011). 2010 i2b2/va challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association : JAMIA*, **18**, 552–6.

ZWEIGENBAUM P. & HABERT B. (2006). Faire se rencontrer les parallèles : regards croisés sur l'acquisition lexicale monolingue et multilingue. In *Glottopol : Traitements automatisés des corpus spécialisés : contextes et sens*.

Automatic summarization of medical conversations, a review

Jessica López Espejel [1, 2]
(1) CEA, LIST, DIASI, F-91191 Gif-sur-Yvette, France.
(2) Paris 13 University, LIPN, 93430 Villateneuse, France.
jessica.lopez-espejel@cea.fr

RÉSUMÉ

L'analyse de la conversation joue un rôle important dans le développement d'appareils de simulation pour la formation des professionnels de la santé (médecins, infirmières). Notre objectif est de développer une méthode de synthèse automatique originale pour les conversations médicales entre un patient et un professionnel de la santé, basée sur les avancées récentes en matière de synthèse à l'aide de réseaux de neurones convolutionnels et récurrents. La méthode proposée doit être adaptée aux problèmes spécifiques liés à la synthèse des dialogues. Cet article présente une revue des différentes méthodes pour les résumés par extraction et par abstraction et pour l'analyse du dialogue. Nous décrivons aussi les utilisation du Traitement Automatique des Langues dans le domaine médical.

ABSTRACT

Conversational analysis plays an important role in the development of simulation devices for the training of health professionals (doctors, nurses). Our goal is to develop an original automatic synthesis method for medical conversations between a patient and a healthcare professional, based on recent advances in summarization using convolutional and recurrent neural networks. The proposed method must be adapted to the specific problems related to the synthesis of dialogues. This article presents a review of the different methods for extractive and abstractive summarization, and for dialogue analysis. We also describe the use of Natural Language Processing in the medical field.

MOTS-CLÉS : Résumé automatique, domaine médical, dialogues, revue.

KEYWORDS: Automatic summarization, medical domain, dialogues, review.

1 Introduction

Spoken medical dialogues are common and useful among doctors, nurses and patients. Automatically gathering the information of these dialogues is relevant for the medical practitioners and patients. For instance, the doctor can generate a medical history without omitting information, and a patient can review the diagnostic and the treatment later in time. However, none of them want a full record or transcript all dialogues. Besides, doctors and nurses have many interactions with different patients all day long, so it wouldn't be productive that they now have to deal with all these records. Accordingly, a challenge for Natural Language Processing (NLP) arises : obtaining automatic summaries from medical conversations.

According to the above, our final goal is to develop an automatic summarization method for medical

conversations between patients and doctors or nurses. However, our work is related to a project in our laboratory implementing a system for conversations between a virtual patient (chatbot) and medicine students (Laleye *et al.*, 2019). In this serious game, the student establishes a conversation with the chatbot to establish its diagnostic about the medical condition of the patient. The proposed method should be adapted to the specific problems of summarizing dialogues based on recent methods used in automatic summarization, such as recurrent neural networks. The hypothesis is that pertinent segments of dialogue might be detected and abstracted for their inclusion in a summary through deep learning. We suppose that, by identifying pertinent blocks of dialogues, it would be easier to detect important clinical issues and discard issues fragments like greetings, acknowledgments, social concerns, etc.

In this paper we review the work related to our goal of summarizing medical conversations. We first describe the two main types of summaries : extractive and abstractive. Extractive methods consist of copying chunks, usually sentences, from the original text. Abstractive methods are able to generate new words and sentences. In the literature there are more extractive approaches since they were easier than abstractive methods, even if the latter are becoming conceivable (See *et al.*, 2017). This paper describes different techniques for each one. Besides, it also reviews the specific mechanisms used in research on dialogue summarization.

The last section is crucial because it describes different approaches and works in NLP in the medical domain. These approaches include medical conversation systems and summarization in the medical domain. We discuss the challenges they represent.

2 Methods for summarization

In this section we will mention different types of approaches that researchers have been working along the time to get automatic summaries.

2.1 Summarization criteria

In the following lines we will describe two important criteria to summarize text. The first criteria is based on the frequency of the words. The second criteria are focused on the text features to detect the importance of the sentences.

Frequency criteria Over the years, several criteria have been developed to generate extractive summaries. One of the most cited in the literature is TF-IDF. TF (Term-Frequency) was proposed by Luhn (1958) and is the frequency of a word in the document. IDF (Inverse Document Frequency) was proposed by Sparck Jones (1972) and attenuates the weight of words that appear in a lot of the documents of the collection and increases the weight of words that occur in a few of them. The first works in summarization were based on $TF - IDF$. For instance, Wang & Cardie (2011) used unsupervised methods like TF-IDF, LDA, topic modeling and supervised clustering to produce a concise abstract of the decisions taken in spoken meetings.

Surface features criteria An alternative to detect the relevance of a sentence is through features of different kind. Yogan *et al.* (2016) mentioned the following features : title/headline words, sentence position, sentence length, proper noun and term weight. Chopra *et al.* (2016) captured the word position in which it occurs in the sentence and its context in which appears in the sentence. Lacson *et al.* (2006) used the length of dialogue turns to detect important information in a conversation. Nallapati *et al.* (2016) combined features to get better results.

2.2 Summarization methods

Probabilistic Models There are also probabilistic models, such as Context Free Grammars and Markov Models (MM). Probabilistic Context Free Grammars (PCFG) is a probabilistic model of syntax for tree structures. Rahman *et al.* (2001) worked on automatic summarization of Web pages and used PCFG to define syntactic structures, analyze and understand content. Besides, Knight & Marcu (2002) worked on sentence compression and developed a probabilistic noisy-channel model that used PCFG to assign probabilities to a tree. On the other hand, Chen & Withgott (1992) used Markov Model(MM) on speech summarizing. Jing & McKeown (1999) proposed an algorithm based on Hidden Markov Model(HMM) that decomposes human-written summary sentences to determine the relations between the sentences in a summary written by humans and sentences in the original text. Conroy & O'leary (2001) proposed a method for text summarization, which considers three features : the position of the sentence in the document (using Hidden Markov Model), the number of terms in the sentence and the probabilities of the terms.

Optimization methods On the other hand, other important approach to get summaries from text is based on Integer Linear Program (ILP). Gillick & Favre (2009) used ILP to exact inference under a maximum coverage model in automatic summarization and sentence compression. Mnasri *et al.* (2017) used the same methods for multidocument update summarization, improving it by taking into semantic similarity and document structure.

Graph-based methods Mihalcea (2004) used graph-based ranking algorithms to extract the most important sentences from DUC (2002). Unlike Mihalcea (2004), Litvak & Last (2008) used graphs to identify keywords to be used in extractive summarization of text documents.

Machine Learning Approaches In recent years, researchers have proposed methods based-on machine learning to summarize text. Naive Bayes was used by Kupiec *et al.* (1995) to chose if a sentence belongs or not to a summary. Recently Ramanujam & Kaliappan (2016) extended the application of the Naive Bayes algorithm to automatic summarization in multi-documents. Aside from Nayve Bayes, clustering algorithms have been used by Aliguliyev (2009) and KM & Soumya (2015) to get extractive summaries. Aliguliyev (2009) proposed a method based on sentence clustering, while KM & Soumya (2015) prepared the cluster center using a n-dimensional vector space and the documents similarity is measure by cosine similarity to generate the documents clusters.

Besides, researchers have used Support Vector Machine (SVM). For example, (Schilder & Kondadadi, 2008) work on query-based multi-document summarization and they use SVM to rank all sentences in the topic cluster for summarization. Another algorithm that researchers have been used is genetic algorithms. For instance, Chatterjee *et al.* (2012) worked on extractive summaries representing the

document as a Direct Acyclic grapth (DAC). They used genetic algorithm to maximize the fitness function and get the summary. After few years, Bossard & Rodrigues (2017) used objective function from generic algorithms to explore summaries space and compute the probability distribution of tokens.

One of the most widely used family of machine learning methods for automatic summarization is Neural Networks (NN). In NLP, the currently most relevant recurrent neural networks are : LSTM (Long Short-Term Memory) and GRU (Gated Recurrent Unit). LSTM has the ability to add or remove information through structures called gates : forget gate, input gate, candidate gate and output gate layer. Meanwhile, GRU is a variant of LSTM and it combines the forget and input gates into a single *update gate*. LSTM and GRU present some advantages, such as the ability to store long-term dependencies, they avoid the problem of vanishing gradient and they consider the order of the words. In abstractive text summarization, taking into account the order of words is one of the greatest advances because the summaries present more coherence.

Kaikhah (2004) used NN to learn relevant features of sentences and decide if these sentences should be included in the summary, while Sinha *et al.* (2018) proposed a model based on feed-forward networks for single document summarization. LSTM was used by Cheng & Lapata (2016) and Zhou (2016). Both used an encoder-decoder model. However Cheng & Lapata (2016) worked on extractive summarization of single documents, while Zhou (2016) applied a hierarchical LSTM model for building the sentence representations in abstractive and long summaries. GRUs were used by Nallapati *et al.* (2017) and more recently by Li *et al.* (2018). Nallapati *et al.* (2017) proposed SummaRuNNer (simple recurrent network based sequence classifier) for extractive summarization. Li *et al.* (2018) extended the basic encoder-decoder model. They added an intermediate layer (select layer) that consists in two parts : gated global information filtering and local sentence selection.

Over the past few years, the number of works in abstractive automatic summarization has increased. One of the first NN approaches to tackle abstractive summarization was the Attention RNN (Recurrent NN). Encoder-Decoder Model which was introduced by Bahdanau *et al.* (2015). Rush *et al.* (2015) combine this approach with a generation algorithm for a fully data-driven approach to abstractive sentence summarization. One extension of this work for the same problem was developed by Chopra *et al.* (2016) but they didn't use feed-forward neural language model for generation like Rush *et al.* (2015). They used a convolutional attention-based conditional recurrent neural network model, producing further improvement in performance on Gigaword and DUC datasets.

The previous mentioned works focused on short texts. However Nallapati *et al.* (2016) proposed a new dataset for the task of abstractive summarization of a document into multiples sentences and establish brenchmarks. Furthermore, See *et al.* (2017) used this dataset and proposed a hybrid architecture between sequence-to-sequence attention model and a pointer network Vinyals *et al.* (2015) that facilitates copying words from the source text via pointing or generating words from a fixed vocabulary. They also adapt the coverage model of Tu *et al.* (2016) to solve the problem of repetitions in the generated summary.

Automatic summarization methods have been applied on various kinds of documents, such as text (news, articles, etc), dialogues, in medical and other domains. In the literature, we can find automatic summarization of dialogues in meetings, recorded conversations in call centers and other events that happen everyday. The majority of works on medical summarization have focused on research articles. However in our case we are interested to get automatic summaries from medical conversations. In sections 3 and 4, we describe independently NLP works on dialogue analysis and the medical domain, with a focus on summarization.

3 Summarization methods applied on dialogues

A dialogue is a sequence of conversational turns between multiple participants, where each turn would modify each participant cognitive status and the current dialogue state Chih-Wen & Chen (2018). However, we suppose that two people are enough to establish a dialogue. Currently, a person can also establish a conversation with a system equipped with conversational intelligence Turing (1950), either in written or oral form.

The process of automatic dialogue summarization is a challenging summarization task because, e.g., unlike in journalistic texts, the most important sentence is not the first one in each paragraph, and we have to consider other aspects such as the speakers' turns. In the case of capturing the conversations orally, we face several types of disfluencies such as fillers, repetitions, repairs, or unfinished clauses Zechner & Waibel (2000). Otherwise, on conversations obtained from written sources, it might be necessary to pre-process the data due to lexical irregularities.

Over the past few years, researchers have worked on automatic summaries in different areas of daily life, such as mettings, phone calls and medical domain. For example, Hendrik Buist *et al.* (2004) worked on audio-visual meetings and their goal was to extract all important topics. In another related work, the goal was to produce a concise abstract of decisions taken in spoken meetings Wang & Cardie (2011).

On the other hand, automatic summary systems dedicated to call centers have also been developed to record calls between agents and clients, to know the content of the call and the agent's level of expertise. Tamura *et al.* (2011) proposed an extractive method and they introduced a component that removes frequent sentences from summary, in comparison with (Evgeny *et al.*, 2015) where they focus on abstractive summarization and used domain knowledge to fill hand-written templates from entities detected in the transcript of the conversation.

Additionally, the need to use systems of dialogue in the medical domain has increased in the last two decades Bickmore & Giorgino (2006) . The HOMEY system Piazza *et al.* (2004) was developed to enhance communication between health center specialists and patients with chronic diseases. After few years, Andrenucci (2008) used the automatic question-answering (QA) paradigm in medical domain. They determined that the best approaches for medical applications are deep analysis of language (semantics) and template based.

4 NLP in Medical Domain

In medical domain, doctors and nurses have a lot of information about patients : medical records, appointments, schedule of activities, etc. All information must be organized, and manually involves a lot of time and human effort. Besides, the amount of information increases each second and there is sometimes not enough people.

Currently, there are algorithms capable of helping nurses and doctors to summarize and organize information. However the medical domain is huge. Some works are focused on medical diagnostic tools, such as Doan *et al.* (2016). They detected child patients with high suspicion of Kawasaki Disease (KD) based on standard clinical terms and medical lexicon usage. Likewise, researchers have worked on the detection of depression, suicide risk Pestian *et al.* (2010) Mulholland & Quinn (2013) and mental diseases Thomas *et al.* (2005) Karlekar *et al.* (2018) by clinical notes and social media

Calvo *et al.* (2017) analysis using NLP techniques.

Other researches have developed intelligent agents, such as Allen *et al.* (2006) to help people at home. People can talk to the systems using natural language.

The following subsections concentrate on conversational and summarization systems in the medical domain.

4.1 Medical Conversational Systems

There are many conversational applications that have been developed to help the health field, such as intelligence agents and health dialog systems. Allen *et al.* (2006) developed Chester, a prototype of medical advisor. Chester provides information, advises based on prescribed medications and reports back to medical support team, it can answer questions asked by users. In the meantime, de Rosis *et al.* (2006) developed a conversational agent to talk to patients to influence them to change their dietary behavior.

Migneault *et al.* (2006) describe the approach of how to write dialogues for TLC (Telephone-Linked Care) telephone systems, whose objective is to offer health-related services. At the same time, (Beveridge & Fox, 2006) studied the automatic generation of dialogue combining knowledge of the structure of tasks and ontological knowledge, the objective is to decide if a patient must be referred or not with a cancer specialist.

4.2 Summarization in medical domain

Automatic summarization can be used for medical consultations. The dialogue between patients and doctors can be recorded and a summary been generated with the most important points, to the attention of other doctors and for the medical history of the patient. This is the consultation report.

To the best of our knowledge, there are few works about summarization on medical dialogues. One of them was developed by Lacson *et al.* (2006). They worked on automatic summarization of dialogues between nurses and dialysis patients. Their system consists in two main components : induction (machine-learning algorithm for classifying dialogue turns) and summarization (prediction based on a meaning representation encoded in lexical and contextual features). Unlike them, Sarkar *et al.* (2011) worked on automatic summarization of medical articles to decide if an article is useful or not. The algorithm classify the sentences as positive or negative. After ranking, sentences are selected to generate the final summary.

In addition, researchers have also developed NLP tools focused on medical domain that can help us to generate better quality of summaries through the identification of medical terms. For instance, Li & Wu (2006) implemented KIP (Keyphrase Identification Program) to identify topical concepts from medical documents. KIP combines noun phrase extraction and keyphrase identification. MedPost is a Part-Of-Speech(POS) tagger developed by Smith *et al.* (2004). MedPost is based on Hidden Markov Model. Additionally, Tanabe & Wilbur (2002) worked on extracting gene and proteins names from MEDLINE documents.

4.3 Challenges on medical domain

Clinical notes and medical texts present several challenges for NLP. One of the most important challenge is based on word disambiguation because extracting the meaning from unstructured text is not easy. For example, *cold* can refer to the weather or a disease Townsend (2013).

Besides, clinical texts are often ungrammatical. It means that they contain incomplete sentences and limited context. Also, along the time new terms and abbreviations emerge. A lot of this terms that appear in medical sources such as the Unified Medical Language System (UMLS) Metathesaurus have multiple meanings which depend on the context.

Otherwise, an important challenge on summarization area is to get summaries from medical dialogues. Even finding or building corpus of medical conversations is hard.

Finally, an important challenge in the medical domain is ethical. The building of systems must ensure that private sensitive data cannot leak. Nobody except authorized medical people must be able to access data related to a specific patient. This poses hard to solve problems of anonymisation. Besides, it is very important not to modify the meaning or to give incomplete information in the final summary.

5 Discussion

We have presented works about medical summarization divided in three sections : methods for summarization, NLP for dialogue analysis and medical conversation systems. In the first section, the most used methods on extractive and abstractive summaries have been reviewed. The second section presented works focused on dialogue summaries, and the challenge that they present. In the last section we cited works to help people at home, and software to detect depression, suicide and mental diseases.

After this review of the bibliography, we realized that there are a lot of articles about automatic summarization, but the majority are focused on scientific articles, news, etc. and there are few works on conversations. Furthermore, there is even less works on medical dialogue. One of the most important reason is that it is not easy to get datasets. Similarly, we find more articles on extractive summaries than abstractive summaries, even if the current tendency is clearly to switch to abstractive using neural networks.

Our future work is based on the model by See *et al.* (2017). We will apply this technique on the PubMed dataset (medical articles). We decided to work with this algorithm because they used a hybrid pointer-generator network to decide if a specific word is copied from the source text or if a new word from the vocabulary will be generated. Our hypothesis is that pointer-generator networks can be useful on the medical domain because we can generate new words from its specialized vocabulary. This will make easier to handle medical terms. Besides, they proposed a novel variant of coverage based on Tu *et al.* (2016) to avoid repetitions in summaries. As noted above, it will be very important to insure that the generated text doesn't modify the sense of the original dialogue. We will also make the algorithm able to handle long texts while it was designed only for the first words of a news article.

Our second approach is to work on the AMI corpus developed by Mccowan *et al.* (2005), a dataset of 100 hours of multi-modal meetings recordings. We will develop a mechanism to generate abstractive summaries from dialogues. We decided to work on AMI dataset to get automatic summaries from

real conversations and consider aspects that we can find only in dialogues such as greetings, speakers' turns, etc.

However, our main idea is to develop a mechanism to be able to get abstractive summaries from medical dialogues. To achieve this task, it's necessary to work with both approaches : summarization of dialogues and summarization in the medical domain.

Références

ALIGULIYEV R. M. (2009). A new sentence similarity measure and sentence based extractive technique for automatic text summarization. *Expert Systems with Applications*, **36(4)**, 7764–7772.

ALLEN J. F., FERGUSON G., BLAYLOCK N., BYRON D. K., CHAMBERS N., DZIKOVSKA M. O., GALESCU L. & SWIFT M. D. (2006). Chester : Towards a personal medication advisor. *Journal of Biomedical Informatics*, **39(5)**, 500–513.

ANDRENUCCI A. (2008). Automated question-answering techniques and the medical domain. In *HEALTHINF*.

BAHDANAU D., CHO K. & BENGIO Y. (2015). Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

BEVERIDGE M. & FOX J. (2006). Automatic generation of spoken dialogue from medical plans and ontologies. *Journal of Biomedical Informatics*, **39**(5), 482–499.

BICKMORE T. W. & GIORGINO T. (2006). Health dialog systems for patients and consumers. *Journal of biomedical informatics*, **39(5)**, 556–71.

BOSSARD A. & RODRIGUES C. (2017). An evolutionary algorithm for automatic summarization. In *Proceedings of Recent Advances in Natural Language Processings 2017*.

CALVO R., MILNE D. N., HUSSAIN M. S. & CHRISTENSEN H. (2017). Natural language processing in mental health applications using non-clinical texts. *Natural Language Engineering*, **23**, 649–685.

CHATTERJEE N., MITTAL A. & GOYAL S. (2012). Single document extractive text summarization using genetic algorithms. In *Third International Conference on Emerging Applications of Information Technology*, p. 19–23.

CHEN F. & WITHGOTT M. (1992). The use of emphasis to automatically summarize a spoken discourse. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, p. 229–232.

CHENG J. & LAPATA M. (2016). Neural summarization by extracting sentences and words. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, p. 484–494.

CHIH-WEN G. & CHEN Y.-N. (2018). Abstractive dialogue summarization with sentence-gated modeling optimized by dialogue acts. *CoRR*.

CHOPRA S., AULI M. & RUSH A. M. (2016). Abstractive sentence summarization with attentive recurrent neural networks. In *NAACL-HLT*, p. 93–98.

CONROY J. M. & O'LEARY D. P. (2001). Text summarization via hidden markov models. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, p. 406–407.

DE ROSIS F., NOVIELLI N., CAROFIGLIO V., CAVALLUZZI A. & DE CAROLIS B. (2006). User modeling and adaptation in health promotion dialogs with an animated character. *Journal of Biomedical Informatics*, **39(5)**, 514–531.

DOAN S., MAEHARA C. K., CHAPARRO J., LU S., LIU R., GRAHAM A., BERRY E., HSU C.-N., T. KANEGAYE J., D. LLOYD D., OHNO-MACHADO L., C. BURNS J. & TREMOULET A. (2016). Building a natural language processing tool to identify patients with high clinical suspicion for kawasaki disease from emergency department notes. *Academic Emergency Medicine*, **23**, 328–636.

DUC (2002). Documentunderstandingconference2002. urlhttps ://www-nlpir.nist.gov/projects/duc/.

EVGENY S., FAVRE B., FIROJ A., SHAMMUR C., SINGLA K., TRIONE J., B'ECHET F. & GIUSEPPE R. (2015). Automatic summarization of call-center conversations. In *Conference on Empirical Methods in Natural Language Processing*.

GILLICK D. & FAVRE B. (2009). A scalable global model for summarization. In *Proceedings of the Workshop on Integer Linear Programming for Natural Language Processing*, p. 10–18 : Association for Computational Linguistics.

HENDRIK BUIST A., KRAAIJ W. & RAAIJMAKER S. (2004). Automatic summarization of meeting data : A feasibility study. In *Proc. Meeting of Computational Linguistics in the Netherlands (CLIN)*.

JING H. & MCKEOWN K. R. (1999). The decomposition of human-written summary sentences. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, p. 129–136 : ACM.

KAIKHAH K. (2004). *Text Summarization using Neural Networks*. PhD thesis, Department of Computer Science, Texas State university.

KARLEKAR S., NIU T. & BANSAL M. (2018). Detecting linguistic characteristics of alzheimer's dementia by interpreting neural models. In *NAACL-HLT*, volume 2, p. 701–707.

KM S. & SOUMYA R. (2015). Text summarization using clustering technique and svm technique. *International Journal of Applied Engineering Research*, **10**, 25511–25519.

KNIGHT K. & MARCU D. (2002). Summarization beyond sentence extraction : A probabilistic approach to sentence compression. *Artificial Intelligence*, **139**, 91–107.

KUPIEC J., PEDERSEN J. & CHEN F. (1995). A trainable document summarizer. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, p. 68–73.

LACSON R. C., BARZILAY R. & LONG W. J. (2006). Automatic analysis of medical dialogue in the home hemodialysis domain : Structure induction and summarization. *Journal of Biomedical Informatics*, **39(5)**, 541–555.

LALEYE F. A. A., BLANIÉ A., BROUQUET A., BENHAMOU D. & DE CHALENDAR G. (2019). Hybridation d'un agent conversationnel avec des plongements lexicaux pour la formation au diagnostic médical. In *23ème Conférence Sur Le Traitement Automatique Des Langues Naturelles (TALN-RECITAL 2019)*, Toulouse, France.

LI Q. & WU Y.-F. (2006). Identifying important concepts from medical documents. *Journal of biomedical informatics*, **39**, 668–679.

LI W., XIAO X., LYU Y. & WANG Y. (2018). Improving neural abstractive document summarization with explicit information selection modeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

LITVAK M. & LAST M. (2008). Graph-based keyword extraction for single-document summarization. In *Proceedings of the Workshop on Multi-source Multilingual Information*, p. 17–24.

LUHN H. (1958). The automatic creation of literature abstracts. *IBM Journal of Research Development*, **2**, 159–165.

MCCOWAN I., LATHOUD G., LINCOLN M., LISOWSKA A., POST W., REIDSMA D. & WELLNER P. (2005). The ami meeting corpus. In *5th International Conference on Methods and Techniques in Behavioral Research.*

MIGNEAULT J. P., FARZANFAR R., WRIGHT J. A. & FRIEDMAN R. H. (2006). How to write health dialog for a talking computer. *Journal of Biomedical Informatics*, **39**, 468–481.

MIHALCEA R. (2004). Graph-based ranking algorithms for sentence extraction, applied to text summarization. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, p. 170–173.

MNASRI M., DE CHALENDAR G. & FERRET O. (2017). Taking into account inter-sentence similarity for update summarization. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2 : Short Papers)*, p. 204–209, Taipei, Taiwan : Asian Federation of Natural Language Processing.

MULHOLLAND M. & QUINN J. (2013). Suicidal tendencies : The automatic classification of suicidal and non-suicidal lyricists using nlp. In *6th International Joint Conference on Natural Language Processing*, p. 19–28.

NALLAPATI R., ZHAI F. & ZHOU B. (2017). Summarunner : A recurrent neural network based sequence model for extractive summarization of documents. In *AAAI*.

NALLAPATI R., ZHOU B., NOGUEIRA DOS SANTOS C., GÜLÇEHRE C. & XIANG B. (2016). Abstractive text summarization using sequence-to-sequence rnns and beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, p. 280–290.

PESTIAN J. P., NASRALLAH H. A., MATYKIEWICZ P., BENNETT A. & LEENAARS A. (2010). Suicide note classification using natural language processing : A content analysis. *Biomedical informatics insights*, **2010 (3)**, 19–28.

PIAZZA M., GIORGINO T., AZZINI I., STEFANELLI M. & LUO R. (2004). Cognitive human factors for telemedicine systems. *Studies in Health Technology and Informatics*, p. 974–978.

RAHMAN A., ALAM H., HARTONO R. & ARIYOSHI K. (2001). Automatic summarization of web content to smaller display devices. In *ICDAR Proceedings of Sixth International Conference on Document Analysis and Recognition*, p. 1064–1068.

RAMANUJAM N. & KALIAPPAN M. (2016). An automatic multidocument text summarization approach based on naïve bayesian classifier using timestamp strategy. In *The Scientific World Journal*.

RUSH A. M., CHOPRA S. & WESTON J. (2015). A neural attention model for sentence summarization. *Association for Computational Linguistics*, **Conference on Empirical Methods in Natural Language Processing**, 379–389.

SARKAR K., NASIPURI M. & GHOSE S. (2011). Using machine learning for medical document summarization. *International Journal of Database Theory and Application*, **4**.

SCHILDER F. & KONDADADI R. (2008). Fastsum : Fast and accurate query-based multi-document summarization. In *Association for Computational Linguistics*, p. 205–208.

SEE A., LIU P. J. & MANNING C. D. (2017). Get to the point : Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, p. 1073–1083.

SINHA A., YADAV A. & GAHLOT A. (2018). Extractive text summarization using neural networks. *CoRR*, **abs/1802.10137**.

SMITH L., RINDFLESCH T. & WILBUR W. (2004). Medpost : A part-of-speech tagger for biomedical text. *Bioinformatics (Oxford, England)*, **20**, 2320–1.

SPARCK JONES K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, **28**, 11–21.

TAMURA A., ISHIKAWA K., SAIKOU M. & TSUCHIDA M. (2011). Extractive summarization method for contact center dialogues based on call logs. *5th International Joint Conference on Natural Language Processing*, p. 8–13.

TANABE L. & WILBUR W. J. (2002). Tagging gene and protein names in full text articles. In *Proceedings of the Workshop on Natural Language Processing in the Biomedical Domain*, volume 20, p. 9–13.

THOMAS C., KESELJ V., CERCONE N., ROCKWOOD K. & ASP E. (2005). Automatic detection and rating of dementia of alzheimer type through lexical analysis of spontaneous speech. *IEEE International Conference Mechatronics and Automation, 2005*, **3**, 1569–1574.

TOWNSEND H. (2013). Natural language processing and clinical outcomes : the promise and progress of nlp for improved care. *Journal of AHIMA / American Health Information Management Association*, **84**, 44–5.

TU Z., LU Z., LIU Y., LIU X. & LI H. (2016). Modeling coverage for neural machine translation. In *ACL*.

TURING A. M. (1950). Computing machinery and intelligence. *Mind*, **49**, 433–460.

VINYALS O., FORTUNATO M. & JAITLY N. (2015). Pointer networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*.

WANG L. & CARDIE C. (2011). Summarizing decisions in spoken meetings. In *Proceedings of the Workshop on Automatic Summarization for Different Genres, Media, and Languages*, p. 16–24.

YOGAN J. K., GOH O. S., HALIZAH B., NGO H. C. & PUSPALATA C. S. (2016). A review on automatic text summarization approaches. **12**, 178–190.

ZECHNER K. & WAIBEL A. H. (2000). Diasumm : Flexible summarization of spontaneous dialogues in unrestricted domains. In *International Conference on Computational Linguistics*.

ZHOU Z. (2016). A hierarchical model for text autosummarization. `https://cs224d.stanford.edu/reports/zhenpeng.pdf`.

Détection automatique de chaînes de coréférence pour le français écrit: règles et ressources adaptées au repérage de phénomènes linguistiques spécifiques

Bruno Oberle[1]
(1) Linguistique, Langues, Parole (LiLPa, EA-1339), Université de Strasbourg, F-67084 Strasbourg, France
oberleb@unistra.fr

RÉSUMÉ

Nous présentons un système *end-to-end* de détection automatique des chaînes de coréférence, à base de règles, pour le français écrit. Ce système insiste sur la prise en compte de phénomènes linguistiques négligés par d'autres systèmes. Nous avons élaboré des ressources lexicales pour la résolution des anaphores infidèles (*Mon chat... Cet animal...*), notamment lorsqu'elles incluent une entité nommée (*La Seine... Ce fleuve...*). Nous utilisons également des règles pour le repérage de mentions de groupes (*Pierre et Paul*) et d'anaphores zéros (*Pierre boit et ø fume*), ainsi que des règles pour la détection des pronoms de première et deuxième personnes dans les citations (*Paul a dit : "**Je** suis étudiant."*). L'article présente l'élaboration des ressources et règles utilisées pour la gestion de ces phénomènes spécifiques, avant de décrire le système dans son ensemble, et notamment les différentes phases de la résolution de la coréférence.

ABSTRACT

Automatic coreference resolution for written French : rules and resources for specific linguistic phenomena

We introduce a new rule-based coreference resolution system for written French. This system takes into account linguistic phenomena often ignored by other systems. First, we have built lexical resources to improve full NP coreference resolution (*My cat... The animal*), especially when a named entity is involved (*The Seine... The river...*). We have defined rules to detect groups of individuals (*Peter and Paul*) and null anaphora (*Peter drinks and ø smoke*). We have also defined rules to detect first and second person pronouns in quotations (*Paul said : "**I** am a student"*). This paper first presents how we built our lexical resources and how we defined our rules, then it describes how our system works and specifically what are the steps to resolve coreference.

MOTS-CLÉS : expression référentielle, coréférence, détection automatique de la coréférence, entité nommée.

KEYWORDS: referring expression, coreference, automatic coreference detection, named entity.

1 Introduction

Une expression référentielle, ou *mention*, est un segment de texte qui renvoie à une entité extralinguistique : le référent. Lorsque le même référent est repris dans le même texte par une autre mention, une *relation de coréférence* (Corblin, 1995) s'établit entre les mentions, qui font alors partie d'une

même *chaîne de coréférence* (Chastain, 1975; Corblin, 1985; Charolles, 1988; Schnedecker, 1997). Ainsi, dans l'exemple suivant, les termes *Platon, Il, ses* et *il* renvoient tous au même référent :

(1) *Platon* est un philosophe antique de la Grèce classique... *Il* reprit le travail philosophique de certains de *ses* prédécesseurs, notamment Socrate dont *il* fut l'élève. (Wikipédia)

On considère généralement (Charolles, 1988; Schnedecker & Landragin, 2014) que les mentions sont des expressions nominales (noms et pronoms), mais également des déterminants possessifs puisqu'ils varient en personne (*mon vs. ton vs. son*, etc.) et en nombre (*mon vs. notre, son vs. leur*, etc.) en fonction du possesseur, et qu'ils comprennent une référence à une entité.

Un système de détection automatique des chaînes de coréférence assemblent les mentions (expressions référentielles) d'un texte en chaînes de coréférence. On dit qu'il est *end-to-end* (ou *de bout-en-bout*) lorsqu'il repère également les mentions, et qu'il accepte donc en entrée un texte brut, plutôt qu'un texte pré-annoté en mentions.

Plusieurs systèmes de ce type ont été créés pour le français (notre langue cible étant le français, nous n'évoquerons pas les systèmes anglais ; voir cependant (Poesio *et al.*, 2016), notamment la partie 2). Le système de Trouilleux (Trouilleux, 2001), qui ne s'occupe que de certaines anaphores pronominales, celui de Dupont (Dupont, 2003; Victorri, 2005) et RefGen (Longo, 2013) sont des systèmes symboliques, qui résolvent la coréférence à partir de règles prédéfinies. Leurs performances sont généralement moindres que les systèmes suivants, développés à partir du seul corpus francophone pour l'heure annoté en coréférences : ANCOR (Muzerelle *et al.*, 2013). Trois de ces systèmes font appel à l'apprentissage automatique : une adaptation de BART au français dans le cadre du projet SENSEI (Kabadjov & Stepanov, 2015), ainsi que CROC (Désoyer *et al.*, 2015, 2016) et le système décrit dans (Brassier *et al.*, 2018). Ces systèmes ne fonctionnent que sur des mentions gold, c'est-à-dire qu'ils ne détectent pas eux-mêmes les expressions référentielles, mais seulement la coréférence. Un quatrième système, décrit par (Godbert & Favre, 2017), qui se limite aux anaphores pronominales et aux anaphores fidèles (reprise d'un nom par le même nom, par exemple *"le chat... ce chat"*, ce que les concepteurs d'ANCOR appellent "relations directes"), utilise des règles. Tout récemment, des approches à partir des réseaux de neurones ont été développées (Grobol, 2019).

Le corpus ANCOR, cependant, est un corpus d'oral transcrit (conversations téléphoniques et entretiens) dont les spécificités et l'annotation ne correspondent pas toujours aux caractéristiques de l'écrit.

Notre objectif est d'élaborer, pour le français écrit, un système *end-to-end* de détection automatique de la coréférence, qui s'attache à repérer des phénomènes coréférentiels fréquents dans les textes écrits mais qui sont souvent négligés par les autres systèmes.

L'une des difficultés des systèmes de détection de la coréférence est la résolution des anaphores infidèles (reprise d'un nom par un synonyme, par exemple *Le chat... Cet animal...*), alors qu'il s'agit d'un phénomène très fréquent en français écrit, qui supporte mal la répétition. Notre intérêt s'est donc d'abord porté sur la résolution de ce type d'anaphores, notamment lorsqu'elles impliquent des entités nommées (*Le Rhin... Ce fleuve...*, ou *Paris... La Ville Lumière*).

Ensuite, nous nous sommes intéressé aux phénomènes coréférentiels dans lesquels la détection de la mention elle-même est difficile :

— lorsque le référent est un groupe qui englobe plusieurs autres mentions individuelles : ***Sophie et Marie*** *sortent ensemble.* ***Elles*** *sont très heureuses*. Dans cet exemple, c'est le groupe, qui a des caractéristiques propres (féminin, pluriel), qui est repris ;

— lorsque le référent est repris sans être exprimé (cas de “l’anaphore zéro”) : ***Paul*** *boit et* ***ø*** *fume*. Ici, pour être en mesure de comprendre que Paul fait deux actions (boire et fumer), il faut être en mesure de “distribuer” le sujet sur les deux verbes.

Enfin, il est fréquent de trouver dans la presse écrite des citations qui contiennent des pronoms et déterminants possessifs de première et deuxième personnes (*je, tu, nous, vous, mon, ton, notre, votre*). Pour comprendre qui, dans ces citations, exprime une opinion, il faut pouvoir associer le “je” et le “tu” à des référents dont les noms sont donnés dans le texte environnant la citation :

(2) *Lionel Jospin* se livre en revanche à une longue analyse de *son* échec du 21 avril. “*Ma* part de responsabilité dans l’échec existe forcément. *Je* l’ai assumée en quittant la vie politique”, explique-t-*il* d’emblée. (L’Est Républicain)

Ces cas complexes demandent des stratégies spécifiques. Pour la résolution des anaphores infidèles, nous avons construit deux ressources lexicales : l’une qui déduit des informations utiles pour la détection de la coréférence à partir d’entités nommées liées à une base de connaissances, l’autre qui est un dictionnaire d’hyperonymes.

Les autres phénomènes sont traités par des règles appliquées sur une analyse syntaxique faite par Talismane (Urieli, 2013).

La suite de cet article insiste d’abord sur la conception des ressources lexicales (section 2) que nous avons créées et des règles que nous avons établies (section 3), avant de décrire le fonctionnement général du système de détection de la coréférence ansi que son évaluation (section 4).

Le système que nous proposons est donc un système qui s’appuie à la fois sur des méthodes d’apprentissage statistiques (Talismane) pour la détection des mentions et l’analyse syntaxique, et sur des règles pour la détection de la coréférence et la gestion de phénomènes linguistiques complexes mais bien définis dans le français écrit.

2 Détection des anaphores infidèles

La reprise d’un référent dans le texte peut se faire de plusieurs façons. Il peut s’agir d’un pronom : *Paul est dehors.* ***Il*** *attend.*

Lorsque la reprise se fait par un nom, on parle d’*anaphore fidèle* quand le nom est le même (cependant, la détermination et les expansions peuvent être différentes) :

(3) *Le chat* que j’ai adopté court partout... Mais j’aime beaucoup *ce chat*.

On parle au contraire d’*anaphore infidèle* lorsque le nom de reprise est différent :

(4) *Mon chat* boit du lait. *Cet animal* est heureux.

C’est à ce dernier cas que nous nous intéressons ici.

2.1 Hyperonymes

Les deux noms peuvent s’incrire dans une relation lexicale telle que la synonymie, ou l’hyperonymie lorsque la relation est hiérarchique (par exemple : chat > félin > mammifère > animal).

Tester une telle relation entre deux noms permet de vérifier qu'ils sont sémantiquement compatibles : *chat* et *animal* pourront (ou non) être coréférents, mais *chat* et *chien* ne le seront jamais.

Il faut noter que les relations lexicales peuvent varier d'un domaine de spécialité à l'autre. Ainsi, alors que dans l'usage courant *chat* est généralement compris comme un animal, dans le domaine technologique, le terme est synonyme de *conversation électronique*, dans le domaine maritime, il désigne un type de *yacht* (qui est alors un hyperonyme de *chat*), dans le domaine militaire, il renvoie à une machine de siège, etc.

Pour connaître la relation entre deux termes, il faut une ressource contenant une liste de synonymes et d'hyperonymes. Le *WOrdnet Libre du Français (WOLF)* (Sagot & Fišer, 2008) a été construit par traduction automatique du WordNet originel de Princeton. Nous avons cependant décidé d'élaborer une nouvelle ressource à partir d'un dictionnaire français, le *Glawi* (Hathout & Sajous, 2016; Sajous & Hathout, 2015), qui a été construit à partir du *Glaff* (Hathout *et al.*, 2014) et du *Wiktionnaire* [1], un dictionnaire "collaboratif" au sens où chaque internaute peut proposer et modifier des gloses (les définitions), des exemples, mais aussi des synonymes, hyponymes et hyperonymes. Ce dictionnaire permet de relier les différents sens d'un mot à des domaines sémantiques (marine, militaire, technologique, etc.). Nous n'avons cependant pas pu utiliser ces relations proposées par les internautes, parce qu'elles sont codées dans le *Wiktionnaire* et donc dans le *Glawi* d'une façon telle qu'on ne peut pas distinguer les domaines sémantiques.

Néanmoins, les gloses du *Wiktionnaire* sont généralement rédigées de façon à faire apparaître un hyperonyme en début de phrase (définitions dites "par le genre prochain"). Ainsi un *chat* est :

— un ***mammifère*** *carnivore félin de taille moyenne...*,
— un ***bélier*** *recouvert par un chat...*,
— un ***navire*** *servant au chargement et au déchargement...*,
— etc.

De même, une *pomme* est le ***fruit*** *comestible du pommier...*, une *table* est un ***meuble*** *composé d'un plateau posé sur un ou plusieurs pieds...*, et ainsi de suite.

Le *Glawi* propose pour chacune des gloses une analyse syntaxique faite avec Talismane (Urieli, 2013) : nous avons donc extrait pour chaque terme un ou plusieurs hyperonymes à partir du premier nom commun de la glose. Par transitivité (un chat est un mammifère, un mammifère est un animal, etc.), nous avons pu constituer pour chaque terme une collection d'hyperonymes sur plusieurs niveaux. De plus, comme chaque glose est associée à un ou plusieurs domaines de spécialité (marine, botanique, etc.), chaque collection d'hyperonymes est associée à un domaine sémantique particulier. Il est donc possible de ne faire usage que de tel ou tel domaine en fonction du texte à annoter automatiquement : *chat* pourra ainsi être reconnu comme un type d'animal, de conversation, de navire ou d'équipement de guerre en indiquant au système le domaine sémantique du texte.

Nous avons annoté de la sorte 160 000 noms communs (puisque, parmi les classes de mots lexicaux, seuls les noms sont considérés comme référentiels).

2.2 Entités nommées

Les anaphores infidèles peuvent impliquer des entités nommées, c'est-à-dire des désignations de personnes, de lieux et d'organisations (Nouvel *et al.*, 2015) mais aussi d'objets (la "Pyramide du

1. fr.wiktionary.org

Louvre”) ou d’oeuvres (“Les Misérables”).

Dans une relation entre deux expressions, l’une peut être une entité nommée et l’autre un nom commun :

(5) *Paris* est située sur la Seine. *La plus grande ville de France* compte plus de 10 millions d’habitants.

Mais les expressions peuvent aussi être toutes les deux des entités nommées, comme *Paris* et la *Ville Lumière*.

Le seule détection des entités nommées (par exemple : *JRC Names* [2], *CasEN* (Friburger & Maurel, 2004; Maurel *et al.*, 2011), *mXS* (Nouvel *et al.*, 2011), voir (Nouvel *et al.*, 2015, ch. 4)) ne suffit donc pas ici : il faut également obtenir d’une part une liste de noms communs avec lesquels il est possible de reprendre chaque entité (Paris est une *ville*, une *capitale*, le Rhin est un *fleuve*, etc.), d’autre part une liste d’entités nommées (désignations) qui renvoient à la même entité (*Paris* et la *Ville Lumière*, l’*Organisation des Nations Unies* et l’*ONU*, le *Festival de Cannes* et le *Festival International du Film de Cannes* par exemple). C’est pourquoi nous avons cherché à extraire une liste d’entité nommées et à la lier à une base de connaissances.

Parmi les bases disponibles (par exemple *BabelNet* [3], *Wikidata* [4], *DBPedia* [5]), nous avons choisi *Yago* (Mahdisoltani *et al.*, 2014), extrait de *Wikipédia*, *WordNet* et *GeoNames*, qui est téléchargeable, modifiable et redistribuable librement (contrairement à *BabelNet*, par exemple, qui n’est disponible que *via* une API), et qui contient toutes les informations dont nous avons besoin.

La ressource est constituée de plus d’un milliard de triplets de la forme : `(sujet, prédicat, relation)`, par exemple

(6) `<Zinedine_Zidane> <redirectedFrom> "Zizou"@fra`

Cet exemple montre une redirection de “Zizou” vers l’entité `<Zinedine_Zidane>`. En listant toutes les redirections, on obtient la liste de toutes les désignations possibles de Zidane : *El Zid, Zizou, Zinedine Yazid Zidane*, etc. Avec les prédicats `<hasGivenName>` et `<hasFamilyName>`, on ajoute encore des désignations par son nom (*Zidane*) ou son prénom (*Zinédine*). Des règles permettent également de considérer l’usage d’initiales comme *Z. Zidane*.

Par ailleurs, toutes les entités de *Yago* sont reliées à l’ontologie *WordNet* :

(7) `<Zinedine_Zidane> rdf:type <wordnet_football_player_110101634>`

dont les éléments sont associés à des “étiquettes”, y compris en français :

(8) `<wordnet_football_player_110101634> rdfs:label "footballeur"@fra`

Ainsi, Zidane est un *footballeur*, mais aussi, en remontant l’ontologie, un *sportif*, un *concurrent*, un *homme*, etc. À chaque niveau se trouvent un ou plusieurs termes synonymes (*sportif, athlète* ; *concurrent, rival, participant, compétiteur...*). Cette liste d’hyperonymes et de synonymes permet de favoriser un lien de coréférence entre, par exemple, *Zidane* et *athlète* plutôt qu’entre *Zidane* et *ministre* ou *Zidane* et *fleuve*.

2. https://ec.europa.eu/jrc/en/language-technologies/jrc-names
3. http://babelnet.org
4. https://wikidata.org
5. https://dbpedia.org

Un autre prédicat (`<hasGender>`) disponible dans *Yago* est utile pour la coréférence, même s'il concerne les anaphores pronominales, c'est le sexe de la personne : Zinédine Zidane sera repris par *il* et non par *elle* :

(9) `<Zinedine_Zidane> <hasGender> <male>`

Au total, nous avons extrait de *Yago* une liste de 6,4 millions de désignations pour 2,3 millions d'entités, chacune annotée avec

— les désignations possibles de l'entité,
— son type,
— ses hyperonymes,
— le genre du pronom par lequel l'entité peut être reprise (masculin, féminin).

Nous prévoyons également d'utiliser les "infoboxes" de *Wikipédia*, également présentes dans *Yago*, pour la détection des désignations comprenant des noms communs, comme "le ministre de l'Éducation Nationale".

3 Repérage des mentions de groupes et d'anaphores zéros

Les autres phénomènes coréférentiels annoncés dans l'introduction (groupes, anaphores zéros) sont traités à partir d'une analyse syntaxique automatique faite par Talismane (Urieli, 2013). La sortie est convertie en arbre dont les noeuds peuvent être déplacés ou modifiés en fonction de règles prédéfinies.

L'application de ces règles est sensible aux erreurs de l'analyseur, c'est pourquoi nous appliquons auparavant des règles de correction. Ce sont des règles linguistiques, dont nous donnons deux exemples. D'abord, une série de règles vérifie que seuls des éléments de même type sont coordonnés (par exemple, un verbe ne peut pas être coordonné à un nom) : si ce n'est pas le cas, les noeuds sont déplacés pour obtenir une coordination linguistiquement correcte. Une telle correction est notamment nécessaire pour trouver les groupes et les anaphores zéros. Ensuite, une autre règle vérifie que toutes les propositions relatives sont dépendantes d'un nom, et non d'un verbe ou d'une autre partie du discours, sinon la relative est déplacée et "accrochée" au nom le plus plus probable. Une telle correction est notamment nécessaire pour trouver l'antécédent du pronom relatif, qui ne peut être qu'un nom.

3.1 Les groupes

On appelle ici "groupe" les expressions référentielles coordonnées qui, du fait de leur coordination, forment une nouvelle expression référentielle. Ainsi, dans *Pierre et Marie*, il y a trois expressions référentielles et trois référents : *Pierre*, *Marie*, mais aussi aussi le couple (groupe) *Pierre et Marie*.

Le repérage des groupes est nécessaire pour la détection de la coréférence, car ils peuvent être repris par des pronoms, ou même par un nom collectif :

(10) *Jack et Rose* commencent à faire connaissance. *Ils* s'entendent bien. [...] *Le couple*... (Vikidia)

Nous détectons les groupes en recherchant, dans les arbres de chaque phrase, les juxtapositions ou coordinations de deux ou plusieurs noms. Ces coordinations sont marquées pour signaler une mention, mais elles sont aussi annotées en genre et en nombre. Le nombre est toujours pluriel, mais le genre

varie en fonction des individus qui composent le groupe (*Sophie et Marie* est repris par *elles*, et non par *ils*).

Parfois, les groupes ne sont pas coordonnés (cas des antécédents dits “dispersés”) ; ces cas sont pas repérés par notre système :

(11) *Pierre* retrouva *sa femme* au restaurant. *Ils* dînèrent jusqu’à tard dans la nuit.

3.2 Les anaphores zéros

On parle d’anaphore zéro, ou sujet zéro ou non exprimé (indiquée par “ø” dans les exemples), lorsqu’un sujet est mis en “facteur commun” de plusieurs verbes coordonnés :

(12) *Le renard* se glisse, ø se traîne, ø arrive, et ø fait rarement des tentatives inutiles. (Buffon)

Le repérage de ces segments pourtant absents du texte est important car ils donnent des informations sur le référent (Schnedecker & Landragin, 2014) et ils sont donc utiles pour l’interprétation automatique des textes (dans l’exemple, l’action du renard ne se réduit pas à *se glisser*) ou l’étude des chaînes de coréférence (voir par exemple (Schnedecker, 2014)). Il est donc nécessaire de pouvoir “distribuer” ce facteur commun à tous les verbes coordonnés.

Pour ce faire, nous cherchons dans les arbres les verbes juxtaposés ou coordonnés, vérifions que le premier verbe a un sujet mais pas le ou les suivants, et nous dupliquons le sujet du premier verbe sur le deuxième (troisième, etc.) verbe. La mention est marquée sur le verbe (plutôt que d’ajouter le symbole ∅, ce qui modifierait le texte original). Un lien de coréférence est immédiatement établi.

On ne considère ici que les verbes conjugués. On peut aussi parler de sujet zéro pour les participes et les verbes à l’infinitif (Landragin, 2011), mais nous ignorons ces cas car ils n’acceptent pas l’explicitation du sujet. On peut en effet dire *Paul boit et {ø,Paul} fume*, mais pas *Paul pense {ø,*Paul} aller au cinéma ce soir.*

Ce sont parfois des groupes (voir la section précédente) qui sont des anaphores zéros :

(13) *Jack et Rose* sortent des profondeurs du navire et *assistent* au début de la fin du Titanic. (Vikidia)

3.3 Détection de la coréférence des pronoms de première et deuxième personnes dans les citations

Dans les articles de presse, on trouve fréquemment des citations (discours direct) incluant des pronoms (par simplicité, nous parlons de *pronoms*, mais la discussion s’étend aux déterminants possessifs : *mon, ton, son, notre, votre, leur*, etc.) de première et deuxième personnes, dont le référent varie selon la citation, ce qui veut dire qu’on ne peut pas lier tous les “je” du texte ensemble, comme dans l’extrait suivant :

(14) Comme *Solange*, présidente du club des aînés de la commune : “*J’*ai un ordinateur mais *je* ne sais pas *m’*en servir...” *Jacques*, ancien agriculteur, embraie : “L’autre jour, *j’*ai imprimé un document. Toutes les cases se chevauchaient. C’était le bordel, alors *j’*ai débranché l’appareil.” *Marivonne*, épouse de gendarme, assise derrière lui, abonde : “*J’*ai un ordinateur à la maison, *je* m’en sers pour jouer gratuitement au solitaire et puis c’est tout.” (Le Monde)

Les pronoms de première et deuxième personnes peuvent être considérés comme des déictiques plutôt que des anaphoriques : c'est pourquoi certains corpus (par exemple ANCOR (Muzerelle *et al.*, 2013)) font le choix de ne pas les mettre dans une relation de coréférence. Néanmoins, tant dans la perspective de la fouille de textes que dans l'étude des chaînes de coréférence, il semble important de savoir qui se cache derrière les "je" et les "tu" des citations.

Il convient aussi de repérer les incises (*Paul,* ***dit-il****, a eu une bonne note.*) et les verbes "de dire" (*dire* et ses synonymes), notamment quand ils sont situés à la fin avec inversion du sujet (*Jacques s'emporta : "Je ne suis pas d'accord avec Paul : et je pense qu'il a tort",* ***affirma-t-il avec force***), afin d'annoter correctement le sujet de ces verbes, qui est le locuteur du discours direct.

Notre méthode consiste à repérer, avant même l'analyse syntaxique par Talismane, le discours rapporté direct à partir de la présence de guillemets. Dans le discours direct, nous repérons ensuite les incises et les verbes "de dire" avec inversion du sujet afin de les extraire et d'en faire des phrases indépendantes : cela est nécessaire pour que l'analyseur syntaxique fournisse un arbre qui soit ensuite plus aisément exploitable. Une fois l'analyse syntaxique effectuée, les pronoms (et déterminants) de première et deuxième personnes sont repérés et marqués comme mentions. La résolution de la coréference tient compte des différentes annotations : le type de discours (direct ou non), la personne, le nombre, etc.

Notre repérage reste cependant limité : le système détecte des citations telles qu'on les trouve, le plus souvent, dans les articles de presse : elles doivent être encadrées par des guillemets (pas de discours direct libre) et mono-voix (pas de dialogue à l'intérieur de la même citation).

4 Présentation générale du système et de la résolution de la coréférence

Dans cette dernière section, nous décrivons l'architecture de notre système en insistant sur la façon dont nous détectons la coréférence.

Le système étant *end-to-end*, il doit effectuer deux tâches : (1) repérer les mentions dans un texte brut, (2) établir des liens de coréférence entre les expressions référentielles qui renvoient au même référent.

4.1 Repérage des mentions

Le repérage des mentions consiste en plusieurs étapes :

- Repérage des entités nommées et leur annotation (désignations, type, hyperonymes, genre du pronom de reprise, c'est-à-dire le genre du pronom par lequel l'entité peut être reprise : masculin s'il s'agit d'un homme ou d'une entité désignée par un nom masculin, féminin sinon) à partir de la ressources décrite ci-dessus.
- Repérage du discours direct, extraction des incises et des verbes de dire comme décrit ci-dessus.
- Tokenisation pour tenir compte des expressions figées non référentielles (comme *à peine* ou *jouer des coudes*).
- Analyse syntaxique avec Talismane (Urieli, 2013).
- Construction et correction de l'arbre résultant de l'analyse syntaxique.
- Ajout d'informations et d'annotations (groupe, anaphore zéro, etc.).

— Repérage des pronoms non référentiels (adjectifs de météo, verbes impersonnels, etc.).
— Repérage des mentions (expressions référentielles) à partir de l'analyse syntaxique.

4.2 Détection de la coréférence

Les relations de coréférence sont détectées en plusieurs passes, selon le type des relations et d'informations nécessaires (syntaxiques et/ou sémantiques). Faute de place, nous ne donnons qu'une description succincte de ces différentes passes.

Passe 1 : les anaphores liées. On dit qu'il y a anaphore liée lorsque "l'antécédent... est choisi en vertu d'un calcul purement syntaxique" (Corblin, 1995). On obtient alors des "chaînes liées", dont "la construction de la chaîne est exclusivement régie par la syntaxe" (voir aussi (Zribi-Hertz, 1996)).

Le cas le plus évident est celui des *pronoms réfléchis* : leur antécédent est toujours le sujet de la proposition dans laquelle ils apparaissent. Il n'y a besoin d'aucune analyse supplémentaire, ce seul fait syntaxique permet de résoudre l'anaphore dans tous les cas.

Nous résolvons aussi à ce stade les *pronoms relatifs*, qui renvoient toujours au nom dont dépend la proposition relative (éventuellement par l'intermédiaire d'une préposition, puisque le pronom relatif peut être le régime d'une préposition).

Passe 2 : les autres anaphores pronominales. La résolution des autres anaphores (pronoms et déterminants possessifs) requiert à la fois des connaissances syntaxiques et des connaissances sémantiques. Les connaissances syntaxiques permettent de savoir où chercher l'antécédent, et où ne pas le chercher. Par exemple, il ne peut pas y avoir dans la même structure argumentale d'un verbe deux fois le même référent. Dans l'exemple suivant, *lui* ne peut pas désigner Pierre.

(15) Pierre lui donne un cadeau.

La syntaxe permet aussi de détecter les cas de cataphore (lorsque le pronom est *avant* son "antécédent" : *Quand **il** était petit, **Paul** jouait au foot.*). Notre système détecte ces cas de cataphores lorsque le pronom est dans une subordonnée en début de phrase et coréfère au sujet de la proposition principale. Afin d'écarter les faux positifs, nous ne cherchons les cas de cataphores qu'au début du paragraphe.

Dans les autres cas, des candidats sont cherchés dans la phrase courante et la phrase précédente, et, si aucun candidat n'est trouvé, jusqu'au début du paragraphe. Les candidats incompatibles sont éliminés, notamment en fonction du genre et du nombre, mais aussi d'autres règles. Par exemple, le système considère (même si ce n'est pas toujours le cas dans les faits) qu'un pronom démonstratif ne peut avoir pour antécédent qu'un nom dans la phrase précédente. Autre exemple : un pronom de première personne ne peut pas être coréférent à un nom dans une même citation.

Les candidats sont ensuite triés selon plusieurs critères, qui diffèrent selon qu'il s'agit de noms ou de pronoms : la saillance (voir (Mitkov, 2002; Landragin, 2015)) dans le texte et dans le paragraphe en cours (calculée à partir du nombre d'occurrences du terme, de la fonction, du fait que la phrase d'occurrence est une phrase en début de texte ou de paragraphe, etc.), la fonction du candidat et de l'anaphore, la distance en termes de phrases et de mentions, etc. Le meilleur candidat est retenu comme antécédent de l'anaphore.

Passe 3 : les entités nommées. À ce stade, nous avons des chaînes partielles, et tous les pronoms et déterminants possessifs sont censés avoir un antécédent. Les passes suivantes cherchent à fusionner les chaînes partielles qui renvoient au même référent.

Le système commence par les chaînes partielles dont l'une des mentions est une entité nommée : si les entités renvoient au même référent, les chaînes sont fusionnées. Par exemple, deux chaînes contenant l'une *Zinédine* et l'autre *Zizou* (ou l'une *Paris* et l'autre *Ville Lumière*, etc.) seront fusionnées. Nous utilisons la resource lexicale décrite plus haut pour cela.

Passe 4 : les noms communs. Le système assemble ensuite les chaînes partielles en se servant des noms communs. Chaque chaîne partielle peut être fusionnée avec l'une des chaînes précédentes du texte : ce sont des chaînes candidates. Pour chaque chaîne candidate, le ou les noms têtes des syntagmes nominaux de la chaîne candidate et celui ou ceux de la chaîne partielle sont comparés. Les chaînes candidates dont le ou les noms ne sont pas sémantiquement compatibles avec la chaîne courante sont éliminées. Deux noms sont sémantiquement compatibles si l'un est un synonyme ou un hyperonyme de l'autre (y compris pour les entités nommées).

Les chaînes candidates restantes sont triées selon des critères similaires à ceux utilisés pour la phase 2 (saillance, fonction, distance, etc.), ainsi que des critères propres aux syntagmes nominaux, comme la détermination. Par exemple, un défini aura plus de chance de suivre un indéfini que l'inverse (*J'ai vu **un chat** dans la rue... **L'animal** m'a suivi tout l'après-midi* sera plus probable que *J'ai vu **le chat** dans la rue... **Un animal** m'a suivi tout l'après-midi*).

4.3 Évaluation

L'évaluation des systèmes de détection de la coréference est rendue complexe par le manque de corpus de référence. ANCOR (Muzerelle *et al.*, 2013), pour l'heure le seul corpus annoté en coréférences, est un corpus d'oral transcrit, et dont les choix d'annotation ne correspondent pas aux nôtres.

Nous avons comparé notre système à RefGen (Longo, 2013), dont l'architecture est est similaire à la nôtre :

— les mentions sont détectées à partir d'un étiqueteur en partie du discours (TTL (Ion, 2007)),
— la coréférence est résolue à partir de règles. Une première passe détecte les premières mentions de chaînes, en utilisant des éléments de la théorie de l'accessibilité décrite dans (Ariel, 1990). Une seconde passe rattache les expressions référentielles restantes aux différentes chaînes, en fonction de contraintes syntaxiques et sémantiques (forme de la tête du syntagme, compatibilité en genre et nombre, distance, fonction syntaxique, fréquence de co-occurrence, etc.).

Les choix d'annotation de notre système sont quelque peu différents de ceux de RefGen : par exemple, Refgen considère qu'une chaîne recommence à chaque nom propre, et ne prend en compte ni les anaphores infidèles, ni les anaphores zéros, ni les groupes. Nous avons donc réannoté manuellement un sous-ensemble de 3 250 tokens du corpus d'évaluation de RefGen (composés d'un extrait de texte littéraire, de faits divers et de textes de FLE) afin que chaque système soit évalué sur un corpus adapté à ses propres choix d'annotations.

Nous donnons dans le tableau 1 la mesure pour l'identification des mentions et les F-mesures de chacune des quatre mesures habituellement utilisées pour l'évaluation de la coréférence (calculées

avec l'implémentation officielle du script utilisé pour l'évaluation des campagnes CoNLL-2011/2012), en comparant avec RefGen.

	identification	MUC	B^3	CEAF	BLANC
notre système	83.62	58.1	64.53	69.28	53.08
RefGen	50.2	36	31.9	29.9	24.9

TABLE 1 – Évaluation de notre système et de RegGen (Longo, 2013) (F-mesures).

Parmi les systèmes récents, celui de (Godbert & Favre, 2017) reprend les choix d'annotation d'ANCOR et a été évalué sur une partie de ce corpus (23 079 mentions). Les auteurs ne donnent que le score BLANC qui est de 65.7.

CROC (Désoyer *et al.*, 2016) et le système de (Brassier *et al.*, 2018) ont des scores meilleurs sur ANCOR, mais ne fonctionnent que sur des mentions gold.

5 Conclusion

Nous avons présenté un système de détection de la coréférence en français qui s'attache à traiter des phénomènes négligés par les autres systèmes : détection d'anaphores infidèles, notamment avec entités nommées, repérage de groupes et d'anaphores zéros et prise en compte des pronoms et déterminants possessifs de première et deuxième personnes dans le discours rapporté direct.

Notre système utilise des règles plutôt qu'une approche par apprentissage automatique. Même si cette dernière est plus performante (Désoyer *et al.*, 2016; Brassier *et al.*, 2018) pour la détection de la coréférence en générale, les règles semblent bien adaptées pour le traitement de phénomènes spécifiques comme ceux que nous avons évoqués. C'est pourquoi elles pourraient venir en complément de systèmes par apprentissage pour la gestion de ces aspects. Nous obtiendrions alors un système hybride qui combinerait le meilleur des deux approches.

Remerciements

Ce travail a été réalisé avec le soutien du projet ANR Democrat ("Description et modélisation des chaînes de référence : outils pour l'annotation et le traitement automatique", ANR-15-CE38-0008).

Références

ARIEL M. (1990). *Accessing Noun Phrase Antecedents*. Routledge.

BRASSIER M., PURET A., VOISIN-MARRAS A. & GROBOL L. (2018). Classification par paires de mention pour la résolution des coréférences en français parlé interactif. In *Traitement Automatique des Langues Naturelles (TALN'18)*.

CHAROLLES M. (1988). Les plans d'organisation textuelle : périodes, chaînes, portées et séquences. *Pratiques*, **57**, 3–13.

CHASTAIN C. (1975). Reference and context. In K. GUNDERSON, Ed., *Language, mind, and knowledge* : University of Minnesota Press.

CORBLIN F. (1985). Les chaînes de référence : analyse linguistique et traitement automatique. *Intellectica*, **1**(1), 123–143.

CORBLIN F. (1995). *Les formes de reprise dans le discours. Anaphores et chaînes de référence*. Presses Universitaires de Rennes.

DÉSOYER A., LANDRAGIN F., TELLIER I., LEFEUVRE A. & ANTOINE J.-Y. (2015). Les coréférences à l'oral : une expérience d'apprentissage automatique sur le corpus ANCOR. *Traitement Automatique des Langues*, **55**(2), 97–121.

DÉSOYER A., LANDRAGIN F., TELLIER I., LEFEUVRE A., ANTOINE J.-Y. & DINARELLI M. (2016). Coreference resolution for French oral data : Machine learning experiments with ANCOR. In *International Conference on Intelligent Text Processing and Computational Linguistics (CICLing'2016)*.

DUPONT M. (2003). *Une approche cognitive du calcul de la référence*. Thèse de doctorat, Université de Caen.

FRIBURGER N. & MAUREL D. (2004). Finite-state transducer cascades to extract named entities in texts. *Theoretical Computer Science*, **313**(1), 93–104.

GODBERT E. & FAVRE B. (2017). Détection de coréférences de bout en bout en français. In *Traitement Automatique des Langues Naturelles (TALN'17)*.

GROBOL L. (2019). Neural coreference resolution with limited lexical context and explicit mention detection for oral French. In *Computational Models of Reference, Anaphora and Coreference (CRAC)*.

HATHOUT N. & SAJOUS F. (2016). Wiktionnaire's wikicode glawified : a workable French machine-readable dictionary. In *International Conference on Language Resources and Evaluation (LREC'16)*.

HATHOUT N., SAJOUS F. & CALDERONE B. (2014). GLÀFF, a Large Versatile French Lexicon. In *International Conference on Language Resources and Evaluation (LREC'14)*.

ION R. (2007). *Word sense disambiguation methods applied to English and Romanian*. Thèse de doctorat, Romanian Academy, Bucharest.

KABADJOV M. & STEPANOV E. (2015). *The SENSEI Discourse Analysis Tools, 2*. Rapport interne, University of Essex.

LANDRAGIN F. (2011). Une procédure d'analyse et d'annotation des chaînes de coréférence dans des textes écrits. *Corpus*, **10**, 61–80.

LANDRAGIN F. (2015). Sur les aspects multicritères et multidimensionnels de la saillance. In M. BOISSEAU & A. HAMM, Eds., *Saillance. La saillance en langue et en discours, Volume 2*, Annales Littéraires de l'Université de Franche-Comté, p. 15–29. Presses universitaires de Franche-Comté.

LONGO L. (2013). *Vers des moteurs de recherche" intelligents" : un outil de détection automatique de thèmes. Méthode basée sur l'identification automatique des chaînes de référence*. Thèse de doctorat, Université de Strasbourg.

MAHDISOLTANI F., BIEGA J. & SUCHANEK F. (2014). Yago3 : A knowledge base from multilingual wikipedias. In *7th Biennial Conference on Innovative Data Systems Research*.

MAUREL D., FRIBURGER N., ANTOINE J.-Y., ESHKOL I. & NOUVEL D. (2011). Cascades de transducteurs autour de la reconnaissance des entités nommées. *Traitement automatique des langues*, **52**(1), 69–96.

MITKOV R. (2002). *Anaphora resolution*. Routledge.

MUZERELLE J., LEFEUVRE A., ANTOINE J.-Y., SCHANG E., MAUREL D., VILLANEAU J. & ESHKOL I. (2013). ANCOR, premier corpus de français parlé d'envergure annoté en coréférence et distribué librement. In *Traitement Automatique des Langues Naturelles (TALN'13)*.

NOUVEL D., ANTOINE J.-Y., FRIBURGER N. & SOULET A. (2011). Recognizing named entities using automatically extracted transduction rules. In *Language & Technology Conference (LTC'11)*.

NOUVEL D., EHRMANN M. & ROSSET S. (2015). *Les entités nommées pour le traitement automatique des langues*. ISTE editions.

POESIO M., STUCKARDT R. & VERSLEY Y. (2016). *Anaphora resolution*. Springer.

SAGOT B. & FIŠER D. (2008). Building a free French wordnet from multilingual resources. OntoLex.

SAJOUS F. & HATHOUT N. (2015). GLAWI, a free XML-encoded Machine-Readable Dictionary built from the French Wiktionary. In *Proceedings of the of the eLex 2015 conference*, p. 405–426, Herstmonceux, England.

SCHNEDECKER C. (1997). *Nom propre et chaînes de référence*. Metz : Librairie Klincksieck.

SCHNEDECKER C. (2014). Chaînes de référence et variations selon le genre. *Langages*, **195**(3), 23–42.

SCHNEDECKER C. & LANDRAGIN F. (2014). Les chaînes de référence : présentation. *Langages*, **195**(3), 3–22.

TROUILLEUX F. (2001). *Identification des reprises et interprétation automatique des expressions pronominales dans des textes en français*. Thèse de doctorat, Université Blaise-Pascal de Clermont-Ferrand.

URIELI A. (2013). *Robust French syntax analysis : reconciling statistical methods and linguistic knowledge in the Talismane toolkit*. Thèse de doctorat, Université de Toulouse II le Mirail.

VICTORRI B. (2005). Le calcul de la référence. In P. ENJALBERT, Ed., *Sémantique et traitement automatique des langues* : Hermès.

ZRIBI-HERTZ A. (1996). *L'anaphore et les pronoms : une introduction à la syntaxe générative*. Presses Universitaire du Septentrion.

État de l'art des méthodes d'apprentissage profond pour l'extraction automatique de termes-clés

Ygor Gallina
LS2N, Université de Nantes, France
ygor.gallina@univ-nantes.fr

RÉSUMÉ
Les termes-clés facilitent la recherche de documents dans de larges collections de données. Le coût d'annotation de document en termes-clés très élevé, c'est pourquoi les chercheurs s'intéressent à cette problématique. Dans cet article nous présentons un état de l'art sur l'extraction automatique de termes-clés en nous intéressant particulièrement aux modèles d'apprentissage profond. En effet, la récente publication d'un demi-million de documents annotés à permis le développement de modèles neuronaux profonds.

ABSTRACT
State of the art of deep learning methods for automatic keyphrase extraction

The use of keyphrases benefits document retrieval in large collection of data. The high cost of annotating documents with keyphrases leads researchers to address this issue. In this article, we outline a state of the art regarding the automatic extraction of keyphrases while focusing on deep learning models. Indeed, the recent availability of half a million of annotated documents made the development of deep neural models possible.

MOTS-CLÉS : extraction de termes-clés ; apprentissage profond ; état de l'art.

KEYWORDS: keyphrase extraction ; deep learning ; state of the art.

1 Introduction

Les termes-clés, également appelés mots-clés, sont des mots ou expressions polylexicales qui représentent les concepts importants d'un document (Evans & Zhai, 1996). Ils se présentent généralement sous la forme de syntagmes nominaux non récursifs contenant majoritairement des noms communs et adjectifs qualificatifs et relationnels (Daille, 2000). Les termes-clés sont essentiellement utilisés pour indexer les documents et naviguer dans les bibliothèques numériques (Witten *et al.*, 2009). En effet, ils sont une alternative plus synthétique et abstraite à l'indexation plein texte, qui elle, se limite aux seuls mots présents dans le document. Ils sont aussi utilisés pour d'autres tâches du Traitement Automatique de la Langue comme le résumé automatique (Litvak & Last, 2008) et la classification de documents (Hulth & Megyesi, 2006).

Malgré les nombreux avantages que procurent les termes-clés, très peu de documents en sont pourvus. En effet, le coût d'annotation par des indexeurs professionnels (ingénieurs documentalistes) est très élevé. C'est pourquoi la communauté scientifique s'intéresse à l'extraction automatique de termes-clés, notamment au travers de campagnes d'évaluations telles que SemEval (Kim *et al.*, 2010; Augenstein

Espace : la station spatiale chinoise Tiangong-1 est tombée dans le Pacifique

4 avril 2018. – Le lundi 2 avril, les autorités spatiales chinoises ont confirmé le désorbitage du laboratoire spatial Tiangong-1. La majeure partie de la structure s'est consumée lors de la réentrée atmosphérique avant que les restes ne s'écrasent dans l'Océan Pacifique à 8 h 15 heure de Pékin (0 h 15 UTC).

Tiangong-1 («Palais Céleste 1»), était la première station spatiale chinoise. Lancée en 2011, les autorités ont rappelés les derniers astronautes en 2013 avant d'en perdre le contrôle en 2016.

Selon les rapports, la deuxième station spatiale chinoise, Tiangong-2, reste en orbite. La Chine prévoit de lancer une station spatiale permanente en 2022, tandis que les États-Unis songent à se retirer du financement de la Station spatiale internationale à l'horizon 2024.

Termes-clés extraits : Tiangong-1 ; Océan pacifique ; réentrée athmosphérique.

Espace : la station spatiale chinoise Tiangong-1 est tombée dans le Pacifique

4 avril 2018. – Le lundi 2 avril, les autorités spatiales chinoises ont confirmé le désorbitage du laboratoire spatial Tiangong-1. La majeure partie de la structure s'est consumée lors de la réentrée atmosphérique avant que les restes ne s'écrasent dans l'Océan Pacifique à 8 h 15 heure de Pékin (0 h 15 UTC).

Tiangong-1 («Palais Céleste 1»), était la première station spatiale chinoise. Lancée en 2011, les autorités ont rappelés les derniers astronautes en 2013 avant d'en perdre le contrôle en 2016.

Selon les rapports, la deuxième station spatiale chinoise, Tiangong-2, reste en orbite. La Chine prévoit de lancer une station spatiale permanente en 2022, tandis que les États-Unis songent à se retirer du financement de la Station spatiale internationale à l'horizon 2024.

Termes-clés extraits : station spatiale ; Océan pacifique ; *chute* ; Chine.

FIGURE 1 – Exemple d'indexation par termes-clés d'une brève journalistique par deux annotateurs. Les concepts similaires sont surligné avec la même couleur. Les termes-clés extraits absent du document sont écrit en italique.

et al., 2017) [1] ou DEFT (Paroubek *et al.*, 2012; Daille *et al.*, 2017). L'atelier NCA (Gollapalli *et al.*, 2015) (Novel Computationel Approach to Keyphrase Extraction) a été consacré en 2015 à cette tâche. Les conférences traitant des bibliothèques numériques telles que la Joint Conference on Digital Libraries (JCDL) s'y intéressent aussi.

L'extraction automatique de termes-clés consiste a assigner à un document un ensemble de termes-clés extraits à partir du contenu du document ou de ressources externes telles que des thésaurus. Il s'agit d'une tâche complexe de par sa subjectivité. Nous présentons dans la Figure 1 l'annotation d'un article de nouvelles journalistiques en termes-clés par deux lecteurs. Dans cet exemple, bien que les concepts de « station spatiale », d'« océan pacifique » et de « désorbitage » soient capturés par les deux annotateurs, seulement un terme-clé est commun aux deux annotations.

1. La tâche décrite par SemEval-2017 consiste à identifier et classifier des relations entre des processus, matériaux et tâches décrites dans des articles scientifiques.

Dans cet article nous présentons un état de l'art des méthodes d'extraction automatiques de termes-clés en se restreignant aux cadres applicatifs des publications scientifiques et des articles journalistiques. Une attention toute particulière sera portée aux modèles neuronaux qui présentent à l'écriture de cet article les meilleures performances.

Le reste de l'article se présente comme suit. La section 2 présente les modèles d'extraction automatique de termes-clés en commençant par les modèles dits traditionnels puis les modèles neuronaux. La section 3 présente les différents jeux de données et les mesures utilisées pour l'évaluation puis les résultats des modèles que nous avons identifiés comme représentatifs. Enfin nous conclurons et présenterons des perspectives de recherches.

2 Modèles d'extraction automatique de termes-clés

Les travaux de la littérature peuvent être séparés en deux catégories, ceux qui proposent des modèles dits traditionnels, qui reposent sur une chaîne de traitement (c.-à-d. segmentation en mots, étiquetage morpho-syntaxique, analyse en dépendances), et ceux qui exploitent des modèles neuronaux dits de bout-en-bout.

2.1 Modèles traditionnels

Les modèles traditionnels reposent sur un schéma de pondération des unités textuelles présentes dans le document. Ils opèrent le plus souvent en deux étapes, la sélection des termes-clés candidats et leur ordonnancement à partir de traits définis manuellement. Lorsqu'ils s'appuient sur une méthode supervisée, ces derniers considèrent l'extraction automatique de termes-clés comme une tâche de classification binaire (terme-clé ou non).

2.1.1 Sélection des termes-clés candidats

L'étape de sélection des termes-clés candidats consiste à identifier, à l'aide d'heuristiques, les mots ou expressions polylexicales du document possédant les caractéristiques d'un terme-clé. Pour cela, deux heuristiques différentes sont couramment utilisées : l'extraction de n-grammes et l'extraction de syntagmes nominaux.

Les n-grammes sont des séquences de n mots adjacents qui apparaissent dans le document. La nature exhaustive de l'extraction de n-grammes permet de maximiser le nombre de candidats corrects (p. ex. « station spatiale »), mais aussi le nombre de candidats incorrects (p. ex. « spatiale chinoise »). Plusieurs stratégies ont été proposées pour résoudre ce problème comme le filtrage des n-grammes contenant des mots outils (Hulth, 2003) ou ceux dont la fréquence est inférieure à un seuil donné (Medelyan *et al.*, 2009).

Les syntagmes nominaux sont des séquences de plusieurs mots constitués d'une tête, qui est un nom, et de satellites, le plus souvent des adjectifs. Des patrons grammaticaux, qu'ils soient définis manuellement (Bougouin *et al.*, 2013) ou inférés à partir de données annotées (Hulth, 2003), sont utilisés pour extraire les syntagmes nominaux du document. À la différence des n-grammes, les syntagmes nominaux sont naturellement des candidats corrects, et ne nécessitent donc pas de filtrage

supplémentaire. En revanche, ils ne représentent qu'une partie limitée des unités textuelles du document, ce qui engendre des problèmes de silence.

2.1.2 Ordonnancement des termes-clés

Une fois les termes-clés candidats identifiés, il s'agit de les pondérer en fonction de leur importance dans le document. Pour cela, les schémas de pondération issus de la recherche d'information comme le TF-IDF (Salton *et al.*, 1975) peuvent être mis en oeuvre. D'autres modèles, proposés spécifiquement pour l'extraction de termes-clés, offrent cependant de meilleures performances.

Witten *et al.* (1999) ont été les premiers à proposer un modèle dédié à l'extraction automatique de termes-clés. Reposant sur un algorithme de classification naïve bayesienne et seulement deux traits (TF-IDF et la position), leur modèle calcule pour chaque candidat la probabilité qu'il soit un terme-clé. Ce travail précurseur a ouvert la voie à d'autres travaux combinant différents algorithmes de classification et/ou proposant de nouveaux traits (Hulth, 2003; Medelyan *et al.*, 2009; Jiang *et al.*, 2009; Sarkar *et al.*, 2010). Jusqu'à très récemment (Meng *et al.*, 2017), peu d'études portant sur les modèles supervisés ont vues le jour car la quantité de données annotées était alors très limitée. C'est la raison pour laquelle de nombreux chercheurs se sont penchés sur le développement de modèles non supervisés.

TextRank (Mihalcea & Tarau, 2004) est sans doute un des modèles non supervisés les plus populaires. Dans ce modèle, le document est représenté sous la forme d'un graphe sur lequel l'algorithme PageRank (Page *et al.*, 1999) est appliqué pour estimer l'importance des mots. L'idée sous-jacente à TextRank est qu'un mot est important s'il cooccurre avec un grand nombre de mots, et si les mots avec lesquels il cooccurre sont eux aussi importants. Parmi les améliorations proposées dans la littérature pour ce modèle, deux directions de recherche émergent. La première concerne l'utilisation de ressources externes pour améliorer la représentation en graphe du document (Wan & Xiao, 2008a; Liu *et al.*, 2010; Gollapalli & Caragea, 2014). La seconde se concentre sur l'amélioration de l'algorithme d'ordonnancement des noeuds du graphe (Bougouin *et al.*, 2013; Florescu & Caragea, 2017; Boudin, 2018).

Certain travaux se démarquent des autres par leurs approches originales. Les travaux de (Tomokiyo & Hurst, 2003), par exemple, considèrent qu'un terme-clé doit être grammatical et informatif et utilisent des modèles de langues pour ordonner les candidats selon ces deux critères. Liu *et al.* (2009), quand à eux, font l'hypothèse que les concepts importants du document sont véhiculés par plusieurs candidats et se fondent sur des regroupements automatiques pour les pondérer.

Si l'on regarde les travaux précédents, peu d'entre eux exploitent la sémantique des termes-clés, et la plupart se contentent de traits de surfaces. Bennani-Smires *et al.* (2018) proposent un premier effort dans cette direction en utilisant les plongements de mots pour ordonner les termes-clés candidats en fonction de leur proximité sémantique au document.

Les modèles présentés dans cette section souffrent de deux écueils : ils ne capturent ni n'exploitent la sémantique des unités textuelles du document ; et l'architecture en chaîne de traitement, sur laquelle ils s'appuient, propage et aggrave les erreurs commises à chaque étape. Pour pallier à ces écueils, les chercheurs se sont orientés vers le développement de modèles neuronaux profonds, que nous décrivons dans la section suivante.

2.2 Modèles neuronaux profond

Contrairement aux modèles présentés précédemment, les modèles neuronaux reposent sur une architecture de bout-en-bout et infèrent des traits à partir de grandes quantités de données d'apprentissage. Accéder à de grandes quantités de données annotées reste cependant compliqué et a longtemps constitué le frein principal au développement de modèles supervisés. Zhang *et al.* (2016) ont été les premiers à apporter une solution à ce problème à partir de données folksonomiques (issues de l'indexation personnelle). Leur idée est de considérer les mots-dièses (*hashtags*) présents dans les microblogs comme une vérité terrain, pour ensuite entraîner un modèle neuronal d'annotation en séquence. Bien que sortant du cadre de cette étude de part les données utilisées, ce travail montre qu'il est possible d'entraîner des modèles neuronaux profonds et a initié une branche de recherche reposant sur l'acquisition massive de données folksonomiques.

Continuant dans cette direction, Meng *et al.* (2017) ont constitué « Kp20k », un corpus comptant un demi-million de résumés d'articles scientifiques annotés en termes-clés par leurs auteurs dans le but d'entraîner un modèle encodeur-décodeur. Il s'agit, à ce jour, de la plus grande collection disponible de données de ce type.

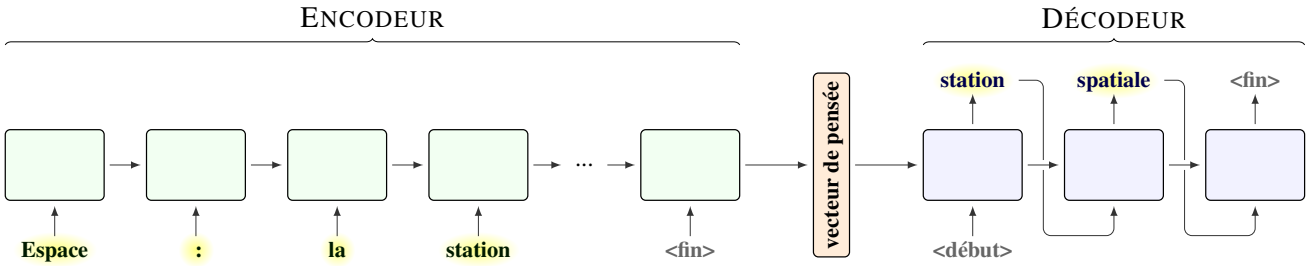


FIGURE 2 – Exemple de modèle *encodeur-décodeur* récurrent appliqué à l'extraction automatique de termes-clés.

Le modèle encodeur-décodeur (Cho *et al.*, 2014; Sutskever *et al.*, 2014) (cf. Figure 2) se décompose en deux parties : un encodeur qui crée un vecteur de pensée (également appelé vecteur latent) à partir de la séquence d'entrée ; et un décodeur qui génère une séquence de sortie. L'encodeur récurrent calcule pour chaque mot un vecteur de pensée conditionné par le vecteur de pensée précédent. Une fois la séquence d'entrée encodée, le décodeur génère séquentiellement les mots en fonction du mot précédemment généré et de son vecteur de pensée.

Ce modèle impose de définir un vocabulaire, certains mots peu fréquents ou non présents dans le corpus d'entraînement ne pourront donc pas être générés. Pour étendre le vocabulaire Gu *et al.* (2016) et See *et al.* (2017) proposent des mécanismes de copie, qui ajoutent la possibilité au réseau de copier un mot à partir de la séquence d'entrée.

La stratégie de décodage utilisée par Meng *et al.* (2017) permet de générer une séquence de mots, donc un seul terme-clé. Pour générer un ensemble de termes-clés, un algorithme de recherche en faisceau (OW & MORTON, 1988) est utilisé. Le décodage est alors effectué à partir des n mots les plus probables au lieu d'un seul. Les termes-clés générés par la recherche en faisceau ne sont pas liés les uns aux autres, ils sont donc souvent redondants. Pour limiter ce phénomène, Chen *et al.* (2018) génèrent directement une séquence de termes-clés divisés par des marqueurs de séparation, et appliquent un mécanisme de revue qui favorise la diversité des mots dans la séquence produite. Bien que bénéfique pour les performances, l'ajout de ces différents mécanismes augmente significativement le nombre de paramètres et en complexifie l'entraînement.

L'entraînement des modèles encodeur-décodeur nécessite de très grandes quantités de données annotées pour atteindre un niveau de performance satisfaisant (Meng *et al.*, 2017). Collecter des données annotées n'est cependant pas si simple, c'est pourquoi ils se sont intéressés à l'apprentissage semi-supervisé. Ye & Wang (2018) proposent deux moyens d'utiliser des données non annotées pour l'entraînement du modèle encodeur-décodeur. Le premier consiste à appliquer des modèles existants sur des données non annotées pour produire de nouvelles données d'entraînement. Le second consiste à entraîner conjointement le modèle à effectuer deux tâches mutuellement bénéfiques : l'extraction de termes-clés et la génération de titre.

3 Résultats expérimentaux

Dans cette section nous présentons les jeux de données ainsi que les mesures d'évaluation principalement utilisées par la communauté. Dans un second temps nous présentons les résultats expérimentaux des modèles de l'état de l'art et mettons en exergue leurs forces et faiblesses.

3.1 Jeux de données et mesures d'évaluation

Les ensembles de données disponibles dans la littérature sont différenciés par rapport à la nature des documents qui les composent : les articles scientifiques, les notices (résumés et titres d'articles scientifiques) et les nouvelles d'actualités. Nous présentons dans la Table 1 un sous-ensemble de ces jeux de données ventilés selon leur nature.

Nom	Lang.	Ann.	#entr.	#test	#mots	#tc	%abs
CSTR (Witten *et al.*, 1999)	en	A	130	500	11 501	5.4	18.7
SemEval-2010 (Kim *et al.*, 2010)	en	$A \cup L$	144	100	7 961	14.7	19.7
Inspec (Hulth, 2003)	en	I	1 000	500	135	9.8	22.4
TermITH-Eval (Bougouin *et al.*, 2016)	fr	I	-	400	164.7	11.8	59.8
KP20k (Meng *et al.*, 2017)	en	A	530 809	20 000	176	5.3	42.6
DUC-2001 (Wan & Xiao, 2008b)	en	L	-	308	847	8.1	3.7
110-PT-BN-KP (Marujo *et al.*, 2013)	pt	L	100	10	439	27.6	7.5

TABLE 1 – Statistiques des jeux de données utilisés. Les documents sont annotés par les auteurs (A), des lecteurs (L) ou des ingénieurs documentalistes (I). Les colonnes #entr. et #test présentent le nombre de documents dans les corpus d'entraînement et de test. Les colonnes #tc et #mots présentent le nombre moyen de termes-clés et de mots par document. La colonne %abs présente le taux de termes-clés qui n'apparaissent pas dans le document (calculé sur le corpus de test).

La plupart des jeux de données sont en anglais, même si quelques initiatives ont vues le jour dans d'autres langues comme le français ou le portugais. Il est important de noter que ces ensembles sont de taille plutôt réduite et que seul Kp20k dispose de suffisamment de documents pour permettre l'entraînement de modèles neuronaux profonds.

Chaque type d'annotateur annote d'une manière différente. En effet, les ingénieurs documentalistes – professionnels de l'indexation – annotent principalement les notices scientifiques. Leur annotation,

coûteuse, constitue une référence de qualité. Ils réalisent l'extraction de termes-clés en domaine ouvert ou fermé. L'extraction de termes-clés en domaine fermé implique d'assigner un terme-clé choisi parmi une liste de référence, par exemple dans un thésaurus. L'extraction de termes-clés en domaine ouvert, aussi utilisée par les auteurs et lecteurs, fait appel à l'expertise de l'annotateur dans le domaine pour le choix des termes-clés. Ce type d'extraction non contraint peut introduire de la variabilité dans les termes-clés choisis. À la différence des ingénieurs documentalistes dont la finalité de l'annotation est l'indexation, les lecteurs choisissent les termes-clés selon leur expertise du domaine et leur usage de cette annotation. Witten *et al.* (1999) fait l'hypothèse que les auteurs choisissent les termes-clés dans le but de maximiser la visibilité de leur article et constitue une référence biaisée par les sujets de recherche populaires au moment du choix de ces termes-clés.

Les modèles traditionnels, présentés dans la section 2, sont majoritairement extractifs. Ils limitent l'extraction de termes-clés aux seuls mots ou expressions polylexicales présentes dans le document. Le taux de termes-clés de référence absent dans le document permet alors de donner une idée des performances maximales de ces modèles.

Le taux de termes-clés absents diffère en fonction de la nature du document. Les nouvelles d'actualités sont des documents courts, pourtant la très grande majorité des termes-clés annotés sont présents dans le document. Ce phénomène s'explique en partie par l'annotation généralement extractive des lecteurs (Wang *et al.*, 2015). Les notices, elles aussi, sont très courtes ($\approx$ 160 mots) et le taux de termes-clés de référence absents est plus élevé que dans les articles scientifiques, en effet elles en sont des représentations synthétiques.

La disponibilité de différents corpus annotés permet d'effectuer une évaluation automatique des modèles, bien moins coûteuse qu'une évaluation manuelle, en comparant la sortie du modèle avec les termes-clés de référence du corpus.

Il n'y a pas de consensus sur les mesures d'évaluation utilisée, mais les plus usités sont la précision, le rappel et la f-mesure calculés à partir des 5 ou 10 premiers termes-clés. Ces mesures ne prennent pas en compte l'ordre des termes-clés extraits, or les méthodes s'astreignent à les ordonner. Pour préférer les modèles classant les termes-clés corrects en premier, des mesures comme la Bpref (Binary Preference), la MRR (Mean Reciprocal Rank), le NDCG (Normalized Discounted Cumulative Gain) et la MAP (Mean Average Precision) sont utilisés par (Basaldella *et al.*, 2016; Liu *et al.*, 2010; Boudin, 2018; Marujo *et al.*, 2013; Chen *et al.*, 2018).

Les termes-clés extraits sont considérés comme corrects s'ils correspondent exactement à la référence. Ce type de correspondance ne prend pas en compte la variabilité lexicale, c'est pourquoi il est d'usage de comparer les formes racinisées des termes-clés. Cela peut parfois introduire des erreurs (p. ex. la forme racinisée de « empire » et « empirique » est « empir »). Une évaluation manuelle permet de prendre en compte d'autres variantes, telles que les variantes lexicales (p. ex. « apprentissage machine » et « apprentissage automatique »), et de mettre en avant la sous estimation des évaluations automatiques (Bougouin *et al.*, 2016).

3.2 Résultats

Nous présentons dans la Table 2 un sous-ensemble représentatif des résultats des modèles de la littérature. Les résultats des modèles présentés sont ceux reportés dans les articles les décrivant.

Nous observons la grande diversité des mesures et des jeux de données utilisés pour l'évaluation des

Collection \ Modèle(s)	F_1	F_1@5	F_1@10	Bpref	MRR	MAP
CSTR						
(Witten *et al.*, 1999)	-	18.3†	**18.4**†	-	-	-
SemEval-2010						
(Boudin, 2018)	-	12.2	14.5	-	-	11.8
(Chen *et al.*, 2018)	-	32.0‡	**32.0**‡	-	-	-
Inspec						
(Hulth, 2003)	33.9	-	-	-	-	-
(Mihalcea & Tarau, 2004)	36.2	-	-	-	-	-
(Liu *et al.*, 2010)	-	24.2	-	27.4	**58.3**	-
(Boudin, 2018)	-	25.9	30.6	-	-	29.2
Kp20k						
(Meng *et al.*, 2017)	-	**32.8**‡	25.5‡	-	-	-
DUC2001						
(Liu *et al.*, 2010)	-	-	31.2	21.4	**63.8**	-

TABLE 2 – Résultats présentés dans les articles. Les meilleurs score reportés pour chaque jeu de données sont mis en gras. † Calculé à partir de la moyenne des vrai positifs. ‡ Scores calculés à partir des termes-clés présent dans le document.

modèles, ce qui rend très complexe la comparaison des modèles entre eux. Bien que la f-mesure soit la métrique la plus utilisée, le choix du rang ne fait pas consensus bien que les modèles récents la rapportent généralement aux rangs 5 et 10. Il est important de noter que les performances de certains modèles sont calculés sans tenir compte des termes-clés de référence absents du contenu du document. Les mesures tenant compte de l'ordre des termes-clés sont très peu utilisées, bien qu'apportant un moyen de comparaison supplémentaire.

Ce tableau montre aussi que les valeurs de f-mesures obtenues varient entre 10 et 30%, ces scores sont très bas par rapport à d'autres tâches de traitement automatique de la langue. Par exemple, le modèle de la boîte à outils Spacy [2] pour la reconnaissance d'entité nommée obtient une valeur de f-mesures de 85%.

4 Discussion/Perspectives

Dans cet article, nous avons présenté l'état actuel de la recherche pour l'extraction automatique de termes-clés.

Beaucoup de travaux s'intéressent à cette tâche, nous avons présenté les modèles traditionnels reposant sur une chaîne de traitement en deux étapes : la sélection des candidats puis leur ordonnancement. La disponibilité du corpus Kp20K a permis le développement de modèles utilisant l'apprentissage profond pour générer des termes-clés. Ces modèles utilisent une architecture de bout-en-bout sans étape de sélection de candidats, permettant ainsi d'outrepasser les propagations d'erreurs liés aux

2. https://spacy.io/models/en

modèles utilisant des chaînes de traitement.

Malgré cela, nous avons montré que les modèles les plus performants atteignent une valeur de f-mesure de 30%. Ces performances s'expliquent par une évaluation sous-optimale ne prenant pas en compte la variabilité lexicale des termes-clés extrait, l'extractivité des modèles proposés, mais aussi par la subjectivité de la tâche. L'évaluation de cette tâche reste un champ de recherche ouvert. Elle peut être améliorée en prenant en compte les variations lexicales liés à la synonymie et aux relations d'hyperonymies, avec l'utilisation de réseaux lexicaux par exemple.

Nos travaux futurs s'intéresseront au développement des modèles neuronaux profonds, notamment à l'amélioration des modèles génératifs et de leur abstractivité. Mais aussi à l'investigation de moyens alternatifs d'appliquer les modèles neuronaux profonds à l'extraction de termes-clés.

Références

AUGENSTEIN I., DAS M., RIEDEL S., VIKRAMAN L. & MCCALLUM A. (2017). SemEval 2017 Task 10 : ScienceIE - Extracting Keyphrases and Relations from Scientific Publications. In *Proceedings of the 11th International Workshop on Semantic Evaluations (SemEval-2017)*, Vancouver, Canada : Association for Computational Linguistics. arXiv : 1704.02853.

BASALDELLA M., CHIARADIA G. & TASSO C. (2016). Evaluating anaphora and coreference resolution to improve automatic keyphrase extraction. In *In Proceedings of COLING 2016, 26th International Conference on Computational Linguistics*, p.1̃1, Osaka, Japan.

BENNANI-SMIRES K., MUSAT C., HOSSMANN A., BAERISWYL M. & JAGGI M. (2018). Simple Unsupervised Keyphrase Extraction using Sentence Embeddings. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, p. 221–229, Brussels, Belgium : Association for Computational Linguistics.

BOUDIN F. (2018). Unsupervised Keyphrase Extraction with Multipartite Graphs. In *Proceedings of NAACL-HLT 2018* : Association for Computational Linguistics. arXiv : 1803.08721.

BOUGOUIN A., BARREAUX S., ROMARY L., BOUDIN F. & DAILLE B. (2016). TermITH-Eval : a French Standard-Based Resource for Keyphrase Extraction Evaluation. In *LREC - Language Resources and Evaluation Conference*, Potoroz, Slovenia.

BOUGOUIN A., BOUDIN F. & DAILLE B. (2013). TopicRank : Graph-Based Topic Ranking for Keyphrase Extraction. In *International Joint Conference on Natural Language Processing (IJCNLP)*, p. 543–551, Nagoya, Japan.

CHEN J., ZHANG X., WU Y., YAN Z. & LI Z. (2018). Keyphrase Generation with Correlation Constraints. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium : Association for Computational Linguistics. arXiv : 1808.07185.

CHO K., VAN MERRIENBOER B., GULCEHRE C., BAHDANAU D., BOUGARES F., SCHWENK H. & BENGIO Y. (2014). Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *arXiv :1406.1078 [cs, stat].* arXiv : 1406.1078.

DAILLE B. (2000). Morphological Rule Induction for Terminology Acquistion. In *Proceedings of the 18th Conference on Computational Linguistics - Volume 1*, COLING '00, p. 215–221, Stroudsburg, PA, USA : Association for Computational Linguistics. event-place : Saarbrücken, Germany.

DAILLE B., BARREAUX S., BOUGOUIN A., BOUDIN F., CRAM D. & HAZEM A. (2017). Indexation d'articles scientifiques Présentation et résultats du défi fouille de textes DEFT 2016. *Recherche d'information, document et web sémantique*, **17**(1).

EVANS D. A. & ZHAI C. (1996). Noun-phrase Analysis in Unrestricted Text for Information Retrieval. In *Proceedings of the 34th Annual Meeting on Association for Computational Linguistics*, ACL '96, p. 17–24, Stroudsburg, PA, USA : Association for Computational Linguistics. event-place : Santa Cruz, California.

FLORESCU C. & CARAGEA C. (2017). PositionRank : An Unsupervised Approach to Keyphrase Extraction from Scholarly Documents. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 1105–1115, Vancouver, Canada : Association for Computational Linguistics.

GOLLAPALLI S. D. & CARAGEA C. (2014). Extracting Keyphrases from Research Papers Using Citation Networks. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*.

S. D. GOLLAPALLI, C. CARAGEA, X. LI & C. L. GILES, Eds. (2015). *Proceedings of the ACL 2015 Workshop on Novel Computational Approaches to Keyphrase Extraction*. Beijing, China : Association for Computational Linguistics.

GU J., LU Z., LI H. & LI V. O. K. (2016). Incorporating Copying Mechanism in Sequence-to-Sequence Learning.

HULTH A. (2003). Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the 2003 conference on Empirical methods in natural language processing -*, volume 10, p. 216–223, Not Known : Association for Computational Linguistics.

HULTH A. & MEGYESI B. B. (2006). A Study on Automatically Extracted Keywords in Text Categorization. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, ACL-44, p. 537–544, Stroudsburg, PA, USA : Association for Computational Linguistics. event-place : Sydney, Australia.

JIANG X., HU Y. & LI H. (2009). A ranking approach to keyphrase extraction. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval - SIGIR '09*, p. 756, Boston, MA, USA : ACM Press.

KIM S. N., MEDELYAN O., KAN M.-Y. & BALDWIN T. (2010). SemEval-2010 Task 5 : Automatic Keyphrase Extraction from Scientific Articles. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, SemEval '10, p. 21–26, Stroudsburg, PA, USA : Association for Computational Linguistics.

LITVAK M. & LAST M. (2008). Graph-based Keyword Extraction for Single-document Summarization. In *Proceedings of the Workshop on Multi-source Multilingual Information Extraction and Summarization*, MMIES '08, p. 17–24, Stroudsburg, PA, USA : Association for Computational Linguistics. event-place : Manchester, United Kingdom.

LIU Z., HUANG W., ZHENG Y. & SUN M. (2010). Automatic Keyphrase Extraction via Topic Decomposition. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, p. 366–376, Stroudsburg, PA, USA : Association for Computational Linguistics.

LIU Z., LI P., ZHENG Y. & SUN M. (2009). Clustering to find exemplar terms for keyphrase extraction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing Volume 1 - EMNLP '09*, volume 1, p. 257, Singapore : Association for Computational Linguistics.

MARUJO L., GERSHMAN A., CARBONELL J., FREDERKING R. & NETO J. P. (2013). Supervised Topical Key Phrase Extraction of News Stories using Crowdsourcing, Light Filtering and Co-reference Normalization. *arXiv :1306.4886 [cs]*. arXiv : 1306.4886.

MEDELYAN O., FRANK E. & WITTEN I. H. (2009). Human-competitive Tagging Using Automatic Keyphrase Extraction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing : Volume 3 - Volume 3*, EMNLP '09, p. 1318–1327, Stroudsburg, PA, USA : Association for Computational Linguistics.

MENG R., ZHAO S., HAN S., HE D., BRUSILOVSKY P. & CHI Y. (2017). Deep keyphrase generation. In *ACL 2017 - 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, volume 1, p. 582–592.

MIHALCEA R. & TARAU P. (2004). TextRank : Bringing Order into Texts. In *Proceedings of {EMNLP-04}and the 2004 Conference on Empirical Methods in Natural Language Processing*, p.$\tilde{8}$, Barcelona, Spain.

OW P. S. & MORTON T. E. (1988). Filtered beam search in scheduling†. *International Journal of Production Research*, **26**(1), 35–62.

PAGE L., BRIN S., MOTWANI R. & WINOGRAD T. (1999). *The PageRank Citation Ranking : Bringing Order to the Web*.

PAROUBEK P., ZWEIGENBAUM P., FOREST D. & GROUIN C. (2012). Indexation libre et contrôlée d'articles scientifiques Présentation et résultats du défi fouille de textes DEFT2012. In *DEFT@TALN*.

SALTON G., WONG A. & YANG C. S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, **18**(11), 613–620.

SARKAR K., NASIPURI M. & GHOSE S. (2010). A New Approach to Keyphrase Extraction Using Neural Networks. *International Journal of Computer Science Issues*, **7**(2). arXiv : 1004.3274.

SEE A., LIU P. J. & MANNING C. D. (2017). Get To The Point : Summarization with Pointer-Generator Networks. *arXiv :1704.04368 [cs]*. arXiv : 1704.04368.

SUTSKEVER I., VINYALS O. & LE Q. V. (2014). Sequence to Sequence Learning with Neural Networks. *arXiv :1409.3215 [cs]*. arXiv : 1409.3215.

TOMOKIYO T. & HURST M. (2003). A language model approach to keyphrase extraction. In *Proceedings of the ACL 2003 workshop on Multiword expressions analysis, acquisition and treatment -*, volume 18, p. 33–40, Not Known : Association for Computational Linguistics.

WAN X. & XIAO J. (2008a). CollabRank : towards a collaborative approach to single-document keyphrase extraction. In *Proceedings of the 22nd International Conference on Computational Linguistics - COLING '08*, volume 1, p. 969–976, Manchester, United Kingdom : Association for Computational Linguistics.

WAN X. & XIAO J. (2008b). Single Document Keyphrase Extraction Using Neighborhood Knowledge. In *Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 2*, AAAI'08, p. 855–860, Chicago, Illinois : AAAI Press.

WANG R., LIU W. & MCDONALD C. (2015). Using Word Embeddings to Enhance Keyword Identification for Scientific Publications. In M. A. SHARAF, M. A. CHEEMA & J. QI, Eds., *Databases Theory and Applications*, Lecture Notes in Computer Science, p. 257–268 : Springer International Publishing.

WITTEN I. H., BAINBRIDGE D. & NICHOLS D. M. (2009). *How to Build a Digital Library, Second Edition*. San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., 2nd edition.

WITTEN I. H., PAYNTER G. W., FRANK E., GUTWIN C. & NEVILL-MANNING C. G. (1999). KEA : practical automatic keyphrase extraction. In *Proceedings of the fourth ACM conference on Digital libraries - DL '99*, p. 254–255, Berkeley, California, United States : ACM Press.

YE H. & WANG L. (2018). Semi-Supervised Learning for Neural Keyphrase Generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium. arXiv : 1808.06773.

ZHANG Q., WANG Y., GONG Y. & HUANG X. (2016). Keyphrase Extraction Using Deep Recurrent Neural Networks on Twitter. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, p. 836–845, Austin, Texas : Association for Computational Linguistics.

Extraction d'opinions pour l'analyse multicritère à partir de corpus oraux transcrits : État de l'art

Emmanuelle Kelodjoue
Laboratoire Informatique de Grenoble, Université Grenoble Alpes
700 Avenue centrale, 38401 Saint Martin d'Hères, France
emmanuelle.kelodjoue-nguemegne@univ-grenoble-alpes.fr

RÉSUMÉ

Cet article présente une revue de la littérature dans les domaines de l'analyse de sentiments et du résumé automatique appliqués aux corpus oraux transcrits. Peu de travaux ont été réalisés dans ces deux domaines avec des corpus oraux transcrits. C'est pourquoi nous présentons ci-dessous les tendances générales dans ces deux domaines et nous nous concentrons ensuite sur les systèmes développés pour le traitement de ce type de données. Les méthodes supervisées pour l'analyse de sentiments et les méthodes extractives pour le résumé automatique sont actuellement dominantes dans le développement des systèmes automatiques pour le traitement des corpus oraux transcrits. Le présent article est une vue d'ensemble de l'état de l'art en analyse de sentiments et du résumé automatique appliqués à des corpus oraux transcrits pour une aide à l'analyse des verbatim.

ABSTRACT

Opinion extraction for multi-criteria analysis of transcribed oral corpora.

This article presents a review of the literature in the areas of sentiment analysis and automatic summarization applied to transcribed oral corpora. Little work has been done in these two areas with transcribed oral corpora. For this reason, we present general trends in these two domains and then focus on the systems developed for processing these type of data. Supervised methods for sentiment analysis and extractive methods for automatic summarization are currently dominant in the development of automatic systems for the processing of transcribed oral corpuses. This article is an overview of the state of the art in sentiment analysis and automatic summarization applied to transcribed oral corpora in order to propose a review system of verbatim.

MOTS-CLÉS : Analyse de sentiments, résumé automatique, corpus oraux transcrits, subjectivité.

KEYWORDS: Sentiment analysis, automatic summarization, spoken transcribed corpuses, subjectivity.

1 Introduction

L'analyse des verbatim a considérablement évolué ces dernières années avec le développement d'Internet et la multiplication des contenus en ligne. Elle permet de caractériser à l'aide de diverses méthodes (analyse lexicale, sémantique, fouille de données, etc.) les paroles échangées dans le cadre d'une conversation. Poland (1995) définit le terme "verbatim" comme la « reproduction mot pour mot de données verbales ». Les mots écrits sont la réplique exacte des paroles enregistrées. Cette reproduction inclut donc toutes les répétitions, les interjections, le bégaiement et les marqueurs

discursifs tels que *heu, hein, ben, etc.* Il existe des campagnes d'évaluation proposant des tâches en analyse de sentiments (DEFT (Défi Fouille Texte) (Nzali *et al.*, 2017), SemEval (Pontiki *et al.*, 2016)), et des tâches de résumé automatique TAC (Text Analysis Conference) (Collins *et al.*, 2017), mais les données mises à disposition pour évaluer les systèmes construits ont été essentiellement des données textuelles (tweets, critiques cinématographiques, messages de forums, etc.). De même, les visées de ces tâches en analyse de sentiments portaient essentiellement sur l'étude de la polarité des sentiments et/ou des "opinions". Nos travaux ont pour objectif de proposer un système d'aide à l'analyse des verbatim d'opinion issus d'entretiens semi-directifs. Par verbatim d'opinion, nous entendons tout énoncé indiquant les opinions et/ou les ressentis des personnes interrogées au sujet d'une thématique présentée dans le cadre d'une conversation. Pour retrouver ces opinions, il est donc important de capturer dans ces transcriptions d'entretiens le contenu subjectif y étant énoncé et le catégoriser. Compte-tenu de notre objectif, nous proposons d'explorer deux domaines de recherche : l'analyse de sentiments et le résumé automatique. Le premier utilise les méthodes en traitement du langage naturel pour automatiser l'extraction et la classification des sentiments dans les documents. Le second permet d'extraire l'information et de condenser l'information pertinente contenue dans les textes pour en produire une synthèse.

Cet article est organisé comme suit : la section 2 présente le problème de l'analyse de sentiments en caractérisant ses approches, la section 3 est dédiée aux types et approches de résumé automatique et dans la section 4, nous présentons une synthèse des deux états de l'art et nous concluons par nos perspectives.

Contribution : Un état de l'art de l'analyse de sentiments et du résumé automatique appliqués aux corpus oraux transcrits pour une aide à l'analyse des verbatim.

2 Analyse de sentiments

L'analyse de sentiments est un sous-domaine de la fouille de textes et se distingue de ses méthodes classiques (qui visent à retrouver des faits) car elle s'intéresse à la détection d'opinions et à leur caractérisation à l'aide d'attributs tels que la polarité, la subjectivité, l'émotion ou encore l'intensité. Lorsque l'on parle d'analyse de sentiments, il s'agit essentiellement d'une tâche de classification. Cette tâche peut suivre trois directions principales. La première consiste à déterminer si un texte est subjectif ou objectif, la deuxième s'attelle à classifier les textes subjectifs en positifs ou négatifs ; enfin, la troisième essaie de déterminer le degré de positivité ou de négativité des textes (Wiebe *et al.*, 2001). Cette tâche peut être appliquée au niveau du texte, ou encore à des niveaux intermédiaires comme le paragraphe, la phrase, ou le mot.

2.1 Approches génériques

Il existe deux approches pour traiter automatiquement le problème de l'analyse d'opinions : les approches fondées sur les lexiques et les approches fondées sur l'apprentissage automatique. Quelques travaux ont également combiné ces deux méthodes.

2.1.1 Approches fondées sur les lexiques

Elles utilisent un lexique de mots, généralement des adjectifs auxquels une polarité est attribuée. Elles ne nécessitent ni de données étiquetées, ni de phases d'entraînement. Les lexiques utilisés dans ces approches peuvent être construits manuellement (Abdulla *et al.*, 2013), semi-automatiquement (Abdaoui *et al.*, 2017) ou en utilisant de grand corpus ou des thésaurus (Goyal & Daumé III, 2011). Pour chaque polarité, un score est assigné par différents algorithmes.

Il existe des lexiques de sentiments qui attribuent un score de -1 aux mots négatifs et +1 aux mots positifs (Mulki *et al.*, 2017). La polarité d'un texte est calculée en cumulant la somme des scores des termes négatifs et positifs. La polarité totale est déterminée en observant le signe de la valeur obtenue. On peut également trouver des dictionnaires où chaque mot est associé à une valence ou intensité pouvant aller de +1 à +5 (Abdulla *et al.*, 2013). Les mots avec une polarité d'intensité +5 sont considérés comme très positifs par rapport aux mots ayant une intensité de +1. On a également des dictionnaires dans lesquels chaque entrée a deux scores de sentiments (positif et négatif) (Mulki *et al.*, 2017). La polarité est calculée en additionnant les scores positifs et les scores négatifs dans le texte. L'orientation du texte est déterminée en observant le score le plus élevé.

Bien qu'il existe plusieurs lexiques de sentiments pour la langue anglaise, nous n'avons connaissance que d'un très petit nombre de lexiques disponibles pour le français :

— Feel [1] (Abdaoui *et al.*, 2017) est un lexique obtenu en traduisant le lexique anglais NRC EmoLex. Il est composé de 14 129 mots. Une polarité (positive ou négative) est attribuée à chaque mot ainsi qu'une émotion parmi les six émotions suivantes : joie, colère, peur, surprise, dégoût et tristesse.
— JeuxDeMots [2] est un lexique polarisé obtenu à travers le jeu en ligne *likeit* (Lafourcade *et al.*, 2015). Le jeu consiste à assigner à chaque mot une des trois polarités disponibles. Un nombre de votes pour chaque polarité est alors généré à partir des réponses des joueurs. De plus, une liste de 21 émotions à attribuer aux mots est également proposée. Les joueurs peuvent également rajouter une nouvelle émotion lorsqu'ils jugent qu'elle est manquante dans la liste préétablie. Cette ressource contient à ce jour 824 434 mots.
— Polarimots [3] (Gala & Brun, 2012) est un lexique de polarité construit en utilisant la deuxième version de POLYMOTS [4], une ressource lexicale qui est composée de 19 009 mots répartis en 2 069 familles morpho-phonologiques. La ressource Polarimots est composée de 7 483 mots dont des noms, verbes, adjectifs et adverbes.
— Le lexique Affect (Augustyn *et al.*, 2006) contient environ 1 300 mots français ainsi que leurs polarités respectives. Construit automatiquement, il inclut également 45 catégories hiérarchiques émotionnelles.

Bien que les méthodes fondées sur l'utilisation de lexiques sont moins complexes à mettre en œuvre et peu coûteuses en temps de calcul, elles sont très limitées. En effet, elles ne tiennent pas compte du contexte (Ding *et al.*, 2008) dans lequel les mots sont employés. Ainsi un mot peut avoir un sens positif dans un certain contexte et un sens négatif dans un autre. C'est le cas, par exemple, du mot ‹‹simple›› qui est négatif lorsqu'il est associé à ‹‹personne›› et positif lorsqu'il est associé à ‹‹solution››.

1. http ://www.lirmm.fr/ abdaoui/FEEL
2. http : www.jeuxdemots.org/jdm-accueil.php
3. http ://polarimots.lif.univ-mrs.fr/
4. http ://polymots.lif.univ-mrs.fr/v2/

2.1.2 Approches fondées sur l'apprentissage automatique

On les regroupe en deux catégories : les méthodes supervisées et les méthodes non-supervisées. Dans cette partie, nous nous sommes davantage focalisés sur les méthodes supervisées.

Les méthodes supervisées sont les plus utilisées pour la tâche de classification. Deux ensembles de données sont nécessaires : les données d'entraînement et les données de test. Les données d'entraînement sont utilisées par un classifieur pour apprendre à caractériser et différencier les données. Les données de test servent quant à elles à évaluer la performance du classifieur sur la tâche de classification. Pour que ces méthodes soient efficaces, elles doivent être apprises sur de grands corpus. Des prétraitements linguistiques (stemming, suppression des mots-vides et ponctuations, lemmatisation, etc.) peuvent également être appliqués aux corpus pour réduire leur densité. Une fois le corpus d'entraînement obtenu, il est non seulement important de sélectionner la technique d'apprentissage à utiliser mais aussi de déterminer les descripteurs. Le choix des descripteurs permet d'avoir une représentation structurelle du document et d'améliorer la performance des systèmes. Parmi les descripteurs les plus utilisés, on retrouve : les n-grammes ou uni-grammes, les catégories morpho-syntaxiques (Parts of speech), les mots d'opinion (opinion words), et la négation. Les algorithmes d'apprentissage les plus utilisés pour la tâche de classification d'opinions sont les modèles Naïfs de Bayes (NB), les machines à vecteurs de support (SVM) ou encore les réseaux de neurones. Ces derniers ont récemment montré de très bonnes performances dans les tâches de détection d'opinions (Gaillat *et al.*, 2018; Moore & Rayson, 2017).

D'après les recherches que nous avons effectuées, l'approche par apprentissage profond est celle qui a été retenue dans la majorité des travaux récents en détection d'opinions dans les corpus écrits tels que les réseaux sociaux (Cliche, 2017; Ali *et al.*, 2018) et les avis de consommateurs (Kooli & Pigneul, 2018; Barhoumi *et al.*, 2018).

Les travaux ayant recours aux méthodes supervisées se fondent uniquement sur un corpus annoté pour effectuer la tâche de classification. Dans la majorité des cas, la construction du corpus se fait manuellement, ce qui est très consommateur en temps et en effort humain.

Les méthodes non-supervisées utilisent moins d'informations que les méthodes supervisées. Elles ne nécessitent pas de données annotées. Ces méthodes ont pour but de regrouper les données par paquets ou *clusters* (Liu & Zhang, 2012). Ces clusters sont ensuite triés en fonction de leur importance et de leur pertinence. Contrairement à l'apprentissage supervisé, l'apprentissage non-supervisé exige beaucoup moins travail et de données.

2.1.3 Approches hybrides

Plusieurs travaux ont exploré les approches hybrides en combinant des méthodes de classification à des techniques utilisées pour la construction de lexiques de sentiments. Dans cette partie, nous n'en présentons que quelques-uns.

Hedar & Doss (2013) proposent un classifieur fondé sur un algorithme SVM pour classer des commentaires en langue arabe sur Facebook. En outre, un lexique de sentiments d'expressions argotiques et idiomatiques (SSWIL [5]) est construit à partir du vocabulaire utilisé dans les commentaires. Leurs expériences montrent que l'utilisation du lexique SSWIL combiné au classifieur améliore en termes

5. Slang Sentimental Words and Idioms Lexicon

de précision la tâche de classification avec 86.86 % au lieu de 75.35 % avec un lexique classique d'opinion. Maurel *et al.* (2008) confirment que la combinaison d'une méthode statistique et d'une méthode symbolique donne des résultats plus précis que les deux méthodes utilisées séparément. Ils proposent le système hybride *SYBILLE* qui combine les deux. La méthode symbolique s'appuie sur un parseur qui analyse le texte et extrait pour chaque phrase les relations de sentiments à l'aide d'un lexique de mots polarisés. Une relation de sentiments a deux arguments : le premier est une expression linguistique qui véhicule un sentiment et le deuxième est l'objet ou la cause du sentiment. Pour la phrase *Je déteste les bonbons*, la relation sera SENTIMENT_POSITIF(détester, bonbons). Une grammaire est appliquée à l'ensemble des textes contenant des opinions et dans un deuxième temps à chaque texte selon la thématique générale qu'il contient. S'agissant de la méthode statistique, ils utilisent une technique d'apprentissage automatique fondée sur la représentation n-gramme de caractères (n = 12). Pour la tâche d'entraînement, les algorithmes SVM et NB sont utilisés. Le système *SYBILLE* prend donc comme entrée la sortie des autres systèmes et calcule selon un indice de confiance pour chaque résultat, une moyenne qui sera traduite en score positif ou négatif. La méthode statistique permet d'obtenir les polarités des messages et la grammaire qui a été configurée peut être modifiée ou améliorée pour l'obtention de meilleurs résultats. L'approche prend la forme d'un cycle itératif ou les résultats sont constamment améliorés.

2.2 Travaux en analyse d'opinions sur des corpus oraux transcrits

Dans la section précédente, nous avons abordé les types d'approches en analyse d'opinion. Dans ce qui suit, nous décrivons les différentes méthodes adoptées pour le traitement de corpus oraux transcrits.

Dans le domaine politique, plusieurs travaux ont abordé le problème de la classification d'opinions à des débats du Congrès américain. Thomas *et al.* (2006) utilisent un modèle de classification supervisée (SVM) pour déterminer si les segments de discours transcrits sont en faveur ou non d'un projet de loi. Ils soulignent que la spécificité du matériau à analyser implique l'existence de plusieurs relations (accord ou désaccord) entre segments de discours appartenant au même débat, qui peuvent être exploitées pour améliorer la classification. De même, ils notent que dans le cadre de débats parlementaires, les orateurs auront tendance à passer plus de temps à défendre leurs arguments en faveur ou en désaccord d'une loi qu'à exprimer leur opinion.

Pour une tâche connexe, Abercrombie & Batista-Navarro (2018) introduisent deux modèles de classification des transcriptions de débats parlementaires (HanDeSet [6]) : le premier effectue une classification de chaque unité du corpus et, dans le deuxième, les discours donnés en réponse à des motions classées positives ou négatives sont classifiées séparément. Pour comparer la performance des deux modèles de classification, ils utilisent deux classifieurs : un classifieur SVM et un perceptron multicouche. Ils évaluent également l'utilisation de représentations n-grammes (uni, bi-, et trigrammes) et d'informations contextuelles (identifiant du débat, l'affiliation politique de l'orateur et le nom du parti soumettant la législation). Les résultats montrent que le deuxième modèle d'analyse associé aux informations contextuelles améliore les résultats pour les deux classifieurs. Ils remarquent également que les informations contextuelles peuvent être très prédictives des résultats du vote des membres.

Pérez-Rosas & Mihalcea (2013) proposent de construire un classifieur de sentiments s'appuyant uniquement sur les descripteurs linguistiques présents dans des transcriptions d'avis oraux. Ils essaient

6. https ://data.mendeley.com/datasets/xsvp45cbt4/2

de déterminer s'il existe une différence significative entre l'analyse de sentiment des avis oraux et des avis écrits. Leurs résultats suggèrent que les avis oraux amènent à une performance égale ou inférieure du classifieur comparée aux avis écrits. Ceci s'explique par le fait que les personnes ont tendance à moins exprimer leurs avis si elles se savent observées ce qui est le cas dans les vidéos d'avis. Ils soulignent par ailleurs une perte de précision lorsque les transcriptions manuelles sont remplacées par les transcriptions automatiques.

Cailliau & Cavet (2010) proposent un modèle de modélisation du déroulement émotionnel de conversations téléphoniques en polarité positive et négative. Premièrement, ils évaluent la dégradation d'un système de détection d'entités sur des transcriptions automatiques. Pour pallier ce problème, ils proposent ensuite un modèle qui s'appuie sur la détection de cinq types d'entités évaluatives. Les entités évaluatives correspondent à des mots (adjectif, verbe, nom) ou expressions indiquant les opinions des interlocuteurs tout au long de la conversation. Cinq des types d'entités qu'ils utilisent proviennent des 12 modalités proposées par Charaudeau (2001) . À chaque type d'entité, une émotion est également ajoutée. Les entités sont également classées en trois polarités et un score d'intensité leur est attribué. Les transcriptions sont découpées en tours de parole et chaque tour de parole reçoit un score positif et négatif en fonction de la somme des poids d'intensité des entités détectées. La somme des poids d'intensité des entités neutres est alors ajoutée au score le plus élevé des polarités positives ou négatives.

Camelin *et al.* (2006) discutent l'impact des différents modèles de langage et de segmentation automatique sur la tâche de classification appliquée à des messages oraux issus d'un service client. Ils utilisent une méthode de classification reposant sur un algorithme SVM. Deux modèles de langage spécifiques aux opinions et de segmentation ont été développés pour les transcriptions automatiques.

— Le premier modèle (REC01) est de type bigramme et est construit sur la base d'un lexique réduit. Les segments pour ce modèle sont obtenus grâce à une segmentation automatique utilisant un automate pour détecter les pauses réalisées par les locuteurs.
— Le deuxième modèle a pour objectif de modéliser uniquement les portions de messages porteuses d'opinions. Un sous-corpus est extrait pour chaque étiquette qui regroupe les segments appartenant à cette étiquette. Pour chaque étiquette et sous-corpus associé, un sous-modèle bigram est calculé. Un autre modèle englobant a aussi été déterminé pour les étiquettes seules. Les parties ne contenant aucune expression d'opinion sont quant à elles modélisées par une boucle de phonèmes en contexte. Le tout est compilé dans un système nommé REC02. Les segments sont eux obtenus à chaque changement du modèle de langage.

Pour la transcription manuelle, les segments sont obtenus manuellement. Deux expériences sont menées en parallèle et deux types de représentation sont choisies (sac de mots et *seeds*). Les mots *seeds* se réfèrent à un lexique contenant un ensemble de mots susceptibles d'exprimer une polarité positive ou négative. Le lexique a été constitué manuellement et chaque mot lemmatisé afin de généraliser la liste. Pour chaque expérience, les modèles sont comparés avec une référence manuelle (REF). La première expérience concerne *la classification des messages sans segmentation* préalable : Les modèles REC01 et REC02 perdent respectivement 6.5 et 7.7 points de précision par rapport à la référence manuelle lorsque le paramètre choisi est le mot. Le rappel lui décroît de 10 points pour le REC01 alors que le REC02 permet de pallier cette perte avec moins de 3 points. Le recours aux *seeds* permet d'améliorer la F-mesure qui passe de 36.5% et 41.6% à 44.4% et 45.4% respectivement pour les deux modèles. La deuxième expérience concerne *la classification avec segmentation* : les résultats en termes de F-mesure sont plus importants que dans la première expérience quelle que soit la représentation choisie. De même, les résultats concernant la classification et la segmentation effectuées sur le corpus REF montrent une amélioration de 20 points pour la F-mesure. L'introduction

de la segmentation dans la construction des modèles de classification et la représentation des messages permet d'améliorer nettement les premiers résultats.

Parmi les corpus oraux transcrits mis à disposition pour la communauté scientifique et utilisés pour la détection d'opinions, on peut citer le corpus AMI (McCowan *et al.*, 2005) qui a été également utilisé pour la tâche de résumé extractif et abstractif, le corpus Switchboard (Godfrey *et al.*, 1992), et le corpus RATP-DECODA (Bechet *et al.*, 2012).

D'après les travaux mentionnés ci-dessus, l'analyse de sentiments dans les corpus oraux transcrits traite de plusieurs aspects :

— La robustesse des systèmes de reconnaissance vocale
— La pertinence des modèles d'opinion
— L'adaptation ou non des méthodes standards de l'écrit à l'oral
— La perte de précision des systèmes de détection d'opinions dans les transcriptions automatiques

Les approches par apprentissage supervisé, notamment les classifieurs, SVM, sont celles qui ont été retenues pour la plupart des travaux. Ces méthodes nécessitent un corpus d'entraînement et de test. Le système apprend à caractériser les données à partir de données déjà annotées pour ensuite les généraliser à des données non annotées. Bien que certains comme Cailliau & Cavet (2010) relèvent la présence de traits linguistiques (les marqueurs de discours, les disfluences, les répétitions, etc.) spécifiques au matériau oral ou encore proposent comme Abercrombie & Batista-Navarro (2018) de tenir compte des informations contextuelles du discours, la majorité des travaux abordent la tâche d'analyse de sentiments sur des données orales comme sur des données écrites. Les conclusions des travaux montrent clairement que le choix des descripteurs joue un rôle crucial dans les résultats obtenus.

3 Le résumé automatique

La section précédente a introduit les approches en analyse de sentiments et les méthodes retenues pour le traitement de corpus oraux transcrits. Dans cette section, nous nous intéressons à la problématique du résumé automatique en décrivant ces approches.

3.1 Les types de résumé automatique

Plusieurs travaux en TAL se sont concentrés sur la tâche du résumé automatique. Les premiers travaux étaient ceux de Luhn (1950). On peut également citer les travaux de Nenkova *et al.* (2011) qui ont proposé une classification des différents types de résumés : le résumé informatif, le résumé indicatif, le résumé orienté, le résumé critique, le résumé générique, le résumé de mise à jour. Le résumé informatif présente un ensemble d'informations renseignant sur le contenu d'un texte et tend à conserver l'organisation générale du texte original ainsi que les principales thématiques. Le résumé indicatif doit pouvoir fournir au lecteur les informations essentielles pour l'aider à choisir de consulter ou non le document source. Pour le résumé orienté, l'objectif est de ne résumer que les informations du document d'entrée qui répondent à une requête spécifique de l'utilisateur. Le résumé critique est un résumé contenant non seulement les informations pertinentes du texte mais également l'analyse critique du document original. Le résumé générique est supposé éviter au lecteur de lire le texte source en présentant de manière succincte la thématique générale du texte. Le résumé de mise à jour

est un résumé multi-document obtenu en condensant un ensemble de documents. Ce résumé suppose que le lecteur a déjà lu ou pris connaissances des documents sources.

En résumé automatique, on distingue deux grandes approches : le résumé par extraction et le résumé par abstraction. Le résumé extractif est le plus répandu, il consiste à extraire les phrases les plus importantes du texte pour former un résumé. Le résumé abstractif consiste à extraire du document source les informations ou notions pertinentes afin de générer automatiquement un nouveau texte. Dans notre revue des méthodes du résumé automatique, nous nous sommes davantage intéressés au résumé extractif mono-document. Le résumé mono-document est un résumé produit à partir d'un document.

3.2 Les approches en résumé automatique

Deux approches sont distinguées en résumé automatique : les méthodes par extraction et les méthodes par abstraction.

3.2.1 Les méthodes par extraction

L'étape la plus cruciale est celle de la sélection des phrases. Ces dernières doivent être pertinentes et non redondantes. Les approches extractives créent un résumé en sélectionnant les phrases les plus importantes du document. Nous présentons maintenant les méthodes les plus utilisées en résumé par extraction :

— Méthodes fondées sur la fréquence des mots : Les travaux de Luhn (1958) ont exploré l'usage de la fréquence lexicale comme indicateur premier de l'importance de la phrase. Comme l'a fait remarquer Luhn (1958), certains mots les plus fréquents dans le document peuvent ne pas être indicatifs de ce qui est important car ils apparaissent souvent dans d'autres documents. L'approche du TF*IDF (Term Frequency * Inverse Document Frequency) a été proposée pour pallier au problème. Radev *et al.* (2004) proposent d'utiliser le Tf*IDF comme indice de pouvoir informatif des unités lexicales dans un système qui combine également une méthode permettant de sélectionner la phrase la moins semblable aux phrases déjà sélectionnées. Les phrases extraites sont celles qui possèdent les mots les plus informatifs du contenu du texte. Edmundson (1969) améliore l'approche de Luhn en définissant de nouvelles caractéristiques pouvant être utilisées pour indiquer l'importance d'une phrase. Ces caractéristiques comprennent la position de la phrase, la présence de mots-clés et de mots du titre, ainsi que d'entités nommées.
— Méthodes fondées sur les graphes : Elles permettent de modéliser le document selon une matrice d'adjacence où chaque nœud représente une phrase et le poids des arêtes entre chaque paire de phrases constitue leur similarité. Une procédure itérative est ensuite utilisée jusqu'à ce que les scores pour les arêtes convergent (Mnasri, 2015; Torres-Moreno *et al.*, 2015). Les travaux de (Radev *et al.*, 2004) ont montré que ces méthodes sont efficaces pour extraire les phrases les plus importantes du texte.
— Méthodes par apprentissage : Il s'agit essentiellement de méthodes supervisées qui visent à apprendre un modèle de choix de phrases à incorporer ou non au résumé. Un classifieur est entraîné à caractériser les phrases qui doivent apparaître dans le résumé et celles qui ne doivent pas l'être. Comme dans le cadre d'une tâche de classification, le classifieur permet de prédire les phrases devant être extraites pour composer le résumé. Ce problème a été soulevé

par Kupiec *et al.* (1995). Leurs travaux démontrent que l'apprentissage automatique combiné aux critères introduits par Luhn et Edmundson donne de bons résultats. Dans le cadre de leurs expérimentations, ils utilisent un classifieur Naïf Bayésien. D'autres ont également intégré les méthodes d'apprentissage automatique dans la tâche de résumé comme Neto *et al.* (2002) qui utilisent également un classifieur bayésien.

3.2.2 Les méthodes par abstraction

La construction d'un résumé par abstraction nécessite trois étapes : la représentation des connaissances, la sélection de ces connaissances et la génération du résumé à partir de ces connaissances (Bossard, 2010). Les méthodes abstractives apparaissent à la fin des années 1970. Le but de ces méthodes est de produire des résumés similaires à ceux que produiraient des humains. On distingue deux étapes en résumé abstractif : la compression du texte et la génération (Mnasri, 2015). La première analyse sémantiquement le contenu du texte soit par extraction, soit par regroupement de phrases. Trois méthodes peuvent être utilisées en résumé abstractif (Salima *et al.*, 2018) : les méthodes qui modélisent les informations contenues dans les textes en ontologies, patrons ou graphes, les méthodes utilisant la représentation sémantique des documents et les méthodes des réseaux de neurones.

3.3 Évaluation automatique du résumé

La tâche d'évaluation des résumés reste très difficile. Il n'existe pas vraiment de résumé de référence idéal pour un document donné compte tenu de la nature subjective de la tâche. Le développement des travaux en résumé automatique ont quand même permis la création de campagnes d'évaluations annuelles (TAC, TREC) afin de tester les performances des systèmes de résumé. Dans cette section, nous présentons quelques métriques d'évaluation des résumés.

3.3.1 Métriques d'évaluations du résumé automatique

— Précision, Rappel et F-mesure : si on considère la tâche de résumé automatique comme un problème de classification binaire des phrases à inclure ou à ne pas inclure dans le résumé final, des métriques d'évaluation comme la Précision, le Rappel et la F-mesure peuvent être utilisées. La Précision correspond au rapport du nombre d'énoncés sélectionnés par le système et présents dans le résumé de référence. Le Rappel est défini par le nombre d'énoncés pertinents retrouvés au regard du nombre d'énoncés pertinents que possède le résumé de référence. La F-mesure est la moyenne harmonique entre la précision et le rappel.
— ROUGE (Lin, 2004) est une métrique qui compare un résumé produit automatiquement à un résumé de référence ou à un ensemble de références (produit par l'homme). Il en existe plusieurs variantes : ROUGE-N, ROUGE-L, ROUGE-W, ROUGE-SU.
— METEOR (Banerjee & Lavie, 2005) est une métrique pour l'évaluation des systèmes de traduction automatique. Elle peut également être utilisée pour la tâche du résumé automatique. Elle utilise la moyenne harmonique de la précision et du rappel des uni-grammes de la phrase. Elle a été proposée pour résoudre certaines faiblesses de la métrique BLEU (Papineni *et al.*, 2002), et également pour produire une bonne corrélation avec le jugement humain au niveau de la phrase ou du segment.

— PYRAMID (Nenkova & Passonneau, 2004) : l'idée derrière cette méthode est que la pertinence d'une unité d'information sémantique nommée SCU (Summary Content Unit) peut être déterminée par le nombre de résumés de référence qui l'incluent. Une SCU est une unité sémantiquement atomique représentant un fait unique, mais n'étant pas liée à sa réalisation lexicale ; deux paraphrases du même fait représentent la même SCU bien qu'elle soit exprimée différemment. Une SCU se voit attribuer un score proportionnel au nombre de résumés de référence qui la contiennent. Le score PYRAMID pour un résumé est calculé en prenant une moyenne normalisée des scores SCUs qu'il contient.

3.4 Travaux récents en résumé automatique appliqués aux corpus oraux transcrits

Historiquement, la tâche du résumé automatique a été appliquée à des données textuelles (articles journalistiques, scientifiques, bandes d'annonce de film, etc.) mais certains travaux se sont intéressés au résumé de conversations parlées transcrites.

Ganesh & Dingliwal (2019) proposent une nouvelle méthode pour le résumé par abstraction de conversations fondée sur les relations et la modélisation discursive. Le modèle proposé comprend l'étiquetage séquentiel des balises de prononciation, le réordonnancement de la conversation pour modéliser les relations discursives, et un modèle de couverture pour le résumé abstractif (*Pointer-Generator, Coverage based model*).

Dans leurs travaux axés sur la génération automatique de comptes-rendus, Alizadeh *et al.* (2018) utilisent la mesure de TF*IDF et une approche de *topic modeling* s'appuyant sur l'algèbre linéaire (NMF [7]) pour la tâche d'extraction d'informations dans des retranscriptions de réunions. Le corpus utilisé est le corpus AMI (McCowan *et al.*, 2005).

Trione (2017) s'intéresse à la tâche du résumé automatique appliquée à des conversations issues des centres d'appels. L'objectif est de produire un résumé contenant les informations essentielles à la compréhension de l'appel. Il propose une méthode de résumé abstractif par remplissage de patrons. Un ensemble de patrons de phrases est généré automatiquement à partir d'un corpus d'entraînement constitué de paires de transcriptions de conversations/résumés. Ensuite, les patrons sont sélectionnés en fonction des variables qui ont été détectées dans le document traité. Le résumé est généré par remplissage de ces patrons. Les conversations utilisées sont issues du corpus RATP-DECODA (Bechet *et al.*, 2012). Le corpus est composé de plus de 1500 conversations. Les versions audio et textuelle (transcriptions manuelles et automatiques) sont également disponibles. Les résultats montrent clairement que les transcriptions manuelles obtiennent de meilleurs résultats que les transcriptions automatiques. En outre, les résultats détaillés confirment que la méthode proposée par (Trione, 2017) produit de meilleurs résultats que les approches extractives (méthode MMR [8]) et les méthodes fondées sur l'utilisation de patrons figés (patrons écrits manuellement).

Dans leurs travaux, Wang & Liu (2015) se sont intéressés au résumé d'opinions appliqué à des conversations spontanées de locuteurs. Ils explorent deux méthodes. La première est une méthode non-supervisée fondée sur les graphes et la deuxième est une méthode supervisée : ils choisissent d'utiliser un classifieur SVM. La méthode fondée sur les graphes qu'ils utilisent incorpore la similarité cosinus entre deux phrases (*sim (s, v)*), la pertinence du sujet (*REL*), et le score de sentiment de

7. Non-negative Matrix Factorization
8. Maximal Marginal Relevance

la phrase (*s* : probabilité qu'une phrase contienne une opinion). En plus de ces trois scores, ils introduisent de nouvelles relations entre phrases pour modéliser la structure du dialogue. Pour la construction du classifieur SVM, ils utilisent quatre descripteurs : les trois premiers sont ceux utilisés pour l'approche fondée sur les graphes et le dernier prend en compte la longueur de la phrase. Ils décident également d'évaluer l'apport d'un module de résolution de coréférence et d'anaphore pronominale dans la production du résumé en introduisant trois nouveaux descripteurs : le contexte de référence, la pertinence de la référence (*REL*) et la chaîne de coréférence. La pertinence (*REL*) correspond au nombre de syntagmes nominaux qui renvoient à la même entité. Ils y ajoutent également un module de post-traitement qui remplace les pronoms par les unités lexicales auxquelles ils se réfèrent après sélection des phrases à introduire au résumé. Ils comparent ce système qu'ils nomment *SYS_PRON* à deux autres systèmes :

— une première baseline qui intègre les méthodes supervisées décrites ci-dessus.
— une deuxième baseline similaire à la première et intégrant le module de post-traitement.

Étant donné que les résumés extractifs de référence sont extraits des conversations originales contenant des pronoms, Wang & Liu (2015) proposent de les comparer avec les résumés abstractifs de référence. Les résultats montrent que le système *SYS_PRON* obtient les meilleurs résultats par rapport aux deux baselines démontrant ainsi que la prise en compte de la coréférence et de l'anaphore améliore les résumés produits automatiquement.

Zhu & Penn (2006) proposent un système de résumé automatique par extraction reposant sur une variété de descripteurs : le score MMR, les caractéristiques lexicales, structurelles, prosodiques et la présence de disfluences (les faux-départs, les phrases répétées ou recommencées, les syllabes répétées, les tics de langage, tels que les grommellements, propos non lexicaux tels que « euh », « hum », « hem »). Pour obtenir un module de sélection d'énoncés qui peut être entraîné et qui peut utiliser différents descripteurs, ils opèrent une classification binaire standard. Deux classifieurs sont choisis : le SVM et LR (Régression logistique). Ils utilisent à la fois les transcriptions automatiques et manuelles du corpus *Switchboard* (Godfrey *et al.*, 1992).

Furui (2005) propose une méthode de résumé en deux étapes consistant en une extraction de phrases importantes et une compression des phrases à partir de mots. Après avoir enlevé tous les *fillers* [9] en se basant sur les résultats de la transcription automatique, un ensemble de phrases relativement importantes est extrait et une compression de phrases est appliquée à l'ensemble des phrases extraites. Le ratio d'extraction et de compression des phrases est contrôlé en fonction d'un ratio de compression initialement déterminé par l'utilisateur. Les unités de phrases et de mots sont extraites des résultats de la reconnaissance vocale et concaténées pour produire des résumés afin de maximiser la somme pondérée de la probabilité linguistique (*linguistic likelihood*), de la quantité d'information, de la mesure de confiance et de la probabilité grammaticale (*grammatical likelihood*) des unités concaténées. La méthode proposée a été appliquée pour résumer les journaux télévisés. La méthode a également été évaluée au moyen de mesures objectives et subjectives. Ces évaluations confirment que la méthode proposée est efficace pour la production de résumé.

En conclusion, les travaux en résumé automatique appliqués aux corpus oraux transcrits se sont focalisés sur les données de conversations téléphoniques (Ganesh & Dingliwal, 2019), de journaux télévisés (Furui, 2005) ou de réunions (Alizadeh *et al.*, 2018). Les premiers systèmes de résumé automatique de conversations exploitaient les méthodes extractives. Néanmoins, les méthodes abstractives énoncées ci-dessus fournissent elles aussi des résultats satisfaisants. Les transcriptions manuelles semblent se rapprocher des textes écrits, ce qui expliquerait la performance élevée des systèmes de

9. tic de langage

résumé automatique extractif alors que la présence d'erreurs liées à la reconnaissance vocale, de disfluences, d'hésitations, de répétitions, et de bruit dans les transcriptions automatiques sont autant d'éléments qui réduisent la précision des systèmes.

4 Conclusions et perspectives

Nous avons présenté deux domaines de recherche appliqués aux corpus oraux transcrits : l'analyse de sentiments et le résumé automatique. S'agissant de ces deux domaines de recherche, les travaux présentés ci-dessus se sont intéressés à divers types de données orales transcrites. À la différence des textes écrits, les conversations sont des échanges spontanés entre personnes. Ces échanges peuvent être altérés par la spontanéité des locuteurs et contenir du bruit. Nous étions particulièrement intéressés de savoir comment les auteurs traitaient les problèmes de l'oral pour ce type de données, les méthodes qu'ils proposaient ou mettaient en avant, et les traitements qu'ils préconisaient.

— *S'agissant du traitement des problèmes de l'oral dans les corpus*, il pose spécifiquement problème pour les transcriptions automatiques car elles sont plus sujettes à des erreurs de transcription, à la présence d'hésitations et de répétitions que les transcriptions manuelles qui peuvent être révisées ;
— *Au niveau du choix des descripteurs pour la construction des classifieurs*, d'autres caractéristiques telles que les informations contextuelles ou méta-données, les pronoms, les relations entre phrases de locuteurs différents ont été mises avant. Les résultats détaillés des articles ont montré qu'ils étaient cruciaux dans l'amélioration des systèmes de classification et d'extraction de phrases ;
— *Au niveau du choix des méthodes*, les méthodes supervisées sont les plus utilisées pour l'analyse de sentiments. Les méthodes extractives pour le résumé automatique sont privilégiées même si de nombreuses approches sur la compression automatique de phrases ont été mises en avant.

Au regard de ces deux états de l'art, nous estimons que nos travaux se différencient d'eux au niveau du matériau à analyser. Les données dont nous disposons sont des transcriptions manuelles d'entretiens semi-directifs non-révisées. Ce type de données est très peu étudié en TAL mais très présent dans le domaine des sciences humaines et sociales. Ces transcriptions se veulent plus fidèles à l'audio et intègrent donc les disfluences qui posent généralement des difficultés pour les systèmes de reconnaissance vocale. Il serait intéressant de se pencher sur des travaux qui se seraient intéressés à l'utilisation des disfluences dans la détection d'opinions ou le résumé automatique comme Zhu & Penn (2006).

Nous relevons aussi que très peu de travaux se sont intéressés au résumé d'opinions d'entretiens semi-directifs. Bien que les travaux de Wang & Liu (2015) ont porté sur le résumé d'opinion dans des conversations et ont permis de valoriser l'apport de l'analyse anaphorique dans la tâche du résumé, elles concernaient des conversations parlées issues de centre d'appels. Il serait également intéressant de pousser nos recherches sur les modules de résolution de coréférence et d'anaphores. Par ailleurs, nous souhaitons également nous tourner vers les travaux en résumé orienté et/ou guidé par la tâche. Les entretiens semi-directifs impliquent un échange entre deux personnes (la personne interrogée et la personne qui interroge). Cela permettrait d'évaluer la capacité de ces systèmes à condenser et à extraire les connaissances pertinentes contenues dans ce type de données.

Références

ABDAOUI A., AZÉ J., BRINGAY S. & PONCELET P. (2017). Feel : a french expanded emotion lexicon. *Language Resources and Evaluation*, **51**(3), 833–855.

ABDULLA N. A., AHMED N. A., SHEHAB M. A. & AL-AYYOUB M. (2013). Arabic sentiment analysis : Lexicon-based and corpus-based. In *Applied Electrical Engineering and Computing Technologies (AEECT), 2013 IEEE Jordan Conference on*, p. 1–6 : IEEE.

ABERCROMBIE G. & BATISTA-NAVARRO R. (2018). 'aye'or'no' ? speech-level sentiment analysis of hansard uk parliamentary debate transcripts. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*.

ALI C. B., MULKI H. & HADDAD H. (2018). Impact du prétraitement linguistique sur l'analyse des sentiments du dialecte tunisien. In *Actes de la conférence Traitement Automatique de la Langue Naturelle, TALN 2018*, p. 383.

ALIZADEH P., CELLIER P., CHARNOIS T., CRÉMILLEUX B. & ZIMMERMANN A. (2018). Étude expérimentale d'extraction d'information dans des retranscriptions de réunions. In *Traitement automatique du langage naturel (TALN)*.

AUGUSTYN M., HAMOU S. B., BLOQUET G., GOOSSENS V., LOISEAU M. & RINCK F. (2006). Lexique des affects : constitution de ressources pédagogiques numériques. In *Colloque International des étudiants-chercheurs en didactique des langues et linguistique.*

BANERJEE S. & LAVIE A. (2005). Meteor : An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, p. 65–72.

BARHOUMI A., CAMELIN N. & ESTÈVE Y. (2018). Des représentations continues de mots pour l'analyse d'opinions en arabe : une étude qualitative. In *Actes de la conférence Traitement Automatique de la Langue Naturelle, TALN 2018*, p. 215.

BECHET F., MAZA B., BIGOUROUX N., BAZILLON T., EL-BEZE M., DE MORI R. & ARBILLOT E. (2012). Decoda : a call-centre human-human spoken conversation corpus. In *LREC*, p. 1343–1347.

BOSSARD A. (2010). *Contribution au résumé automatique multi-documents*. PhD thesis, Université Paris-Nord-Paris XIII.

CAILLIAU F. & CAVET A. (2010). Analyse des sentiments et transcription automatique : modélisation du déroulement de conversations téléphoniques. *Revue Traitement Automatique des Langues*, **51**(3), 131–154.

CAMELIN N., DAMNATI G., BÉCHET F. & DE MORI R. (2006). Détection automatique d'opinions dans des corpus de messages oraux. *Journées d'Etude sur la Parole (JEP'06), Dinard*.

CHARAUDEAU P. (2001). *Grammaire du sens et de l'expression*. Hachette.

CLICHE M. (2017). Bb_twtr at semeval-2017 task 4 : Twitter sentiment analysis with cnns and lstms. *arXiv preprint arXiv :1704.06125*.

COLLINS E., AUGENSTEIN I. & RIEDEL S. (2017). A supervised approach to extractive summarisation of scientific papers. *arXiv preprint arXiv :1706.03946*.

DING X., LIU B. & YU P. S. (2008). A holistic lexicon-based approach to opinion mining. In *Proceedings of the 2008 international conference on web search and data mining*, p. 231–240 : ACM.

EDMUNDSON H. P. (1969). New methods in automatic extracting. *Journal of the ACM (JACM)*, **16**(2), 264–270.

FURUI S. (2005). Spontaneous speech recognition and summarization. In *The Second Baltic Conference on Human Language Technologies*, p. 39–50.

GAILLAT T., SOUSA A., ZARROUK M. & DAVIS B. (2018). Finsentia : Sentiment analysis in english financial microblogs. In *Actes de la conférence Traitement Automatique de la Langue Naturelle, TALN 2018*, p. 271.

GALA N. & BRUN C. (2012). Propagation de polarités dans des familles de mots : impact de la morphologie dans la construction d'un lexique pour l'analyse d'opinions. In *19ème Conférence sur le Traitement Automatique des Langues Naturelles (TALN'2012)*, volume 2, p. 495–502.

GANESH P. & DINGLIWAL S. (2019). Abstractive summarization of spoken and written conversation. *arXiv preprint arXiv :1902.01615*.

GODFREY J. J., HOLLIMAN E. C. & MCDANIEL J. (1992). Switchboard : Telephone speech corpus for research and development. In *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on*, volume 1, p. 517–520 : IEEE.

GOYAL A. & DAUMÉ III H. (2011). Generating semantic orientation lexicon using large data and thesaurus. In *Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis*, p. 37–43 : Association for Computational Linguistics.

HEDAR A. R. & DOSS M. (2013). Mining social networks arabic slang comments. In *IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*.

KOOLI N. & PIGNEUL E. (2018). Analyse de sentiments à base d'aspects par combinaison de réseaux profonds : application à des avis en français. In *conférence sur le Traitement Automatique des Langues Naturelles (TALN)*, volume 1, p. 303–310.

KUPIEC J., PEDERSEN J. & CHEN F. (1995). A trainable document summarizer. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, p. 68–73 : ACM.

LAFOURCADE M., LE BRUN N. & JOUBERT A. (2015). Vous aimez ?... ou pas ? likeit, un jeu pour construire une ressource lexicale de polarité. In *TALN : Traitement Automatique des Langues Naturelles*.

LIN C.-Y. (2004). Rouge : A package for automatic evaluation of summaries. *Text Summarization Branches Out*.

LIU B. & ZHANG L. (2012). A survey of opinion mining and sentiment analysis. In *Mining text data*, p. 415–463. Springer.

LUHN H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of research and development*, **2**(2), 159–165.

MAUREL S., CURTONI P. & DINI L. (2008). A hybrid method for sentiment analysis. In *INFORSID*.

MCCOWAN I., CARLETTA J., KRAAIJ W., ASHBY S., BOURBAN S., FLYNN M., GUILLEMOT M., HAIN T., KADLEC J., KARAISKOS V. *et al.* (2005). The ami meeting corpus. In *Proceedings of the 5th International Conference on Methods and Techniques in Behavioral Research*, volume 88, p. 100.

MNASRI M. (2015). Resumé automatique multi-document dynamique : État de l'art. *Actes des 17e Rencontres des Etudiants Chercheurs en Informatique pour le Traitement Automatique des Langues, Association pour le Traitement Automatique des Langues, Caen*, p. 38–49.

MOORE A. & RAYSON P. (2017). Lancaster a at semeval-2017 task 5 : Evaluation metrics matter : predicting sentiment from financial news headlines. *arXiv preprint arXiv :1705.00571*.

MULKI H., HADDAD H. & GRIDACH M. (2017). Polarity analysis of non figurative tweets : Tw-star participation on deft 2017. In *24e Conférence sur le Traitement Automatique des Langues Naturelles (TALN)*, p.9̃2.

NENKOVA A., MCKEOWN K. *et al.* (2011). Automatic summarization. *Foundations and Trends® in Information Retrieval*, **5**(2–3), 104–141.

NENKOVA A. & PASSONNEAU R. (2004). Evaluating content selection in summarization : The pyramid method. In *Proceedings of the human language technology conference of the north american chapter of the association for computational linguistics : Hlt-naacl 2004*.

NETO J. L., FREITAS A. A. & KAESTNER C. A. (2002). Automatic text summarization using a machine learning approach. In *Brazilian Symposium on Artificial Intelligence*, p. 205–215 : Springer.

NZALI M. D. T., ABDAOUI A., AZÉ J., BRINGAY S., LAVERGNE C., MOLLEVI C. & PONCELET P. (2017). Frenchsenticlass : un système automatisé pour la classification de sentiments en français. In *Actes de l'atelier DEFT de la conférence TALN 2017*.

PAPINENI K., ROUKOS S., WARD T. & ZHU W.-J. (2002). Bleu : a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, p. 311–318 : Association for Computational Linguistics.

PÉREZ-ROSAS V. & MIHALCEA R. (2013). Sentiment analysis of online spoken reviews. In *INTERSPEECH*, p. 862–866.

POLAND B. D. (1995). Transcription quality as an aspect of rigor in qualitative research. *Qualitative inquiry*, **1**(3), 290–310.

PONTIKI M., GALANIS D., PAPAGEORGIOU H., ANDROUTSOPOULOS I., MANANDHAR S., MOHAMMAD A.-S., AL-AYYOUB M., ZHAO Y., QIN B., DE CLERCQ O. *et al.* (2016). Semeval-2016 task 5 : Aspect based sentiment analysis. In *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, p. 19–30.

RADEV D. R., JING H., STYŚ M. & TAM D. (2004). Centroid-based summarization of multiple documents. *Information Processing & Management*, **40**(6), 919–938.

SALIMA L., SAID O. E. A. & BERNARD E. (2018). Resumé automatique guidé de textes : Etat de l'art et perspectives. *Actes des 20e Rencontres des Etudiants Chercheurs en Informatique pour le Traitement Automatique des Langues, Association pour le Traitement Automatique des Langues, Rennes*.

THOMAS M., PANG B. & LEE L. (2006). Get out the vote : Determining support or opposition from congressional floor-debate transcripts. In *Proceedings of the 2006 conference on empirical methods in natural language processing*, p. 327–335 : Association for Computational Linguistics.

TORRES-MORENO J.-M., RAMIREZ J. & DA CUNHA I. (2015). Un résumeur à base de graphes, indéependant de la langue. *arXiv preprint arXiv :1501.01243*.

TRIONE J. (2017). *Méthodes par abstraction et extraction pour le résumé de conversations orales issues de centres d'appels téléphoniques*. PhD thesis, Aix-Marseille.

WANG D. & LIU Y. (2015). Opinion summarization on spontaneous conversations. *Computer Speech & Language*, **34**(1), 61–82.

WIEBE J., WILSON T. & BELL M. (2001). Identifying collocations for recognizing opinions. In *Proceedings of the ACL-01 Workshop on Collocation : Computational Extraction, Analysis, and Exploitation*, p. 24–31.

ZHU X. & PENN G. (2006). Summarization of spontaneous conversations. In *Ninth International Conference on Spoken Language Processing*.

Les systèmes de dialogue orientés-but : état de l'art et perspectives d'amélioration

Léon-Paul Schaub[1, 2, 3] Cyndel Vaudapiviz[4]
(1) Akio, 43, rue de Dunkerque, 75010 Paris, France
(2) LIMSI-CNRS, Campus Universitaire bâtiment 507, Rue John Von Neumann, 91400 Orsay, France
(3) Université-Paris-Saclay, 15 Rue Georges Clemenceau, 91400 Orsay, France
schaub@limsi.fr
(4) ER-TIM INALCO 2, rue de Lille 75007 Paris, France cyndel.vaudapiviz@gmail.com

RÉSUMÉ

La gestion et la sélection des informations pertinentes pour un tour de parole donné restent un problème pour les systèmes de dialogue à domaine ouvert. Pour ces derniers, les interactions possibles entre un utilisateur et un agent sont a priori infinies et indéfinies. La possibilité d'une réponse erronée de l'agent à l'utilisateur demeure donc élevée. Pour les systèmes orientés-but, le problème est considéré comme résolu, mais d'après notre expérience aucun système ne montre une robustesse remarquable lorsqu'il est évalué en situation réelle. Dans cet article, nous dressons un état de l'art des méthodes d'apprentissage de l'agent et des différents modèles d'agent conversationnel. Selon nous, l'une des pistes d'amélioration de l'agent réside dans sa mémoire, car cette dernière (souvent représentée par le triplet : tour de parole courant, historique du dialogue et base de connaissances) n'est pas encore modélisée avec assez de précision. En dotant l'agent d'un modèle de mémoire d'inspiration cognitive, nous pensons pouvoir augmenter les performances d'un système de dialogue orienté-but en situation réelle, par l'emploi d'algorithmes d'apprentissage automatique avec une approche antagoniste en support d'un nouveau modèle de mémoire pour l'agent.

ABSTRACT

Goal-oriented dialog systems : a recent overview and research prospects

Determining which information is relevant for a given dialog turn is still a problem for open-domain dialog systems since the interaction between a user and the system is a priori infinite and indefinite. Thus, the possibility for error remains high for the dialog system. Although the problem is considered solved, to our experience, no system has shown outstanding performance when confronted to "real-world" situation. We review the state-of-the-art of machine learning approaches used for dialog system development and present the various models of agent architecture. We claim that one of the main directions for improvement is the agent architecture. Often the agent's memory (represented by the triple : dialog turn, dialogue history and knowledge base) is not modeled accurately enough. Thus we decided to investigate more cognitive oriented models of memory. We hypothesize that combining machine learning approaches in an adversarial setup with a new memory model for the agent would result in improved performances for goal-oriented dialog systems.

MOTS-CLÉS : systèmes de dialogue, apprentissage par renforcement, modèle de mémoire, apprentissage par antagonisme, systèmes orientés-buts.

KEYWORDS: dialog systems, reinforcement learning, memory model, adversarial learning, goal-oriented systems.

1 Introduction

La tâche de traitement automatique du langage qui nous intéresse est celle des agents conversationnels. Dans ce domaine, plusieurs systèmes de conversation récemment industrialisés sont désormais célèbres : Siri, Cortana, Alexa (Hoy, 2018) IBM Watson (Ferrucci *et al.*, 2010).
On peut distinguer deux types de systèmes de dialogue (Rosset, 2018; Jurafsky & Martin, 2018; Thakkar *et al.*, 2018; Kowtko *et al.*, 1993) : les systèmes sociaux et les systèmes orientés-but. Les systèmes sociaux servent d'interlocuteur, ils ne possèdent pas de but précis. Ils sont évalués sur leur capacité à prolonger l'interaction autant que possible tout en préservant l'illusion de communiquer avec une personne réelle (Turing, 1950; Pinar Saygin *et al.*, 2000; Russell *et al.*, 2010).
Les systèmes orientés-but, quant à eux, servent d'assistant et de ce fait ne sont jugés que sur leur capacité à aider correctement l'humain dans une tâche, par exemple la recherche d'information (Hiroya *et al.*, 1997), l'aide à la réservation (Ilievski *et al.*, 2018) ou encore l'accès aux services commerciaux (p.e. e-shopping ou livraison à domicile) (Yan *et al.*, 2017) .
La problématique qui se pose alors réside dans la nature des données. En effet, les systèmes orientés-but restent surtout utilisés dans un cadre industriel et donc commercial, ce qui implique en général l'utilisation des données propriétaires pour construire le système. De ce fait, ces données sont en général uniquement exploitables par leurs possesseurs. Par conséquent, les systèmes de dialogue deviennent opaques et rigides : opaques car on n'a pas accès aux données, et rigides car généralement ces systèmes-là ne sont pas évalués sur des données extérieures à leur environnement de développement.
Une solution à ce problème de manque de souplesse et de limitation est apparue après la redécouverte des réseaux de neurones (Rosenblatt, 1958) remis au goût du jour par (Lecun, 1985). Les améliorations des machines computationnelles (Aslot *et al.*, 2001) ont rendu possible l'utilisation des réseaux de neuronnes profonds dans des temps de calcul raisonnables. En effet, avec les progrès technologiques dans les domaines de l'information et de communication, il existe une quantité immense de données accessibles, non structurées, en langue naturelle (réseaux sociaux, blogs, forums, médias...). La fouille de ces données permet d'en extraire automatiquement et à faible coût des connaissances, par exemple dans les réseaux sociaux pour détecter les *fake news* ou la propagande (Grabar & Richey, 2017), ou bien encore dans les medias pour générer automatiquement des résumés d'articles (Lamsiyah *et al.*, 2018). La disponibilité de cette masse de données combinée au potentiel d'apprentissage des réseaux de neurones rend maintenant possible d'accroître le pouvoir de généralisation d'un modèle de langue dans une mesure bien supérieure à ce qui était atteignable avec une approche symbolique classique.
Nous nous intéressons dans cet article aux systèmes de dialogue orientés-but pour le langage écrit, dont le modèle est décrit dans (Kölzer & Ulm, 1999). L'illustration de la Figure 1 présente une architecture-type pour ce genre de système, qui possède trois fonctions principales : la compréhension du langage (analyse, *parsing*), la gestion du dialogue (recherche d'information, prise de décision) et la génération de réponse (génération de langue naturelle). Ces trois composantes sont dépendantes les unes des autres.
Tous les travaux de recherche menés jusqu'à aujourd'hui s'inspirent de ce modèle pour construire un système de dialogue. Ceci nécessite la modélisation du dialogue et donc l'analyse de conversations dans une interaction homme-homme. Selon Clark et Marshall, *"les participants accumulent des informations à l'intérieur d'un espace de connaissances mutuelles. Les connaissances mutuelles correspondent aux connaissances que chacun sait être communes."* (Clark & Marshall, 1981). Ils expliquent que chaque interlocuteur d'un dialogue maîtrise non pas un triplet mais un quintuplet d'éléments pour mener à bien la conversation. Aux trois éléments fondamentaux d'un dialogue (tour de parole, historique, connaissance), il faut ajouter l'environnement physique ou virtuel dans lequel

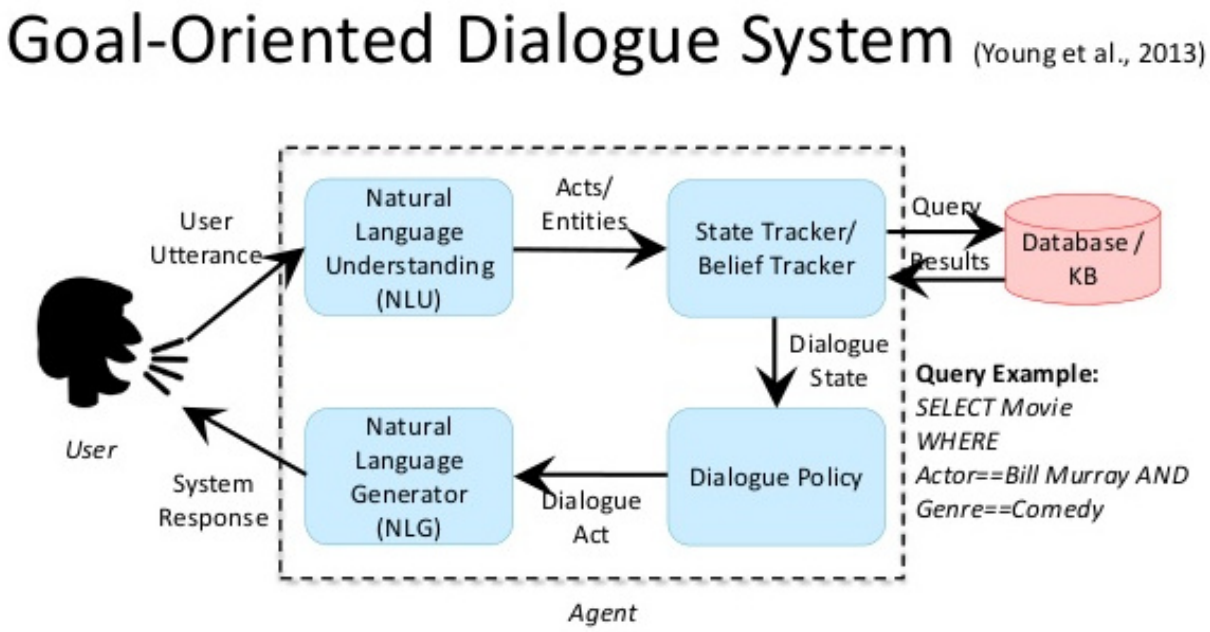


FIGURE 1 – Schéma typique du système de dialogue

se déroule le dialogue, c'est-à-dire l'ensemble des informations internes à chaque interlocuteur (son propre état interne, l'interprétation qu'il fait du possible état interne des autres (Molder & Potter, 2005)), mais également les informations qui lui sont externes (contexte spatio-temporel). Il faut également ajouter l'expérience, que les auteurs considèrent différente de la base de connaissances, car l'expérience (le vécu) sert, d'après eux, de passerelle commune entre les tours de parole et la base de connaissances (la culture).

Lorsque l'on fait un parallèle entre l'environnement dialogique habituellement utilisé dans une interaction et la mémoire humaine, on se rend compte que dans le modèle des triplets d'informations, la mémoire humaine est mise en jeu sous deux formes différentes : le contexte de dialogue correspond à la mémoire à court terme (Stafford *et al.*, 2006), alors que la base de connaissances est associée à la mémoire à long terme (Gobet, 2000). Ce parallèle entre mémoire humaine et représentations dialogiques est explicité dans la section 3.

Avant cela, dans la section 2 nous faisons un état de l'art des techniques d'apprentissage pour un agent conversationnel orienté-but. Puis, nous donnons un aperçu des modèles de conversation les plus performants. Nous dressons ensuite un récapitulatif des différentes méthodes d'évaluation existantes pour ce type de système de dialogue. L'évaluation des systèmes (Lowe *et al.*, 2017; Liu *et al.*, 2016a; Cercas Curry *et al.*, 2017) reste un problème épineux du domaine. Enfin, nous décrivons les implémentations industrielles et commerciales de cette technologie. Dans la section 4, nous présentons le modèle d'apprentissage antagoniste, qui sert à la fois de méthode d'apprentissage et d'évaluation. Nous expliquons pourquoi nous pensons que l'utilisation de cette technique appliquée au modèle de mémoire explicité en section 3, est une piste méritant d'être suivie.

Enfin, dans la conclusion en section 5, nous proposons plusieurs perspectives de recherche et de développement liés à l'amélioration des systèmes de dialogue orientés-but en situation réelle.

2 Les systèmes de dialogue orientés-but

2.1 Tour d'horizon

Depuis la fin des années 2000, l'industrie a commencé à entrevoir le potentiel d'un système de dialogue pour une tâche précise. Les recherches scientifiques se sont alors concentrées sur la question et plusieurs types de systèmes ont été construits. Les plus performants sont ceux basés sur l'apprentissage profond. (Wang, 2018) propose un état de l'art des systèmes orientés-but depuis les origines à aujourd'hui. Il explique que les premiers systèmes ont vu le jour dans les années 1960 avec le *chatbot* (agent conversationnel) ELIZA, un système uniquement basé sur les règles et la reformulation de questions (Weizenbaum, 1983), puis PARRY (Kenneth Mark Colby, 1974), un *chatbot* simulant un patient schizophrène.
Il décrit ensuite les systèmes basés sur la récupération d'information (*retrieval-based*) qui se servent d'anciennes réponses pour répondre à une entrée de l'utilisateur, à l'image du système de Watson (Bartl & Spanakis, 2017; Oddy, 1977; Misu & Kawahara, 2008; Ferrucci *et al.*, 2010; Young *et al.*, 2018). Cependant, une récente expérience menée par Schaub a montré que pour un même corpus et une même tâche de dialogue, un système de dialogue créé avec des outils libres de droit comme scikit-learn (Buitinck *et al.*, 2013) ou TensorFlow (Abadi *et al.*, 2015) plutôt qu'un système commercial réputé pouvaient obtenir de meilleurs résultats que IBM Watson lors de l'évaluation humaine des deux systèmes (Schaub, 2017). Ces techniques, état de l'art avant que n'explosent les méthodes d'apprentissage profond, sont depuis peu concurrencées par les systèmes de génération de réponse. Dans ce cas, la sortie de l'agent n'est pas une réponse existante, mais elle est créée en fonction de l'entrée de l'utilisateur. (Wang, 2018) explique ensuite que les méthodes neuronales qui fonctionnent le mieux sont basées sur le modèle des réseaux récurrents (RNN)(David E. Rumelhart *et al.*, 1988), en particulier les réseaux de mémoire à long-court terme (LSTM)(Hochreiter & Schmidhuber, 1997) et l'apprentissage par renforcement (RL) (Richard S. Sutton & Andrew G. Barto, 1998) comme dans (Kearns *et al.*, 2011; Walker, 2011; Weisz *et al.*, 2018; Schatzmann *et al.*, 2006). Récemment, des méthodes d'apprentissage antagoniste (Borodin & El-Yaniv, 1998) et les réseaux de neurones antagonistes (GAN) (Goodfellow *et al.*, 2014) pour de la génération de dialogue affichent aussi des résultats encourageants par rapport aux scores de référence (Li *et al.*, 2017a; Efstathiou & Lemon, 2014; Liu & Lane, 2018). Nous revenons en détails sur cette méthode dans la section 4.
Dans un autre état de l'art, (Chen *et al.*, 2017) rejoint (Wang, 2018) sur le fait que les méthodes LSTM et RL donnent les meilleurs résultats. Il ajoute que l'apprentissage actif du dialogue (AL) (Asghar *et al.*, 2016) donne aussi de très bons résultats car l'agent apprend grâce au retour de l'utilisateur. Il s'intéresse également aux méthodes les plus performantes pour chacun des modules du dialogue (compréhension, gestion/décision et génération). En ce qui concerne la compréhension du dialogue, ce sont les réseaux de neurones à convolution (CNN)(LeCun *et al.*, 2015) qui fonctionnent le mieux, comme dans (Baradaran Hashemi, 2016; Shen *et al.*, 2014; Huang *et al.*, 2013), car il s'agit d'une tâche de détection d'intention et d'entités (Jiang, 2012; Xiao & Cho, 2016; Lai *et al.*, 2015). Pour la gestion du dialogue, il explique que les RNN sont les plus performants lors de l'évaluation par le Dialog State Tracking Challenge (DSTC) (Williams *et al.*, 2014) sur lequel nous revenons dans la section 2.3. Enfin, pour la génération, il indique que les meilleurs systèmes sont basés sur les LSTM (Hu *et al.*, 2017). Récemment, des systèmes faisant du tout-en-un (qui ne séparent pas les trois fonctions d'une architecture classique de système de dialogue) obtenaient des résultats très intéressants, montrant là l'interdépendance des trois fonctions en les traitant comme une seule unité, avec une seule entrée et une seule sortie par tour de parole. Ce sont les systèmes *end-to-end* (Li *et al.*,

2017b; Young *et al.*, 2018; Wen *et al.*, 2016). (voir figure 2)
Pour le problème de l'indécision (comment l'agent choisit la réponse à donner lorsqu'il y'a plusieurs réponses possibles ou qu'il n'y a aucune réponse), Young utilise le processus de décision markovienne partiellement observable (POMDP) pour optimiser le choix de l'agent au moment de fournir la réponse à l'utilisateur (Young *et al.*, 2010) . Les POMDP ont été récemment utilisés dans les travaux de (Young *et al.*, 2013a,b) avec des résultats satisfaisants. Dès 2006, (Young, 2006) montre que les POMDP est un bon moyen de modéliser un système de dialogue en raison de la fonction d'évaluation inhérente au processus. En effet, contrairement à un processus markovien classique, où l'agent a une connaissance déterministe de l'état courant (comme je suis à gauche je vais à droite), dans le processus partiellement observable, l'agent a une croyance (*belief)* sur l'état dans lequel il doit aller, et donc dans lequel il se trouve. Cela contraint l'agent à se représenter l'historique des transitions dans son automate et donc à en garder une trace en mémoire. Dans un dialogue, cela revient à modéliser la croyance du locuteur sur l'interprétation que fait celui qui écoute de sa propre élocution. Plus récemment encore, des systèmes

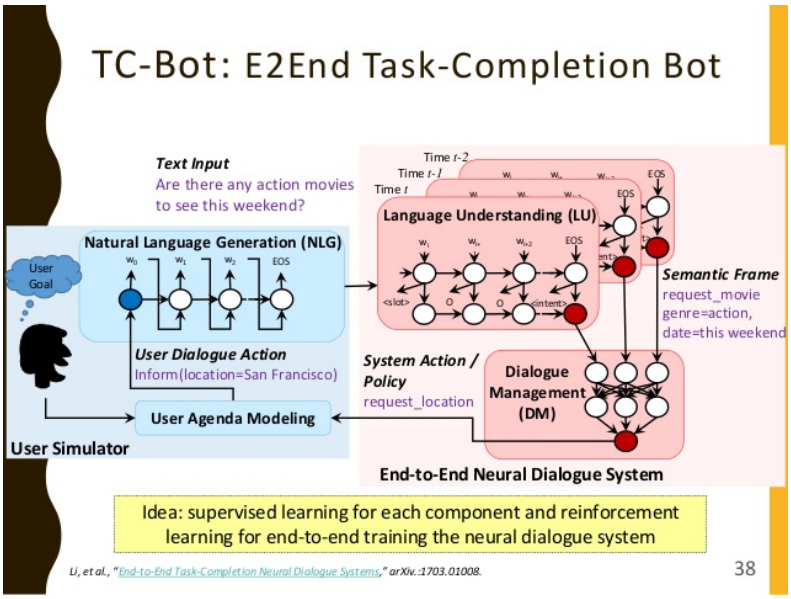


FIGURE 2 – Modèle de système "end-to-end"

comme celui du projet LIHLITH (Agirre *et al.*, 2018) utilisent l'apprentissage continu (lifelong learning) pour que l'agent conversationnel apprenne non seulement en phase de développement mais aussi en production. Contrairement à l'AL qui nécessite l'intervention humaine pour entraîner l'agent, dans l'apprentissage continu c'est un agent conversationnel qui donne lui-même un « feedback »sur la conversation pour entraîner le système, ce qui rend son apprentissage autonome.

2.2 Différents modèles de dialogue

La construction d'un système de dialogue se fait à partir d'une formalisation de la conversation. Dans ce paragraphe, nous décrivons les différents modèles de dialogue. Le modèle de dialogue le plus utilisé est celui de suivi d'état (*Dialogue State Tracking* ou *belief tracking*) qui consiste, pour chaque tour de parole, à pister les intentions et les croyances (*beliefs*) de l'utilisateur. L'évaluation des expériences menées à partir de ce modèle se font avec le DSTC que nous avons évoqué en section 2.1.

Les meilleures performances à ce jour sont atteintes par StateNet (Ren *et al.*, 2018), un pisteur d'états universel. Notons que (Mrksic *et al.*, 2016) obtiennent également des performances très élevées qui dépassent pour la première fois les systèmes de détection d'état manuels grâce à un système neuronal. Plus récemment, le modèle de pistage de cadres (*frames*) de (Schulz *et al.*, 2017) permet au système de s'affranchir de la recherche de l'intention immédiate de l'utilisateur. Il crée un graphe qui pointe sur chaque intention et les lie toutes entre elles de façon non-linéaire tout au long d'un dialogue. Dans la continuité de ce modèle, (Ultes *et al.*, 2019) ont développé le modèle d'entités conversationnelles, qui abandonne le principe de tour de parole pour ne s'intéresser qu'aux relations entre les entités conversationnelles qu'ils définissent comme : " une entité virtuelle qui existe dans le contexte d'une conversation en cours et est soit un objet conversationnel (ce dont on parle) soit une relation conversationnelle (ce qui mène à l'objet suivant)" [1].
Enfin, (Ghosh *et al.*, 2017; Zhou *et al.*, 2017; Asghar *et al.*, 2017) ont exploré l'émotion dans le dialogue et montré que l'analyse des émotions de l'utilisateur améliorait les performances du système. En effet, la détection d'émotion donne à l'agent des informations supplémentaires sur la réponse à fournir. Voyons maintenant comment mesurer la qualité des systèmes et l'évaluation de leurs performances.

2.3 Plusieurs méthodes d'évaluation

L'évaluation d'un système de dialogue orienté-tâche, par sa structure même (trois modules différents plus la satisfaction de l'utilisateur), est compliquée à réaliser dans sa globalité. Dès lors, comme l'évoque (Chen *et al.*, 2017), on utilise différentes métriques pour chaque module du système. Pour la compréhension de l'entrée, on utilise les métriques classiques de fouille de texte (Yong Nahm & J. Mooney, 2002) pour déterminer la capacité d'analyse du système. Notez que par le passé (Devillers *et al.*, 2003) ont proposé d'évaluer la compréhension en contexte en résumant l'historique du dialogue au moyen d'une paraphrase.
L'atelier DSTC a été créé pour évaluer les aspects gestion/décision du dialogue, pour lesquels plus de dix métriques différentes ont été développées. Il en est aujourd'hui à sa septième version [2] dont l'objet est l'évalution des systèmes end-to-end évoqués en 2.1. Pour la génération de réponses, ce sont des métriques de traduction automatique et de résumé automatique : BLEU, ROUGE et METEOR (yew Lin, 2004; Papineni *et al.*, 2002; Lavie & Agarwal, 2007) qui sont utilisées pour évaluer les systèmes (à l'exclusion des systèmes *retrieval-based* qui ne génèrent pas de réponse). Cependant, ces métriques ne reflètent pas la véritable performance du système et peinent à évaluer la satisfaction de l'utilisateur comme souligné par (Liu *et al.*, 2016b) qui proposent une version améliorée de ces trois mesures. Dès lors, les évaluations les plus fiables restent les campagnes d'évaluation des systèmes de dialogue (Devillers *et al.*, 2004) ou les méthodes Test de Turing ou Magicien d'Oz (Kelley, 1983), cette dernière permettant en outre d'augmenter artificiellement la taille des corpus d'interaction. On peut également citer le protocole PARADISE (Walker *et al.*, 1997) dévelopé dans le cadre du projet COMMUNICATOR (Walker *et al.*, 2001) du DARPA et qui repose sur la corrélation entre la satisfaction de l'utilisateur et des mesures de performance objectives, pour proposer un protocole qui soit indépendant de la tâche. Récemment, (Su *et al.*, 2018) présentent un agent conversationnel capable de s'évaluer à chaque tour de parole et de se corriger "en temps réel" en calculant à chaque interaction le taux de satisfaction de l'utilisateur, ainsi qu'en adaptant ses réponses selon ce taux.
Enfin, (Serban *et al.*, 2015) dresse une liste de nombreux corpus de dialogue disponibles pour la

1. traduit de l'anglais par les auteurs de cet article
2. http ://workshop.colips.org/dstc7

plupart en licences libres, servant à la fois pour l'apprentissage de systèmes et pour leur évaluation.

2.4 Frameworks et applications industrielles

2.4.1 Explorations académiques orientées applications

Mis à part les entreprises, de nombreux laboratoires se sont penchés sur la question des agents conversationnels. Nous pouvons notamment citer en France le LIMSI qui a été l'auteur de plusieurs projets, comme par exemple RAILTEL (Bennacef *et al.*, 1996), ou encore ARISE (Lamel *et al.*, 1998) qui ont fait suite à la réalisation de la borne MASK (Gauvain *et al.*, 1997) de réservation ferroviaire réalisée en partenariat avec la SNCF, un projet novateur pour son époque. D'autres travaux, toujours en collaboration avec la SNCF, ont également été menés par le LORIA en 2002 et 2003 pour le projet européen OZONE (Issarny *et al.*, 2005) afin de développer sur tablette, une IHM (Interactions Homme-Machine) pour la réservation de billets. De même, le projet AMITIES (Hardy *et al.*, 2004), issu d'une collaboration entre l'Union Européenne (FP6) et le DARPA, a servi de cadre au développement d'un un Serveur Vocal Intéractif prototype pour les banques. Auparavant, le système GUS (Bobrow *et al.*, 1977) proposait un système prototype pour la gestion du trafic de fret avec des résultats décevants. Les recherches académiques ont parfois donné lieu à de nouveaux langages dédiés à la réalisation d'agents conversationnels comme AIML, devenu populaire grâce à A.L.I.C.E (S. Wallace, 2009), un *chatbot* du début des années 2000, ou comme VoiceXML (Larson, 2003), conçu pour les serveurs vocaux interactifs. Les réalisations présentées dans cette section sont emblématiques d'une époque où les applications phares, indicatrices de ruptures technologiques en cours, provenaient principalement du monde académique à partir de financements publics.
Récemment, les travaux de (Campillos Llanos *et al.*, 2017) ont abouti à un système de dialogue appelé PatientGenesys [3] qui simule un patient pour entraîner les jeunes médecins pendant les rendez-vous médicaux.

2.4.2 Solutions et Frameworks grand public

Suite à l'amélioration des réseaux de neurones et leur popularisation (Lecun, 1985), les GAFAM [4] ont été les premiers acteurs du marché à s'emparer de la technologie. Outre leurs premiers produits finis (Google Assistant pour Google, Alexa pour Amazon, Siri pour Apple, Cortana pour Microsoft), ils ont également racheté des sociétés créant des environnements de développement (*framework*) afin de les proposer au grand public. Google rachète en 2016 api.ai, désormais DialogFlow, basé sur des RNN. Facebook acquiert wit.ai, Amazon développe son propre environnements de développement Lex (technologie utilisée par Alexa, l'assistant personnel d'Amazon disponible sur les enceintes connectées de la game Echo) (Janarthanam, 2017). D'autres plateformes plus indépendantes donnent accès à des environnements de développement comme recast.ai ou encore Destygo, une start-up spécialisée dans des questions de mobilité. Toutes ces plateformes proposent généralement deux accès à leur environnement de développement : un accès web et un accès sous forme de boîte à outils, téléchargeable et intégrable dans différents langages de programmation (python, JS, etc.). Toutefois, les GAFAM ne sont pas les seuls à proposer des environnements de développement d'agents conversationnels. En effet, OpenDial (Lison & Kennington, 2016) propose une boite à outils en Java pour la

3. https ://www.theconnectedmag.fr/patient-genesys-virtuel/
4. Google, Amazon, Facebook, Apple, Microsoft

création de systèmes de dialogue, mais aussi PyDial, (Ultes *et al.*, 2017), une boite à outils en python créée par l'Université de Cambridge. Nous pouvons aussi mentionner ChatterBot [5] une boite à outils en python.
Dernièrement, deux environnements de développement de système de dialogue en libre accès paraissent se dégager du lot. D'abord, nous trouvons *RASA stack* (Bocklisch *et al.*, 2017) qui est remarquable surtout pour son module de compréhension de la langue naturelle, car il utilise FastText (Joulin *et al.*, 2016) et les plongements de mots (Mikolov *et al.*, 2013) pour obtenir un modèle de langage pré-entraîné spécifique au domaine d'application du système de dialogue. Cela rend la classification textuelle très performante (Braun *et al.*, 2017). Ensuite, il existe *DeepPavlov* (Burtsev *et al.*, 2018), un framework conçu pour deux cas d'usage : l'implémentation d'un *chatbot* prêt à l'emploi, et la recherche scientifique en systèmes de dialogue ou TAL.

2.4.3 Limites de l'industrialisation

Malgré l'engouement pour les systèmes de dialogue, il n'existe pas à ce jour un système commercialisé qui mette toute la communauté d'accord sur sa qualité. En effet, nous pouvons citer cet article de *chatbotslife.com* [6] qui explique qu'un*chatbot* demande beaucoup de ressources linguistiques pour être efficace. Ces ressources manquent également à IBM qui annonce licencier entre 50 et 70% des effectifs de sa branche Watson [7]. De plus, il faut préciser que ces *chatbots* ne possèdent pas de méthode d'évaluation officielle ou admise. (Landragin, 2013) explique par ailleurs : "Il risque d'exister bientôt autant de méthodologies d'évaluation que de systèmes proprement dits. On peut s'interroger sur le bien-fondé d'une méthode d'évaluation proposée par les concepteurs d'un système dans le but d'évaluer ce seul système, la méthode étant elle-même évaluée par son application au système en question. Il est alors très difficile de savoir où va la technologie et comment l'amener plus loin" . Par ailleurs, les *chatbots* industriels demeurent des boîtes noires ne permettant pas de prendre facilement connaissance des méthodes avec lesquelles ils ont été constitués. Les corpus d'apprentissage par exemple, sont généralement inconnus lorsque le *chatbot* ne constitue pas une expérience scientifique. D'autre part, il est important de souligner que malgré le grand nombre de travaux académiques, l'implémentation des systèmes de dialogue et leur commercialisation se soldent souvent par un échec. En effet, les utilisateurs se sentent rapidement frustrés de ne pas avoir des résultats satisfaisants et délaissent vite les agents conversationnels au profit d'humains.

2.4.4 Communauté autour des *chatbots*

Malgré le scepticisme entourant cette technologie, de nombreuses communautés se sont organisées. Nous pouvons citer *PandoraBots*, qui au départ était une plateforme recensant différents *chatbots* et possédant un forum de discussion. Elle est devenue une société proposant des *chatbots* dans un langage de programmation spécifique à cette tâche, AIML (Wallace, 2004). *Botlist* [8], un site récent, propose une liste de *chatbots* disponibles sur les réseaux sociaux. *Chatbots Magazine* [9], est un site très actif autour de la technologie, qui regroupe à la fois des articles sur la question, mais aussi des tutoriels et

5. https ://github.com/gunthercox/ChatterBot
6. https ://chatbotslife.com/what-chatbots-can-learn-from-the-death-of-facebooks-failed-ai-acc05ad6d822
7. https ://www.usine-digitale.fr/article/licenciements-tensions-internes-ibm-watson-health-serait-dans-la-tourmente.N705304
8. https ://botlist.co/
9. https ://chatbotsmagazine.com/

des études. Enfin *Chatbots.org* [10] reste la référence en terme de recensement d'agents conversationnels, qu'ils soient encore en usage ou obsolètes. Certains sont même testables sur la plateforme. En ce qui concerne les chatbots dans l'industrie, nous constatons que les plus grands groupes (GAFAM + IBM) ne parviennent pas encore à developper des systèmes réellement satisfaisants, mais que la communauté ainsi que les développements publics autour des chatbots a beaucoup progressé ces dernières années, ce qui montre l'engouement général pour la technologie.Nous revenons maintenant au système de dialogue orienté-but, et nous justifions notre intérêt pour un modèle de mémoire cognitive au sein du système.

3 La mémoire dans les systèmes de dialogue

3.1 La mémoire dans le dialogue

Lors d'une conversation homme-homme, (Ciaramelli *et al.*, 2013; Denny *et al.*, 2014; Brod *et al.*, 2016) ont défini la capacité des personnes à dialoguer selon trois dimensions : connaissance, expérience et empathie, qu'ils regroupent dans la notion de mémoire. (Bangerter, 2004) parle d'attention jointe pour expliquer le focus de tous les interlocuteurs sur la même entité conversationnelle. Cette attention se porte sur le mot dans un dialogue. (Cintrón-Valentín & Ellis, 2016) la décrit comme la saillance linguistique.
(Atkinson & Shiffrin, 1968) définissent la mémoire comme "la faculté d'un système intelligent à enregistrer, conserver, et rappeler les expériences passées pour interpréter les expériences présentes". Leurs recherches leur ont permis de déterminer qu'il existait trois types de mémoire : la mémoire sensorielle (Klatzky, 1980), la mémoire à court terme (Sperling, 1967) et la mémoire à long terme (Rudner & Rönnberg, 2008). (Squire & Zola, 1996) raffine ce modèle de mémoire et explique que la mémoire à long terme se décompose en mémoire déclarative (ou volontaire) et mémoire non déclarative. (Greenberg & Verfaellie, 2010) parle de mémoire épisodique, qui se rapporte aux souvenirs autobiographiques ; et de mémoire sémantique, sorte d'encyclopédie des connaissances. (Ebbinghaus, 2013) décrit le concept de sauvegarde qui permet d'enregistrer dans la mémoire à long terme des informations. (Miller, 1962) appelle la mémoire non déclarative la mémoire procédurale, qui correspond à ce que l'humain retient inconsciemment (p.e. reconnaître son prénom).
La mémoire à court terme est en fait un terme générique qui regroupe la mémoire de travail (Baddeley, 2010) pour traiter les informations à court terme, la boucle phonologique (Baddeley *et al.*, 1998) pour recevoir les sons et les interprétrer et le calepin visio-spatial (Baddeley, 1995) pour se faire une représentation mentale des informations. Cette mémoire ne dure que quelques secondes le temps de visualiser les concepts associés aux mots perçus (ce qui ne permet de conserver en mémoire que quelques mots, entre sept et douze selon des expériences [11]). Les concepts sont transformés en informations qui sont encodées, stockées et interprétées dans la mémoire à long terme. Ce sont les mécanismes de rétention et de rappel (Roediger, 1990; Bodner & Lindsay, 2003). Lorsque les informations on été récupérées dans la mémoire à long terme, elles sont renvoyées dans la mémoire de travail qui fait le tri entre celles à garder et celles à oublier. La mémoire de travail sert également de centre exécutif qui décide également de la réponse à générer (McCabe *et al.*, 2010) . La figure 3 décrit ce schéma. (Poudade, 2006) a décrit un modèle de perception-action, où il définit la mémoire humaine comme une mémoire associative par le contenu. Il a montré l'impact de la taille de la mémoire de

10. https ://chatbots.org/
11. https ://www.cognifit.com/science/cognitive-skills/shortterm-memory

travail sur la complexité des représentations lexicales apprises par des agents lorsqu'ils jouent à des jeux de langage (Steels, 2000).
Un autre modèle utilisé pour décrire la mémoire dans le dialogue est le Belief-Desire-Intention (BDI) (E. Bratman, 1987) qui explique que lors d'un tour de parole, l'interaction est guidée par la croyance de l'interlocuteur, son désir (le but final, ce qu'il n'a pas encore) et l'intention (la façon dont il parvient à son but). Ce modèle est très bien adapté pour représenter la réalité d'une conversation, comme le montrent les réalisations de (Jokinen, 2018; Visser *et al.*, 2015; Wobcke *et al.*, 2005; Wadsley & Ryan, 2013).

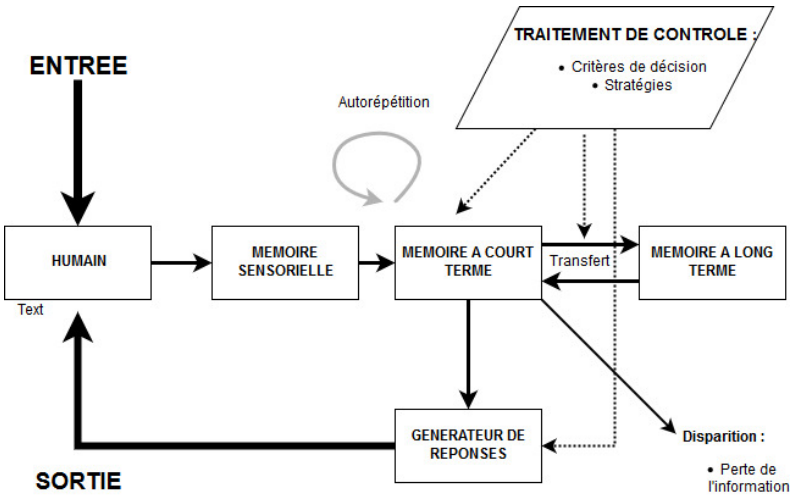


FIGURE 3 – Modèle classique de mémoire humaine

Dans le modèle de dialogue de (Clark & Marshall, 1981), on retrouve cinq concepts évoqués dans l'introduction : l'environnement, le tour de parole, l'historique du dialogue, l'expérience des interlocuteurs et les connaissances des interlocuteurs. Or si l'on compare avec le modèle de mémoire humaine, on retrouve ces cinq composantes principales : la mémoire de saillance (tour de parole, très court, compréhension rapide et très éphémère), la mémoire de travail (historique du dialogue, la mémoire à court terme conserve les informations principales du dialogue en cours pour assurer sa cohérence), la mémoire épisodique (l'expérience des conversations passées pour permettre à un interlocuteur d'être proactif), la mémoire sémantique (la connaissance, la culture de l'interlocuteur) et la mémoire procédurale (l'environnement, les actions réalisées de façon presque automatique, inconsciemment).
Ce parallèle évoqué, *dans quelle mesure l'intégration de ce modèle de mémoire dans un système de dialogue peut-il améliorer les performances de ce dernier ?*

3.2 Quelques systèmes utilisant la mémoire

Cette question a déjà été soulevée par (Rhodes, 1997) en décrivant un prototype d'agent capable de se souvenir de ses interactions. (Sieber & Krenn, 2010) implémente la différence entre mémoire épisodique et mémoire sémantique dans un système de dialogue. Les résultats montrent qu'en créant un graphe des conversations précédentes (expérience) pour l'agent, indépendamment de la base

de connaissances, l'agent est en mesure de résoudre partiellement le problème de coreference et de comprendre plus précisément les intentions de l'utilisateur. (Vetulani, 2005) parle du problème de l'exception dans le dialogue : c'est-à-dire la capacité d'un système de calcul de produire des hypothèses locales à partir de tours de parole précédents. On peut également citer (Young *et al.*, 2018) qui ont produit un modèle de dialogue à mémoire augmentée. Cependant, ils n'ont pas précisé le modèle de façon à ce qu'il corresponde au modèle de la mémoire cognitive. Certains réseaux spécifiques au dialogue tels que les réseaux de mémoire (Sukhbaatar *et al.*, 2015; Wan *et al.*, 2018) montrent également de bons résultats sur les tâches d'évaluation. Leur spécificité repose dans le fait que le réseau de mémoire à long-terme est capable d'inférer sur des mots et des concepts nouveaux, ainsi que de généraliser leur sémantique pour donner la réponse la plus probable pour une entrée inconnue dans leur mémoire à long-terme. C'est un système où l'entrée, son stockage, et la sortie sont effectuées simultanément. Récemment, (Kim *et al.*, 2019) ont combiné ces réseaux de mémoire avec des LSTM à deux directions (Schuster & Paliwal, 1997) (l'analyse du signal d'entée est bidirectionnelle dans une même couche de neurones) pour construire un système end-to-end obtenant des résultats très satisfaisants lors de leur évaluation au DSTC6 (Hori & Hori, 2017). (Zhang *et al.*, 2018) parlent également de mémoire à long terme accessible par le contenu dans leur système, ce qui se rapproche du fonctionnement cognitif de la mémoire et complexifie le stockage des informations dans la mémoire de l'agent. En complexifiant cette mémoire, l'agent est capable de mieux distinguer les différences entre deux "souvenirs" et affiner sa réponse en fonction cette capacité, ce qui lui permet d'augmenter la satisfaction de l'utilisateur. (Bang *et al.*, 2015) présentent des travaux sur de la mémoire à long terme personnalisée . (Chen *et al.*, 2018) ont construit un modèle basé sur des réseaux de mémoire hiérarchiques qui permet à un agent conversationnel de faire le lien entre le tour de parole et la mémoire à long terme afin d'obtenir des résultats état de l'art. Enfin, (El Asri *et al.*, 2017) sont ceux qui s'approchent le plus de notre réflexion car ils utilisent un corpus de dialogue pour étudier en détails le rôle de la mémoire cognitive dans l'interaction conversationnelle. Nous avons observé que depuis peu, les recherches sur les systèmes orientés-but se sont concentrés sur l'implémentation d'un modèle de mémoire d'inspiration cognitive, mais ces travaux n'ont pas pris en compte la dimension réflexive du traitement du langage lors d'une conversation mise en avant dans des travaux récents que nous allons introduire maintenant pour justifier notre intérêt pour l'apprentissage antagoniste.

4 Apprentissage antagoniste

Parmi les trois types de théories qui traitent de la nature des unités de parole, des travaux récents ont mis l'accent sur la théorie percepto-motrice (Schwartz *et al.*, 2008), par exemple avec le modèle COSMO (Barnaud *et al.*, 2017) comprenant cinq variables probabilistes définissent le modèle d'unités de parole, intégrant à la fois le rôle de locuteur et d'auditeur dans un couplage cohérent. Ces travaux suivent la tendance actuelle en faveur des modèles de perception sensori-moteurs (Gordon *et al.*, 2011) en général. Il est intéressant de noter que cette idée de réunir les rôles d'orateur et d'auditeur représente en quelque sorte un exemple d'apprentissage contradictoire, du moins pour toutes les fois dans un dialogue, où les tours de parole se chevauchent, un signe que les interlocuteurs sont en compétition pour prendre la parole (Martine Adda-Decker & Habert, 2008). Dans le principe de l'apprentissage antagoniste, l'oracle cherche à piéger l'agent apprenant qui lui doit déceler les pièges tendus par l'oracle en distinguant les vrais exemples d'apprentissage de ceux créés par l'oracle. On peut voir l'antagonisme comme une forme de co-apprentissage (Blum & Mitchell, 1998), où deux systèmes s'entraînent respectivement sur deux dimensions différentes associées à la même

information (p.e. le piège, et la façon de l'éviter).
Il serait intéressant d'utiliser cet apprentissage antagoniste pour entraîner l'agent conversationnel afin qu'il sache quoi répondre à un utilisateur lorsque celui-ci dit quelque chose d'inattendu, de nouveau, qui met en difficulté l'agent dans son objectif principal : satisfaire la demande de l'utilisateur.
(Kantze, 2007) montre que l'apprentissage par l'erreur dans un système de dialogue présente une alternative intéressante aux POMDP pour le problème de l'indécision. Plus récemment, (Liu *et al.*, 2018) ont développé un système orienté-but utilisant ce procédé d'apprentissage par l'erreur avec des résultats encourageants, qui peut inspirer des recherches futures dans le domaine. L'autre avantage de l'antagonisme est qu'il sert d'évaluation préliminaire. En effet, comme l'agent "oracle" donne à la fois une entrée et une récompense, il est capable d'évaluer la capacité de l'agent "apprenant" tout en essayant de le duper.
Nous avons l'intuition qu'il serait intéressant de déployer le paradigme d'apprentissage actif dans un système de dialogue à l'interface entre la compréhension du langage naturel et la gestion du dialogue. D'abord parce que l'apprentissage antagoniste fournirait une meilleure interprétation, grâce à une exploration plus ciblée de l'espace de recherche d'interprétation valide. En outre, l'apprentissage antagoniste peut souvent pallier le manque de données de formation en générant des données d'apprentissage pour l'adversaire en transformant des exemples positifs en exemples de formation négatifs. Enfin, explorer les avantages possibles d'une architecture plus cognitive, avec un aspect réflexif qui permettrait une distinction plus fine entre le contenu de la mémoire épisodique et sémantique. Bien entendu, de telles expériences devront se prémunir contre le risque habituel de divergence par rapport au modèle de données d'origine et de surapprentissage.

5 Conclusion et perspectives

Malgré leur maturité technologique et les récents progrès en performance liés en partie à la popularisation de nouveaux modèle neuronnaux d'apprentissage, les systèmes de dialogues ont encore devant eux une grande marge de progression illustrée par les limites affichées par les systèmes commercialisés aujourd'hui. Il serait intéressant de se concentrer sur la dimension mémorielle dans les systèmes de dialogue orientés-but, de modéliser le fonctionnement de la mémoire cognitive et de l'implémenter dans le moteur d'un système de dialogue afin d'améliorer sa capacité à converser avec un utilisateur humain. Nous pensons aussi que l'utilisation de l'apprentissage antagoniste permettrait à un agent conversationnel de mieux s'adapter à l'imprévu lors d'une conversation et d'augmenter les probabilités de donner une réponse satisfaisante à l'utilisateur. En effet, en s'inspirant du modèle cognitif de conversation où les participants possèdent chacun un modèle de l'autre et de soi pendant le dialogue, nous pensons que l'implémentation de ce modèle dans un agent conversationnel optimiserait sa capacité à répondre correctement. Nous voyons la combinaison de l'antagonisme pour entraîner un agent conversationnel doté d'un modèle mémoriel complexe de dialogue dans un système orienté-but, comme une piste prometteuse pour améliorer les systèmes existants à ce jour en leur permettant de mieux faire face aux imprévus.

Remerciements

Ce travail a été partiellement financé par l'ANRT et AKIO sous contrat CIFRE # 2017/1543.
Je remercie également les relecteurs dont les remarques ont permis d'améliorer cette présentation.

Références

ABADI M., AGARWAL A., PAULARHAM, BREVDO E., CHEN Z., CITRO C., CORRADO G. S., DAVIS A., DEAN J., DEVIN M., GHEMAWAT S., GOODFELLOW I., HARP A., IRVING G., ISARD M., JIA Y., JOZEFOWICZ R., KAISER L., KUDLUR M., LEVENBERG J., MANE D., MONGA R., MOORE S., MURRAY D., OLAH C., SCHUSTER M., SHLENS J., STEINER B., SUTSKEVER I., TALWAR K., TUCKER P., VANHOUCKE V., VASUDEVAN V., VIEGAS F., VINYALS O., WARDEN P., WATTENBERG M., WICKE M., YU Y. & ZHENG X. (2015). TensorFlow : Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

AGIRRE E., MARCHAND S., ROSSET S., PEÑAS A. & CIELIEBAK M. (2018). LIHLITH : improving communication skills of robots through lifelong learning. *ERCIM News*, **2018**(114).

ASGHAR N., POUPART P., HOEY J., JIANG X. & MOU L. (2017). Affective neural response generation. *CoRR*, **abs/1709.03968**.

ASGHAR N., POUPART P., JIANG X. & LI H. (2016). Online sequence-to-sequence active learning for open-domain dialogue generation. *CoRR*, **abs/1612.03929**.

ASLOT V., DOMEIKA M. J., EIGENMANN R., GAERTNER G., JONES W. B. & PARADY B. (2001). Specomp : A new benchmark suite for measuring parallel computer performance. In *WOMPAT*.

ATKINSON R. & SHIFFRIN R. (1968). Human memory. volume 2 of *Psychology of Learning and Motivation*, p. 89 – 195. Academic Press.

BADDELEY A. (2010). Working memory. *Current Biology*, **20**(4), R136 – R140.

BADDELEY A., GATHERCOLE S. & PAPAGNO C. (1998). The phonological loop as a language learning device. *Psychological review*, **105**, 158–73.

BADDELEY A. D. (1995). chapter The psychology of memory., p. 3–25. John Wiley and Sons : Oxford, England.

BANG J., NOH H., KIM Y. & LEE G. G. (2015). Example-based chat-oriented dialogue system with personalized long-term memory. In *2015 International Conference on Big Data and Smart Computing (BIGCOMP)*, p. 238–243.

BANGERTER A. (2004). Using pointing and describing to achieve joint focus of attention in dialogue. *Psychological science*, **15 6**, 415–9.

BARNAUD M.-L., DIARD J., BESSIÈRE P. & SCHWARTZ J.-L. (2017). Perceptuo-motor speech units in the brain with cosmo. In *11th International Seminar on Speech Production (ISSP 2017)*.

BARTL A. & SPANAKIS G. (2017). A retrieval-based dialogue system utilizing utterance and context embeddings. *CoRR*, **abs/1710.05780**.

BENNACEF S., DEVILLERS L., ROSSET S. & LAMEL L. (1996). Dialog in the railtel telephone-based system. In *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP '96*, volume 1, p. 550–553 vol.1.

BLUM A. & MITCHELL T. (1998). Combining labeled and unlabeled data with co-training. In *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, COLT' 98, p. 92–100, New York, NY, USA : ACM.

BOBROW D. G., KAPLAN R. M., KAY M., NORMAN D. A., THOMPSON H. & WINOGRAD T. (1977). Gus, a frame-driven dialog system. *Artificial Intelligence*, **8**(2), 155 – 173.

BOCKLISCH T., FAULKNER J., PAWLOWSKI N. & NICHOL A. (2017). Rasa : Open source language understanding and dialogue management. *CoRR*, **abs/1712.05181**.

BODNER G. E. & LINDSAY D. S. (2003). Remembering and knowing in context. *Journal of Memory and Language*, **48**(3), 563 – 580.

BORODIN A. & EL-YANIV R. (1998). Online computation and competitive analysis.

BRAUN D., HERNANDEZ-MENDEZ A., MATTHES F. & LANGEN M. (2017). Evaluating natural language understanding services for conversational question answering systems. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, p. 174–185 : Association for Computational Linguistics.

BROD G., LINDENBERGER U., WAGNER A. D. & SHING Y. L. (2016). Knowledge acquisition during exam preparation improves memory and modulates memory formation. *Journal of Neuroscience*, **36**(31), 8103–8111.

BUITINCK L., LOUPPE G., BLONDEL M., PEDREGOSA F., MUELLER A., GRISEL O., NICULAE V., PRETTENHOFER P., GRAMFORT A., GROBLER J., LAYTON R., VANDERPLAS J., JOLY A., HOLT B. & VAROQUAUX G. (2013). API design for machine learning software : experiences from the scikit-learn project. In *ECML PKDD Workshop : Languages for Data Mining and Machine Learning*, p. 108–122.

BURTSEV M., SELIVERSTOV A., AIRAPETYAN R., ARKHIPOV M., BAYMURZINA D., BUSHKOV N., GUREENKOVA O., KHAKHULIN T., KURATOV Y., KUZNETSOV D., LITINSKY A., LOGACHEVA V., LYMAR A., MALYKH V., PETROV M., POLULYAKH V., PUGACHEV L., SOROKIN A., VIKHREVA M. & ZAYNUTDINOV M. (2018). Deeppavlov : Open-source library for dialogue systems.

CHEN H., LIU X., YIN D. & TANG J. (2017). A survey on dialogue systems : Recent advances and new frontiers. *CoRR*, **abs/1711.01731**.

CHEN H., REN Z., TANG J., ZHAO Y. E. & YIN D. (2018). Hierarchical variational memory network for dialogue generation. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, p. 1653–1662, Republic and Canton of Geneva, Switzerland : International World Wide Web Conferences Steering Committee.

CIARAMELLI E., BERNARDI F. & MOSCOVITCH M. (2013). Individualized theory of mind (itom) : When memory modulates empathy. *Frontiers in Psychology*, **4**, 4.

CINTRÓN-VALENTÍN M. C. & ELLIS N. C. (2016). Salience in second language acquisition : Physical form, learner attention, and instructional focus. *Front Psychol*, **7**, 1284–1284. 27621715[pmid].

CLARK H. H. & MARSHALL C. R. (1981). Definite knowledge and mutual knowledge. In A. K. JOSHI, B. L. WEBBER & I. A. SAG, Eds., *Elements of Discourse Understanding*, p. 10–63. Cambridge, UK : Cambridge University Press.

DENNY C. A., KHEIRBEK M. A., ALBA E. L., TANAKA K. F., BRACHMAN R. A., LAUGHMAN K. B., TOMM N. K., TURI G. F., LOSONCZY A. & HEN R. (2014). Hippocampal memory traces are differentially modulated by experience, time, and adult neurogenesis. *Neuron*, **83**(1), 189–201. 24991962[pmid].

DEVILLERS L., BONNEAU-MAYNARD H., ROSSET S., PAROUBEK P., MCTAIT K., MOSTEFA D., CHOUKRI K., CHARNAY L., BOUSQUET-VERNHETTES C., VIGOUROUX N., BÉCHET F., ROMARY L., ANTOINE J.-Y., VILLANEAU J., VERGNES M. & GOULIAN J. (2004). The french media/evalda project : the evaluation of the understanding capability of spoken language dialogue systems. In *LREC*.

DEVILLERS L., MAYNARD H., PAROUBEK P. & ROSSET S. (2003). The peace slds understanding evaluation paradigm of the french media campaign. In *Proceedings of the EACL 2003 Workshop*

on Evaluation Initiatives in Natural Language Processing : are evaluation methods, metrics and resources reusable ?, Budapest, Hungary.

E. BRATMAN M. (1987). Intention, plans and practical reason. *Bibliovault OAI Repository, the University of Chicago Press*, **100**.

EBBINGHAUS H. (2013). Memory : a contribution to experimental psychology. *Ann Neurosci*, **20**(4), 155–156. 25206041[pmid].

EFSTATHIOU I. & LEMON O. (2014). Learning non-cooperative behaviour for dialogue agents. In *Proceedings of the Twenty-first European Conference on Artificial Intelligence*, ECAI'14, p. 999–1000, Amsterdam, The Netherlands, The Netherlands : IOS Press.

EL ASRI L., SCHULZ H., SHARMA S., ZUMER J., HARRIS J., FINE E., MEHROTRA R. & SULEMAN K. (2017). Frames : a corpus for adding memory to goal-oriented dialogue systems. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, p. 207–219 : Association for Computational Linguistics.

FERRUCCI D., BROWN E., CHU-CARROLL J., FAN J., GONDEK D., KALYANPUR A. A., LALLY A., WILLIAM MURDOCK J., NYBERG E., PRAGER J., SCHLAEFER N. & WELTY C. (2010). Building watson : An overview of the deepqa project. *AI magazines*, **31**(3), 59–79.

GAUVAIN J.-L., BENNACEF S., DEVILLERS L., LAMEL L. & ROSSET S. (1997). The spoken language component of the mask kiosk.

GHOSH S., CHOLLET M., LAKSANA E., MORENCY L.-P. & SCHERER S. (2017). Affect-lm : Neural language model for customizable affective text generation. *CoRR*, **abs/1704.06851**.

GOBET F. (2000). Long-term working memory : A computational implementation for chess expertise. *Proceedings of the Third International Conference on Cognitive Modelling*.

GOODFELLOW I., POUGET-ABADIE J., MIRZA M., XU B., WARDE-FARLEY D., OZAIR S., COURVILLE A. & BENGIO Y. (2014). Generative adversarial nets. In Z. GHAHRAMANI, M. WELLING, C. CORTES, N. D. LAWRENCE & K. Q. WEINBERGER, Eds., *Advances in Neural Information Processing Systems 27*, p. 2672–2680. Curran Associates, Inc.

GORDON G., KAPLAN D., LANKOW B., LITTLE D., SHERWIN J., SUTER B. & THALER L. (2011). Toward an integrated approach to perception and action. *Frontiers in Systems Neuroscience*, **5**(20).

GRABAR N. & RICHEY M. (2017). Détection automatique de grandes thématiques de la propagande nord coréenne. In *actes de Atelier Journalisme Computationnel*, Grenoble : EGC workshop.

GREENBERG D. L. & VERFAELLIE M. (2010). Interdependence of episodic and semantic memory : evidence from neuropsychology. *J Int Neuropsychol Soc*, **16**(5), 748–753. 20561378[pmid].

HARDY H., STRZALKOWSKI T., WU M., URSU C., WEBB N., BIERMANN A., INOUYE R. B. & MCKENZIE A. (2004). Data-driven strategies for an automated dialogue system. In *Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics*, ACL '04, Stroudsburg, PA, USA : Association for Computational Linguistics.

HIROYA F., HIROYUKI K., SUMIO O., TAKUYA I., KEN T. & KENJI A. (1997). An intelligent system for information retrieval over the internet through spoken dialogue. In *EUROSPEECH-1997* : 5th European Conference on Speech Communication and Technology.

HOCHREITER S. & SCHMIDHUBER J. (1997). Long short-term memory. *Neural Comput.*, **9**(8), 1735–1780.

HORI C. & HORI T. (2017). End-to-end conversation modeling track in DSTC6. *CoRR*, **abs/1706.07440**.

HOY M. B. (2018). Alexa, siri, cortana, and more : An introduction to voice assistants. *Medical Reference Services Quarterly*, **37**(1), 81–88.

HU Z., YANG Z., LIANG X., SALAKHUTDINOV R. & XING E. P. (2017). Controllable text generation. *CoRR*, **abs/1703.00955**.

HUANG P.-S., HE X., GAO J., DENG L., ACERO A. & HECK L. (2013). : ACM International Conference on Information and Knowledge Management (CIKM).

ILIEVSKI V., MUSAT C., HOSSMANN A. & BAERISWYL M. (2018). Goal-oriented chatbot dialog management bootstrapping with transfer learning. *CoRR*, **abs/1802.00500**.

ISSARNY V., SACCHETTI D., TARTANOGLU F., SAILHAN F., CHIBOUT R., LÉVY N. & TALAMONA A. (2005). Developing Ambient Intelligence Systems : A Solution based on Web Services. *Automated Software Engineering*, **12**(1), 101–137.

JANARTHANAM S. (2017). *Hands-On Chatbots and Conversational UI Development : Build Chatbots and Voice User Interfaces with Chatfuel, Dialogflow, Microsoft Bot Framework, Twilio, and Alexa Skills*. Packt Publishing.

JIANG J. (2012). *Information Extraction from Text*, In C. C. AGGARWAL & C. ZHAI, Eds., *Mining Text Data*, p. 11–41. Springer US : Boston, MA.

JOKINEN K. (2018). *Dialogue Models for Socially Intelligent Robots : 10th International Conference, ICSR 2018, Qingdao, China, November 28 - 30, 2018, Proceedings*, p. 127–138.

JOULIN A., GRAVE E., BOJANOWSKI P. & MIKOLOV T. (2016). Bag of tricks for efficient text classification. *CoRR*, **abs/1607.01759**.

JURAFSKY D. & MARTIN J. H. (2018). Dialog systems and chatbots. In D. JURAFSKY & J. H. MARTIN, Eds., *Speech and language processing*, chapter 24, p. 266–290. Londres : Pearson International Edition.

KANTZE G. S. (2007). Error handling in spoken dialogue systems managing uncertainty , grounding and miscommunication.

KEARNS M. J., LITMAN D. J., SINGH S. P. & WALKER M. A. (2011). Optimizing dialogue management with reinforcement learning : Experiments with the njfun system. *CoRR*, **abs/1106.0676**.

KELLEY J. F. (1983). An empirical methodology for writing user-friendly natural language computer applications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '83, p. 193–196, New York, NY, USA : ACM.

KIM B., CHUNG K., LEE J., SEO J. & KOO M.-W. (2019). A bi-lstm memory network for end-to-end goal-oriented dialog learning. *Computer Speech and Language*, **53**, 217 – 230.

KLATZKY R. (1980). Human memory : Structures and processes. *The American Journal of Psychology*, **93**.

KOWTKO J. C., ISARD S. D. & DOHERTY G. M. (1993). Conversational games within dialogue. In *University of Edinburgh*.

KÖLZER A. & ULM D. (1999). Universal dialogue specification for conversational systems.

LAI S., XU L., LIU K. & ZHAO J. (2015). Recurrent convolutional neural networks for text classification.

LAMEL L., ROSSET S., GAUVAIN J., BENNACEF S., GARNIER-RIZET M. & PROUTS B. (1998). The limsi arise system [rail travel information system]. p. 209 – 214.

LAMSIYAH S., OUATIK EL ALAOUI S. & ESPINASSE B. (2018). Résumé automatique guidé de textes : État de l'art et perspectives. In *actes de RJC*, Rennes : TALN-CORIA.

LANDRAGIN F. (2013). *Dialogue homme-machine*. Hermès Science-Lavoisier.

LARSON J. A. (2003). Voicexml and the w3c speech interface framework. *IEEE MultiMedia*, **10**(4), 91–93.

LAVIE A. & AGARWAL A. (2007). Meteor : An automatic metric for mt evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, p. 228–231, Stroudsburg, PA, USA : Association for Computational Linguistics.

LECUN Y. (1985). Une procedure d'apprentissage pour reseau a seuil asymmetrique (a learning scheme for asymmetric threshold networks). In *Proceedings of Cognitiva 85, Paris, France*, p. 599–604.

LECUN Y., BENGIO Y. & HINTON G. (2015). Deep learning. *Nature*, **521**, 436–44.

LI J., MONROE W., SHI T., RITTER A. & JURAFSKY D. (2017a). Adversarial learning for neural dialogue generation. *CoRR*, **abs/1701.06547**.

LI X., CHEN Y.-N., LI L. & GAO J. (2017b). End-to-end task-completion neural dialogue systems. *CoRR*, **abs/1703.01008**.

LISON P. & KENNINGTON C. R. (2016). Opendial : A toolkit for developing spoken dialogue systems with probabilistic rules. In *ACL*.

LIU B. & LANE I. (2018). Adversarial learning of task-oriented neural dialog models. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*, p. 350–359 : Association for Computational Linguistics.

LIU B., TUR G., HAKKANIUR D. Z., SHAH P. & HECK L. P. (2018). Dialogue learning with human teaching and feedback in end-to-end trainable task-oriented dialogue systems. *CoRR*, **abs/1804.06512**.

LIU C.-W., LOWE R., IULIAN VLAD SERBAN, NOSEWORTHY M., CHARLIN L. & PINEAU J. (2016a). How NOT to evaluate your dialogue system : An empirical study of unsupervised evaluation metrics for dialogue response generation. *CoRR*, **abs/1603.08023**.

LIU C.-W., LOWE R., SERBAN I., NOSEWORTHY M., CHARLIN L. & PINEAU J. (2016b). How NOT to evaluate your dialogue system : An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, p. 2122–2132, Austin, Texas : Association for Computational Linguistics.

LOWE R., NOSEWORTHY M., SERBAN I. V., ANGELAR-GONTIER N., BENGIO Y. & PINEAU J. (2017). Towards an automatic turing test : Learning to evaluate dialogue responses. *CoRR*, **abs/1708.07149**.

MARTINE ADDA-DECKER, CLAUDE BARRAS G. A. P. P. P. B. D. M. & HABERT B. (2008). Annotation and analysis of overlapping speech in political interviews. Marrakech, Morocco : European Language Resources Association (ELRA).

BARADARAN HASHEMI H. (2016). Query intent detection using convolutional neural networks.

CAMPILLOS LLANOS L., ROSSET S. & ZWEIGENBAUM P. (2017). Automatic classification of doctor-patient questions for a virtual patient record query task. In ACL, Ed., *BioNLP Shared-Task Workshop*, Vancouver, Canada : ACL.

CERCAS CURRY A., HASTIE H. & RIESER V. (2017). A review of evaluation techniques for social dialogue systems.

DAVID E. RUMELHART, GEOFFREY E. HINTON & RONALD J. WILLIAMS (1988). Neurocomputing : Foundations of research. chapter Learning Representations by Back-propagating Errors, p. 696–699. Cambridge, MA, USA : MIT Press.

KENNETH MARK COLBY (1974). Ten criticisms of parry. *SIGART Bull.*, (48), 5–9.

PINAR SAYGIN A., CICEKLI I. & AKMAN V. (2000). Turing test : 50 years later. *Minds and Machines*, **10**(4).

RICHARD S. SUTTON & ANDREW G. BARTO (1998). *Introduction to Reinforcement Learning*. Cambridge, MA, USA : MIT Press, 1st edition.

MCCABE D. P., ROEDIGER H. L., MCDANIEL M. A., BALOTA D. A. & HAMBRICK D. Z. (2010). The relationship between working memory capacity and executive functioning : evidence for a common executive attention construct. *Neuropsychology*, **24**(2), 222–243. 20230116[pmid].

MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G. & DEAN J. (2013). Distributed representations of words and phrases and their compositionality. *CoRR*, **abs/1310.4546**.

MILLER B. (1962). "physiologie de l'hippocampe". Colloques internationaux du Centre national de la recherche scientifique.

MISU T. & KAWAHARA T. (2008). Bayes risk-based dialogue management for document retrieval system with speech interface. *Speech Communication*, **52**, 61–71.

MOLDER HEDWIG TE. & POTTER J. (2005). *Conversation and Cognition*. Hedwig te Molder and Jonathan Potter.

MRKSIC N., SÉAGHDHA D. Ó., WEN T., THOMSON B. & YOUNG S. J. (2016). Neural belief tracker : Data-driven dialogue state tracking. *CoRR*, **abs/1606.03777**.

ODDY R. (1977). Information retrieval through man-machine dialogue. *Journal of Documentation*, **33**(1), 1–14.

PAPINENI K., ROUKOS S., WARD T. & ZHU W.-J. (2002). Bleu : A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, p. 311–318, Stroudsburg, PA, USA : Association for Computational Linguistics.

POUDADE J. (2006). *Emergence d'un lexique dans une population d'agents autonomes par l'action, la perception et la contingence sensorimotrice*. PhD thesis, Paris-Sud. Thèse de doctorat dirigée par Jardino, Michèle Informatique Paris 11 2006.

REN L., XIE K., CHEN L. & YU K. (2018). Towards universal dialogue state tracking. *CoRR*, **abs/1810.09587**.

RHODES B. J. (1997). The wearable remembrance agent : a system for augmented memory. In *Digest of Papers. First International Symposium on Wearable Computers*, p. 123–128.

ROEDIGER H. (1990). Implicit memory : Retention without remembering. *The American psychologist*, **45**, 1043–56.

ROSENBLATT F. (1958). The perceptron : A probabilistic model for information storage and organization in the brain. *Psychological Review*, p. 65–386.

ROSSET S. (2018). Introduction 1 aux systèmes de dialogue.

RUDNER M. & RÖNNBERG J. (2008). The role of the episodic buffer in working memory for language processing. *Cognitive processing*, **9**, 19–28.

RUSSELL, J. S., NORVIG P. & DAVIS E. (2010). *Artificial Intelligence : A Modern Approach*. Prentice Hall.

S. WALLACE R. (2009). *The anatomy of A.L.I.C.E*, p. 181–210.

SCHATZMANN J., WEILHAMMER K., STUTTLE M. & YOUNG S. (2006). A survey of statistical user simulation techniques for reinforcement-learning of dialogue management strategies. *The Knowledge Engineering Review*, **21**(2), 97–126.

SCHAUB L.-P. (2017). Récupération d'information dans un système de question-réponse à domaine fermé basé sur une ontologie en utilisant ibm watson : une solution hybride à base de règles et d'apprentissage automatique. Master's thesis, Inalco, Paris, France.

SCHULZ H., ZUMER J., EL ASRI L. & SHARMA S. (2017). A frame tracking model for memory-enhanced dialogue systems. *CoRR*, **abs/1706.01690**.

SCHUSTER M. & PALIWAL K. (1997). Bidirectional recurrent neural networks. *Trans. Sig. Proc.*, **45**(11), 2673–2681.

SCHWARTZ J.-L., SATO M. & FADIGA L. (2008). The common language of speech perception and action. *Revue française de linguistique appliquée*, **13**(2), 9–22.

SERBAN I. V., LOWE R., HENDERSON P., CHARLIN L. & PINEAU J. (2015). A survey of available corpora for building data-driven dialogue systems. *CoRR*, **abs/1512.05742**.

SHEN Y., HE X., GAO J., DENG L. & MESNIL G. (2014). Learning semantic representations using convolutional neural networks for web search. In *Proceedings of the 23rd International Conference on World Wide Web*, WWW '14 Companion, p. 373–374, New York, NY, USA : ACM.

SIEBER G. & KRENN B. (2010). Towards an episodic memory for companion dialogue. In J. ALLBECK, N. BADLER, T. BICKMORE, C. PELACHAUD & A. SAFONOVA, Eds., *Intelligent Virtual Agents*, p. 322–328, Berlin, Heidelberg : Springer Berlin Heidelberg.

SPERLING G. (1967). Successive approximations to a model for short term memory. *Acta Psychologica*, **27**, 285 – 292.

SQUIRE L. & ZOLA S. (1996). Structure and function of declarative and nondeclarative memory. *Proceedings of the National Academy of Sciences*, **93**.

STAFFORD L., BURGGRAF C. S. & SHARKEY W. F. (2006). Conversational Memory : The Effects of Time, Recall, Mode, and Memory Expectancies on Remembrances of Natural Conversations. *Human Communication Research*, **14**(2), 203–229.

STEELS L. (2000). The emergence of grammar in communicating autonomous robotic agents. In H. HORN, Ed., *Proceedings of European Conference on Artificial Intelligence, ECAI*, p. 764–769, Amsterdam : IOS Press.

SU P.-H., GAŠIĆ M. & YOUNG S. (2018). Reward estimation for dialogue policy optimisation. *Computer Speech and Language*, **51**, 24 – 43.

SUKHBAATAR S., SZLAM A., WESTON J. & FERGUS R. (2015). Weakly supervised memory networks. *CoRR*, **abs/1503.08895**.

THAKKAR J., RAUT P., DOSHI Y. & PAREKH K. (2018). Erasmus-ai chatbot. *International Journal of Computer Sciences and Engineering*, **6**.

TURING A. (1950). I.—COMPUTING MACHINERY AND INTELLIGENCE. *Mind*, **LIX**(236), 433–460.

ULTES S., BUDZIANOWSKI P., CASANUEVA I., ROJAS-BARAHONA L. M., TSENG B., WU Y., YOUNG S. J. & GASIC M. (2019). Addressing objects and their relations : The conversational entity dialogue model. *CoRR*, **abs/1901.01466**.

ULTES S., ROJAS BARAHONA L. M., SU P.-H., VANDYKE D., KIM D., CASANUEVA I. N., BUDZIANOWSKI P., MRKŠIĆ N., WEN T.-H., GASIC M. & YOUNG S. (2017). PyDial : A Multi-domain Statistical Dialogue System Toolkit. p. 73–78.

VETULANI Z. (2005). Dialogue processing memory for incident solving in man-machine dialogue. In L. BOLC, Z. MICHALEWICZ & T. NISHIDA, Eds., *Intelligent Media Technology for Communicative Intelligence*, p. 195–204 : Springer Berlin Heidelberg.

VISSER S., THANGARAJAH J., HARLAND J. & DIGNUM F. (2015). Preference-based reasoning in bdi agent systems. *Autonomous Agents and Multi-Agent Systems*, **30**.

WADSLEY T. & RYAN M. (2013). A belief-desire-intention model for narrative generation. In *Intelligent Narrative Technologies - Papers from the 2013 AIIDE Workshop, Technical Report*, volume WS-13-21, p. 105–108, United States : AI Access Foundation.

WALKER M., ABERDEEN J., BOLAND J., BRATT E., GAROFOLO J., HIRSCHMAN L., LE A., LEE S., NARAYANAN S., PAPINENI K., PELLOM B., POLIFRONI J., POTAMIANOS A., PRABHU P., RUDNICKY A., SANDERS G., SENEFF S., STALLARD D. & WHITTAKER S. (2001). Darpa communicator dialog travel planning systems : The june 2000 data collection. In *In proceedings of the 7th European Conference on Speech Processing (EUROSPEECH)*, p. 1371–1374, Aalborg, Denmark.

WALKER M. A. (2011). An application of reinforcement learning to dialogue strategy selection in a spoken dialogue system for email. *CoRR*, **abs/1106.0241**.

WALKER M. A., LITMAN D. J., KAMM C. A. & ABELLA A. (1997). PARADISE : A framework for evaluating spoken dialogue agents. *CoRR*, **cmp-lg/9704004**.

WALLACE R. (2004). The elements of AIML style. ALICE AI Foundation.

WAN Y., YAN W., GAO J., ZHAO Z., WU J. & YU P. S. (2018). Improved dynamic memory network for dialogue act classification with adversarial training. *CoRR*, **abs/1811.05021**.

WANG H. (2018). An annotated reading list of conversational ai. *Medium*.

WEISZ G., BUDZIANOWSKI P., SU P.-H. & GASIC M. (2018). Sample efficient deep reinforcement learning for dialogue systems with large action spaces. *CoRR*, **abs/1802.03753**.

WEIZENBAUM J. (1983). ELIZA - A computer program for the study of natural language communication between man and machine (reprint). *Commun. ACM*, **26**(1), 23–28.

WEN T.-H., GASIC M., MRKSIC N., LINA MARIA ROJAS-BARAHONA, SU P., ULTES S., VANDYKE D. & YOUNG S. J. (2016). A network-based end-to-end trainable task-oriented dialogue system. *CoRR*, **abs/1604.04562**.

WILLIAMS J. D., HENDERSON M., RAUX A., THOMSON B., BLACK A. W. & RAMACHANDRAN D. (2014). The dialog state tracking challenge series. *AI Magazine*, **35**(4), 121–124.

WOBCKE W., HO V., NGUYEN A. & KRZYWICKI A. (2005). A bdi agent architecture for dialogue modelling and coordination in a smart personal assistant. In *IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, p. 323–329.

XIAO Y. & CHO K. (2016). Efficient character-level document classification by combining convolution and recurrent layers. *CoRR*, **abs/1602.00367**.

YAN Z., DUAN N., CHEN P., ZHOU M., ZHOU J. & LI Z. (2017). Building task-oriented dialogue systems for online shopping.

YEW LIN C. (2004). Rouge : a package for automatic evaluation of summaries. p. 25–26.

YONG NAHM U. & J. MOONEY R. (2002). Text mining with information extraction.

YOUNG S. (2006). Using pomdps for dialog management. In *2006 IEEE Spoken Language Technology Workshop*, p. 8–13.

YOUNG S., BRESLIN C., GASIC M., HENDERSON M., KIM D., SZUMMER M., THOMSON B., TSIAKOULIS P. & TZIRKEL-HANCOCK E. (2013a). Evaluation of statistical pomdp-based dialogue systems in noisy environments.

YOUNG S., GASIC M., THOMSON B. & WILLIAMS J. D. (2013b). Pomdp-based statistical spoken dialog systems : A review. *Proceedings of the IEEE*, **101**(5), 1160–1179.

YOUNG S., GAŠIĆ M., KEIZER S., MAIRESSE F., SCHATZMANN J., THOMSON B. & YU K. (2010). The hidden information state model : A practical framework for pomdp-based spoken dialogue management. *Computer Speech and Language*, **24**(2), 150 – 174.

YOUNG T., CAMBRIA E., CHATURVEDI I., ZHOU H., BISWAS S. & HUANG M. (2018). Augmenting end-to-end dialogue systems with commonsense knowledge.

ZHANG Z., HUANG M., ZHAO Z., JI F., CHEN H. & ZHU X. (2018). Memory-augmented dialogue management for task-oriented dialogue systems. *CoRR*, **abs/1805.00150**.

ZHOU H., HUANG M., ZHANG T., ZHU X. & LIU B. (2017). Emotional chatting machine : Emotional conversation generation with internal and external memory. *CoRR*, **abs/1704.01074**.

Lifelong learning et systèmes de dialogue : définition et perspectives

Mathilde Veron[1]
(1) LIMSI, CNRS, Université Paris-Saclay, Campus universitaire, bât. 507, F-91405 Orsay Cedex, France
prénom.nom@limsi.fr

RÉSUMÉ

Le but de cet article est de définir comment le *Lifelong Learning* (LL) pourrait être appliqué aux systèmes de dialogue orientés tâche. Un système de dialogue devrait être en mesure d'apprendre de nouvelles connaissances, après avoir été déployé, et ceci de manière continue grâce à ses interactions avec l'utilisateur. Nous identifions ainsi deux aspects s'appliquant à un tel système : l'amélioration de ses capacités conversationnelles, et l'enrichissement de sa base de connaissances. Nous appliquons ces idées à un chatbot développé dans le cadre du projet LIHLITH. Nous montrons ainsi qu'un tel système doit être capable (1) de détecter la présence d'une situation inconnue (2) de décider quand et comment interagir avec l'utilisateur afin d'extraire de nouvelles connaissances et (3) de s'adapter à ces nouvelles connaissances, tout en considérant la fiabilité de celles-ci.

ABSTRACT

Lifelong learning and dialogue system : definition and discussion

This paper aims to define what could be a Lifelong Learning (LL) task-oriented dialogue system. A dialogue system should be able to learn new knowledge, after deployment and continuously, thanks to its interactions with the user. We identify two ways for a task-oriented dialogue system to perform LL: the improvement of its conversational capabilities, and the enrichment of its knowledge base. We apply these ideas to a chatbot developed as part of the LIHLITH project. We suggest that such a system should be able (1) to detect a new situation, (2) to decide when and how to interact with the user in oder to extract new knowledge and (3) to adapt itself to these new knowledge by considering their reliability.

MOTS-CLÉS : système de dialogue, apprentissage continu, système de dialogue orienté tâche.

KEYWORDS: dialogue system, lifelong learning, task-oriented dialogue system.

1 Introduction

Un système de dialogue permet à un utilisateur d'interagir en langage naturel. Deux familles de systèmes de dialogue existent : les systèmes conversationnels et les systèmes orientés tâche. Un système conversationnel a pour but de générer la réaction la plus appropriée étant donné un énoncé utilisateur et le contexte courant, sans restriction concernant le domaine. Quant à eux, les systèmes orientés tâche ont pour but d'aider l'utilisateur à réaliser des tâches définies, ou bien de l'aider à accéder à des informations. Un système de dialogue consiste généralement en trois modules : la compréhension du langage naturel (en anglais, *Natural Language Understanding* ou *NLU*), la gestion du dialogue, et la génération de langage naturel (en anglais, *Natural Language Generation* ou *NLG*).

Dans cet article, nous nous focaliserons sur les systèmes orientés tâche, et plus précisément sur les systèmes modulaires et non sur les *end-to-end*.

Récemment certains chercheurs se sont intéressés à la création de systèmes de dialogue réalisant de l'apprentissage continu (*Lifelong Learning* ou *LL*). Le LL est un sujet qui a mené à de nombreux travaux dans des domaines variés, tels que l'apprentissage automatique (*machine learning*), l'apprentissage profond (*deep learning*) [1], l'intelligence artificielle et la robotique (Thrun & Mitchell, 1995; Chen & Liu, 2016; Parisi *et al.*, 2018). Le LL est communément considéré comme la capacité à apprendre de manière continue de nouvelles connaissances, tout en conservant les données acquises lors d'expériences passées (Parisi *et al.*, 2018). Réaliser du LL dans chacun des domaines diffère selon la nature de la connaissance à apprendre, et de la façon dont cet apprentissage peut être réalisé. Dans le cadre des systèmes de dialogue, l'application du LL pourrait mener à des systèmes capable d'apprendre de façon continue grâce à leurs interactions avec l'utilisateur, à l'image des humains. Par exemple, un humain peut apprendre de nouvelles connaissances en posant des questions à ses interlocuteurs lorsqu'il fait face à une situation inconnue, ou simplement en déduisant de nouvelles connaissances à partir de l'ensemble des échanges qu'il a pu avoir au cours de sa vie. Un humain est aussi capable d'utiliser directement, ou dans de futures conversations, les connaissances qu'il a acquises. Ces connaissances peuvent correspondre, par exemple, à du vocabulaire, à des informations, ou même à un comportement approprié à avoir dans des conditions spécifiques. Un système de dialogue réalisant du LL devrait donc, à l'image d'un humain, être en mesure d'améliorer ses capacités conversationnelles et, dans le cas de systèmes orientés tâches, d'apprendre de nouvelles connaissances spécifiques à son domaine d'application.

Nous cherchons donc à définir jusqu'où un système de dialogue est en mesure d'imiter l'homme et sa capacité à apprendre de manière continue. Pour cela, nous cherchons à déterminer de façon précise ce qu'un système de dialogue peut apprendre, les limites de son apprentissage, ainsi que les méthodes qui devront être développées pour atteindre cet objectif.

Dans cet article nous proposons une définition du LL adapté aux systèmes de dialogue orientés tâche et montrons comment celle-ci peut s'appliquer dans un cas pratique.

Dans la suite de ce document, nous commençons par présenter en section 2 des travaux qui ont été menés sur le LL, sur l'apprentissage par renforcement et sur les systèmes de dialogue apprenant. Dans la section 3, nous cherchons à définir ce que peut être un système de dialogue réalisant du LL ainsi que les méthodes et techniques qui pourront ou devront être utilisées. Enfin, en section 4, nous appliquons cette définition ainsi que les méthodes présentées précédemment à un chatbot dédié à la cuisine. Il ne s'agit ici que de propositions et de discussions et aucun travail expérimental n'est présenté.

2 État de l'art

Le Lifelong learning (LL) est d'intérêt croissant depuis plusieurs années, ce qui a mené à de nombreux travaux et écrits (Thrun & Mitchell, 1995; Chen & Liu, 2016; Parisi *et al.*, 2018).

Dans leur livre (Chen & Liu, 2016), Chen et Liu définissent trois caractéristiques représentatives d'un système réalisant du LL : (1) l'apprentissage continu, (2) l'accumulation et la maintenance des connaissances dans une base de connaissances (en anglais, *Knowledge Base* ou *KB*) et (3) la capacité

1. par exemple, Continual Learning Workshop, NeurIPS 2018

à utiliser les connaissances acquises pour de futurs apprentissages. De plus, ils considèrent qu'un système réalisant du LL doit être capable de détecter lorsqu'une nouvelle situation se présente dans l'usage, et d'adapter son comportement à ces nouvelles situations, en apprenant à réaliser de nouvelles tâches de manière pro-active. Les connaissances acquises et accumulées doivent de plus permettre au système d'apprendre de nouvelles tâches sans nécessiter de gros volumes de données supplémentaires et sans devoir déployer de moyens trop importants. Cependant, le concept de *nouvelle tâche* reste flou et semble couvrir un large spectre de situations, pouvant aller d'une nouvelle instance (par exemple, un nouveau concept pour un modèle) à un domaine complètement différent (Maltoni & Lomonaco, 2018).

Dans les mêmes temps, des avancés ont été faites dans le domaine de l'apprentissage par renforcement (en anglais, *Reinforcement Learning* ou *RL*). Un système ayant recourt à des méthodes de RL apprend à partir d'expériences passées à estimer le processus de décision le plus approprié à la situation courante, en maximisant une fonction de récompense prédéfinie. Le RL appliqué aux systèmes de dialogues pour l'apprentissage de politique de dialogue a été largement étudié (Young, 2006), permettant la considération de scénarios *online* (Papangelis *et al.*, 2012) et le développement de techniques d'*Inverse Reinforcement Learning (IRL)* (Chandramohan *et al.*, 2011). Le IRL agit à l'inverse du RL en extrayant la fonction de récompense à partir de l'observation du comportement de l'agent considéré, c'est à dire de l'utilisateur dans le cas des systèmes de dialogue.
Plus récemment, Li et ses collègues (Li *et al.*, 2017) ont appliqué les méthodes de RL à leur système de dialogue, dont le but est de répondre à des questions dans le domaine du cinéma. Le RL permet au système de décider quand poser des questions à l'utilisateur et quelles questions poser, dans le but de permettre au système d'améliorer ses performances.
Li et ses collègues ont identifié trois situations durant lesquelles leur système peut être amené à poser des questions :

— *Problème de compréhension :* Le système demande à l'utilisateur de formuler une périphrase ou de corriger son énoncé. Li et ses collègues ne se concentrent ici que sur des fautes de frappes et des fautes d'orthographe/conjugaison.
— *Problème opérationnel :* Le système a besoin d'aide pour faire le lien entre la demande de l'utilisateur et les connaissances à sa disposition. Le système demande alors un indice à l'utilisateur.
— *Manque de connaissance :* Le système demande à l'utilisateur la réponse.

Certains chercheurs ont aussi montré leur intérêt dans la construction de systèmes de dialogue capables d'apprendre, après avoir été déployés, et ceci de manière continue, grâce à ses interactions avec l'utilisateur. En particulier, Mazumder et ses collègues (Mazumder *et al.*, 2018) ont développé des méthodes et des techniques, permettant à un chatbot d'apprendre de nouvelles connaissances au cours de son utilisation. Pour cela, ils se sont focalisés sur la résolution du problème de complétion d'une base de connaissances, en domaine ouvert, grâce à l'apprentissage par l'interaction avec un utilisateur et grâce au processus d'inférence. Ils ont ainsi mis en place un système stockage pour les connaissances acquises, ainsi qu'un mécanisme permettant d'acquérir des connaissances même si l'utilisateur n'est pas en mesure de fournir les informations requises par le système.
Dans le seconde édition de leur livre (Chen *et al.*, 2018), Chen et Liu ont dédié le 8ème chapitre à l'apprentissage continu de connaissances dans le cadre des chatbots. Dans ce chapitre, ils s'intéressent principalement à l'article présenté précédemment (Mazumder *et al.*, 2018) et déclarent que le principe de LL est reflété par le fait que les faits acquis au cours des interactions sont stockés directement dans la base de connaissances, et utilisés pour réaliser de l'inférence pour de futures requêtes. Il déclarent, de plus, que les connaissances accumulées, ainsi que la nouvelle base de connaissances, permettent

de guider les interactions et l'apprentissage futurs.
De leur côté, Hancock et ses collègues (Hancock *et al.*, 2019) se sont focalisés sur la réalisation d'un système purement conversationnel, capable d'extraire de nouveaux exemples à partir des conversations auxquelles il participe. Ces nouveaux exemples lui permettent par la suite de ré-entraîner son modèle afin de s'améliorer au fur et à mesure de son utilisation. Ce système se base en particulier sur les retours de l'utilisateur : si au cours de la conversation l'utilisateur semble satisfait de la réponse du chabot, celle-ci sera considérée comme un nouvel exemple ; s'il ne semble au contraire pas satisfait, le chabot demandera à l'utilisateur une réponse plus appropriée à la situation, et cette réponse sera considérée comme un nouvel exemple. L'estimation de la satisfaction de l'utilisateur via un classifieur permet au système de s'apercevoir lorsqu'une erreur a été commise. Hancock et ses collègues ont d'ailleurs prouvé que cette approche surpassait celle basée sur l'estimation de l'incertitude du système sur la réponse à donner à l'utilisateur.

Cependant, les deux travaux précédents ne donnent que des exemples spécifiques de systèmes de dialogue réalisant du LL. Nous donnons ainsi dans la section 3 une définition plus globale d'un tel système.

3 Le Lifelong Learning appliqué aux systèmes de dialogue orientés tâches

Nous proposons ici notre définition d'un système de dialogue réalisant du LL et nous décrivons les méthodes et techniques qui pourront être utilisées afin de construire un tel système. Nous discutons de plus des protocoles qui devront être créés afin de permettre l'évaluation de tels systèmes.

3.1 Définitions

Avant de définir ce que pourrait être un système de dialogue réalisant du LL, il faut d'abord se rendre compte que le LL ne doit pas être considéré comme une méthode, à l'image de l'apprentissage par renforcement, mais plutôt comme un objectif. En effet, la nature des connaissances pouvant être acquises, les méthodes impliquées ainsi que les objectifs, dépendent fortement du domaine considéré. Il serait cependant intéressant de définir des concepts et des méthodes pouvant s'appliquer à plusieurs domaines d'étude.

Grâce aux définitions données dans (Chen & Liu, 2016; Chen *et al.*, 2018) et aux travaux décrits précédemment (Li *et al.*, 2017; Mazumder *et al.*, 2018; Hancock *et al.*, 2019), nous pouvons dire qu'un système de dialogue réalisant du LL doit être en mesure d'apprendre de nouvelles connaissances, *après avoir été déployé*, et ceci de manière *continue* grâce à ses *interactions* avec l'utilisateur et possiblement en lui posant des questions. Les connaissances acquises doivent ainsi pouvoir être *stockées*, de manière à ce que le système puisse immédiatement y avoir accès, et pour qu'il puisse directement *adapter* son comportement en fonction. Ces modifications devront de plus être conservées pour de futurs dialogues et les connaissances acquises devront aider le système à *déduire* d'autres connaissances et à *apprendre dans le futur de manière plus efficace et plus rapide*. Lors de ses interactions avec l'utilisateur, le système devra donc avant tout être en mesure de *détecter* lorsqu'une *situation inconnue* se présente et être capable d'en extraire de nouvelles connaissances. La nouvelle connaissance peut correspondre, par exemple, à du vocabulaire, à des informations spécifiques au

domaine d'application, à des faits, à des règles, à un comportement à avoir dans des conditions spécifiques, à un nouvel exemple d'entraînement, etc.

Nous nous focalisons à présent sur les systèmes de dialogue orientés tâche. Ce type de système a pour but d'aider l'utilisateur à réaliser une tâche, ou bien de l'aider à accéder à des informations. Pour cela, le système cherche d'abord à comprendre ce que l'utilisateur lui demande en faisant appel au module de compréhension (NLU) qui réalise alors sur l'énoncé utilisateur une opération de détection des concepts (en anglais, *slot-filling*) et une autre de détection de l'intention (en anglais, *intent detection*). Ensuite, le gestionnaire de dialogue cherche à lier les concepts et l'intention détectés avec les données auxquelles il a accès. Enfin, le système retourne le résultat en générant du langage naturel.

À partir de la précédente définition et des distinctions réalisées dans (Li *et al.*, 2017), nous pouvons remarquer qu'un système de dialogue peut apprendre à différents niveaux :

- *Améliorer les interactions :*
  - améliorer la compréhension des énoncés utilisateur (NLU)
    - améliorer la détection des concepts et des intentions
    - être capable de détecter de nouveaux concepts et de nouvelles intentions
    - être capable de détecter des énoncés hors domaine
  - améliorer la génération de langage naturel (NLG)
- *Enrichir la base de connaissance* : le système peut apprendre de nouvelles informations en enrichissant sa base de connaissance
- *Lier des intentions aux données :* le système est capable d'apprendre à faire le lien entre des intentions utilisateur connues ou nouvelles et des données auxquelles il a accès. Il peut ainsi apprendre à réaliser de nouvelles tâches.

Ainsi, nous considérons qu'un système de dialogue réalisant du LL doit être capable (1) de détecter une nouvelle situation (2) de décider quand et comment agir en réponse à une nouvelle situation afin d'en extraire de nouvelles connaissances (3) de s'adapter à ces nouvelles connaissances. Ces trois points sont détaillés en section 3.2 et sont illustrés dans la section 4.

3.2 Méthodes et techniques associées

Pour chacune des compétences listées en fin de section 3.1 peuvent être associées des méthodes et techniques que nous décrivons ici.

(1) *Détecter une nouvelle situation* :
Lors d'un dialogue, le système peut se rendre compte qu'il est face à une situation inconnue de plusieurs manières. Ceci peut se faire par une estimation de son incertitude via l'étude des scores obtenus en sortie des modèles utilisés par le système. Celui-ci peut, en effet, ne pas être sûr d'avoir bien compris l'utilisateur, ou bien, ne pas être sûr de la meilleure manière de réagir. Le système peut aussi se reposer sur les retours de l'utilisateur qui lui indiqueront s'il l'a mal compris ou s'il n'a pas réagit comme il aurait dû, c'est à dire en estimant sa satisfaction à l'aide d'un classifieur par exemple (Hancock *et al.*, 2019). D'autres manières de détecter une nouvelle situation sont envisageables, mais celles-ci dépendent très souvent du domaine d'application du système de dialogue ainsi que des situations auxquelles il doit faire face.

(2) *Décider quand et comment agir* :
Dans certaines situations, ceci peut revenir à décider quand poser des questions à l'utilisateur, et quelle question poser. Cependant, interroger l'utilisateur implique un coût qui reflète la patience de

l'utilisateur (Li *et al.*, 2017). Le système doit donc être en mesure de s'adapter à l'utilisateur, sachant que certains seront plus enclin à donner de leur temps, tandis que d'autres seront moins patients. De plus certaines questions peuvent être plus intéressantes à poser pour permettre au système de s'améliorer le plus possible. L'idée ici serait ainsi d'avoir recours à des méthodes d'apprentissage par renforcement soit classiques, soit inverses, dont les paramètres seraient évolutifs.

(3) *S'adapter aux nouvelles connaissances* :
Dans le cas des système de dialogue orientés tâche, le système peut s'adapter différemment en fonction de la nature de la connaissance acquise. Il peut notamment enrichir sa base de connaissances, ou ré-entraîner son/ses modèle(s) (par exemple ceux associés au NLU ou au NLG). Ces perspectives soulèvent les deux questions suivantes : sous quelles conditions considère-t-on la nouvelle connaissance comme étant pertinente et/ou fiable ? Et doit-on directement enrichir la base de connaissances ou ré-entraîner le modèle associé dès qu'une nouvelle connaissance est acquise ? Ceci implique de plus une entité de stockage spécifique pour ces connaissances acquises qui n'ont pas encore été considérées comme fiables.
En effet, la question de la fiabilité des connaissances acquises se pose puisqu'il est possible que l'utilisateur transmette des informations incorrectes. Par exemple, dans le cas d'un enrichissement de la base de connaissance, le système peut être amené à demander à l'utilisateur une information spécifique qui manquerait à sa base de connaissance. Le problème est qu'il est possible que l'utilisateur ne connaisse pas la réponse, qu'il peut se tromper ou que la donnée peut être subjective. L'utilisateur peut aussi être amené à faire des fautes de frappes ou d'orthographe ou avoir une manière propre de s'exprimer en fonction de son âge et de son lieu de résidence par exemple.
Dans le cadre de l'enrichissement de la base de connaissance, on peut estimer qu'à partir du moment où une connaissance est estimée fiable, celle-ci peut directement être ajoutée à la base de connaissances du système, puisque le coût associé est généralement faible. Dans le cas, cependant, où l'on voudrait enrichir un modèle, étant donné que le ré-entraînement du modèle implique un coût important, ceci devra se faire sous certaines conditions (par exemple de façon périodique, au bout d'un nombre défini d'exemples fiables acquis, etc).
L'apprentissage de nouvelles connaissances peut cependant mener à une baisse de performance sur les tâches concernant les anciennes connaissances, comme on peut l'observer lorsque l'on est amené à ré-entraîner un modèle avec de nouveaux exemples, comme expliqué dans (Parisi *et al.*, 2018).

3.3 Protocoles d'évaluation envisageables

Développer de tels systèmes demande dans le même temps l'accès à des méthodes et des métriques d'évaluation adaptées. Celles-ci devront nous permettre d'évaluer la plus-value du LL sur un système de dialogue. Les difficultés reposent sur notre capacité à évaluer un système *au cours du temps* et sur l'ensemble des diverses sous-tâches impliquées dans le LL. Cette évaluation nécessitera ainsi une définition précise des tâches impliquées dans le LL pour les systèmes de dialogue ainsi que l'élaboration des corpus d'évaluation associés à chaque tâche. Il s'agira de plus d'étudier si l'évaluation pourra se faire avec des métriques existantes ou si celle-ci nécessitera le développement de métriques spécifiques.
Conformément aux compétences nécessaires listées en section 3.1, devront être évaluées (1) la capacité du système à détecter une nouvelle situation (2) la capacité du système à décider quand et comment agir en réponse à une nouvelle situation afin d'en extraire de nouvelles connaissances et (3) la capacité à s'adapter à ces nouvelles connaissances, tout en considérant la fiabilité de celles-ci.

4 Applications possibles du Lifelong Learning à un chatbot appliqué à la cuisine

Dans le cadre du projet LIHLITH [2], nous avons développé une première version d'un chatbot appliqué au monde de la cuisine [3]. Celui-ci permet notamment à l'utilisateur de trouver une recette correspondant à ses critères, tout en interagissant avec le chatbot en anglais.

Nous commençons en section 4.1 par présenter le système de manière générale. Ensuite en section 4.2, nous illustrons à l'aide d'exemples, comment le LL peut s'appliquer à ce chatbot et discutons des méthodes et techniques envisageables.

4.1 Description générale du chatbot

Le chatbot prend en charge deux types de scénarios :

1. L'utilisateur veut trouver une recette qui répond à ses besoins : le système va alors chercher dans la base de données des recettes. Pour cela, l'utilisateur peut par exemple demander "Please find me a recipe of pancakes without eggs".
2. L'utilisateur pose une question concernant le domaine de la cuisine en général : le système va alors chercher dans des données non structurées en ayant recours à des calculs de similarité textuelle sémantique. Pour cela, l'utilisateur peut par exemple demander "Why is -18°C the ideal freezer temperature ?".

La base de recettes a été construite à partir d'articles de type recette venant de Wikipedia:Cookbook et comporte *1,064 recettes*. Cette base de données contient notamment des informations sur le nom de la recette, sur les ingrédients nécessaires, sur les étapes à suivre et sur le temps nécessaire.

Le chatbot est divisé en trois sous-systèmes :

— *Le NLU :* Il prend en entrée un énoncé utilisateur et retourne les concepts et l'intention associés [4]. Ce module est basé sur un réseau de neurones profond réalisant à la fois l'étape de *slot-filling* et celle *d'intent detection* (Tur & Mori, 2011), aussi connu sous le nom de *joint NLU*. Nous utilisons pour cela un système opérationnel [5]. Le modèle de compréhension a été entraîné à partir de données qui ont été automatiquement générées et annotées grâce à des patrons et des listes de termes comme proposé dans (Neuraz *et al.*, 2018).

— *Le gestionnaire de dialogue :* Il décide quelles actions réaliser et quelle réponse donner à l'utilisateur en fonction des résultats du NLU et du contexte courant. Il peut en fonction de la situation, accéder à sa base de recettes ou bien chercher des informations dans les données non structurées.

— *Le NLG :* Il génère du langage naturel à partir de patrons de phrases pour permettre au gestionnaire de dialogue de répondre à l'utilisateur

Le gestionnaire de dialogue et le NLG sont tous les deux basés sur des systèmes de règles.

2. LIHLITH : Learning to Interact with Humans by Lifelong Interaction with Humans

3. Le chatbot peut être testé à l'adresse suivante : `https://lihlith.limsi.fr/dialog.php` avec le login "lihlith" et le mot de passe "recipe ?"

4. Dans l'exemple "Please find me a recipe of pancakes without eggs", les slots détectés seraient "recipe : pancakes" et "neg-ingredient : eggs" et l'intention serait "RECING", c'est à dire que l'utilisateur cherche une recette en donnant le nom de la recette ainsi que des ingrédients.

5. Des informations supplémentaires sur ce système peuvent être trouvées sur : https ://github.com/SNUDerek/multiLSTM

4.2 Exemples d'applications possibles du Lifelong Learning

A partir des définitions données en section 3.1, nous avons identifié plusieurs manières de réaliser du LL dans notre chatbot. Les deux possibilités que nous présentons ici ne concernent que la recherche d'une recette selon des critères utilisateurs. Elles correspondent à l'amélioration de la compréhension et à l'enrichissement de la base de données. Des exemples de dialogue illustrant ces possibilités sont donnés dans cette section. Nous discutons de plus de la manière dont nous pourrons les implémenter dans le système, des méthodes et techniques pouvant être envisagées, ainsi que des difficultés pouvant être rencontrées.

4.2.1 Amélioration de la compréhension

Au cours du dialogue, le chatbot doit pouvoir tirer profit de ses interactions avec l'utilisateur afin d'améliorer ses capacités de compréhension. Dans ce but, le système doit être capable d'apprendre à mieux détecter les concepts et les intentions, d'apprendre de nouveaux concepts et de nouvelles intentions, et d'apprendre à les mettre en lien avec les données dont il a accès. Cependant, le système doit d'abord être en mesure de détecter lorsqu'il fait face à une situation inconnue, comme défini dans la section 3.1. Dans notre cas, la nouvelle situation peut s'illustrer par une mauvaise compréhension de l'énoncé utilisateur. Le système peut alors détecter qu'il a mal compris l'utilisateur de plusieurs manières :

1. s'il y a un conflit entre les concepts et l'intention qui ont été détectés [6] ;
2. si le score associé aux concepts et à l'intention détectés sont bas ;
3. si l'utilisateur réagit négativement en disant par exemple "You misunderstood me" ou "You're wrong".

Dans le deuxième cas, il s'agirait d'évaluer l'incertitude du système de compréhension, avant qu'une erreur puisse être commise ; tandis que dans le troisième cas, il s'agirait d'estimer, après coup, la satisfaction de l'utilisateur pour savoir si une erreur a été commise (Hancock *et al.*, 2019). Dans notre cas, il serait intéressant de se demander si la combinaison de l'estimation de la satisfaction et de l'incertitude serait plus profitable que l'utilisation d'une des deux méthodes uniquement.

La Figure 1 illustre un exemple de conflit entre concept et intention ainsi que les décisions qui pourraient être prises par le système afin d'extraire des connaissances à partir de cette nouvelle situation.

Après avoir détecté la nouvelle situation, le système doit décider s'il doit entamer un processus d'échange avec l'utilisateur, afin d'en extraire une nouvelle connaissance. Dans l'exemple, le processus d'échange consiste en la clarification du concept et de l'intention qui auraient dû être détectés. En effet, il n'est pas toujours intéressant pour le système d'entamer ce processus : l'utilisateur peut, par exemple, être peu enclin à aider le chabot à s'améliorer ou il peut être devenu impatient après avoir répondu à plusieurs questions, ce qui peut entraîner le système à ne pouvoir extraire aucune nouvelle connaissance. Cette décision pourrait se reposer sur des méthodes d'apprentissage par renforcement classiques ou bien inverses dont le paramètre de coût et la fonction de récompense seraient évolutifs selon le comportement de l'utilisateur.

6. Par exemple le NLU peut avoir détecté l'intention/la classe RECING, c'est à dire que l'utilisateur cherche une recette à partir du nom de la recette et de critères sur les ingrédients, mais n'avoir détecté que le slot recipe (nom de la recette)

	Énoncé utilisateur	Action du système
U	i'd like to prepare a mousse	Le système n'a pas détecté que "mousse" correspond au concept "recipe" mais a détecté l'intention "REC"
S	Are you asking for a specific recipe?	Demande confirmation sur l'intention détectée
U	Yes	
S	Can you tell me the name of the recipe alone?	Demande le concept manquant
U	Mousse	Enregistre l'énoncé annoté : i'd / like / to / prepare / a / mousse / INTENT O / O / O / O / O / b-recipe / REC
S	I found 3 recipes, including: atkins-friendly chocolate mousse, chocolate mousse cake (vegan) and chocolate mousse . Which one do you want?	Répond à la requête initiale

FIGURE 1 – Exemple de dialogue entre le système (S) et un utilisateur (U) présentant une contradiction entre les concepts et l'intention détectés.

Si le système a pu extraire une connaissance de cette nouvelle situation, il lui faudra non seulement régler le problème au cours du dialogue courant, mais aussi s'adapter à cette nouvelle connaissance et en déduire d'autres connaissances pour de futurs dialogues. Dans l'exemple, la clarification du concept et de l'intention permet au système d'annoter l'énoncé utilisateur dont était issu la contradiction. Cette annotation peut être utilisées de différentes manières. Dans un premier temps, on peut imaginer enregistrer cette annotation dans un fichier spécifique. Ensuite, lorsque le système sera de nouveau confronté à des contradictions entre concepts et intention, il pourra vérifier si la situation qui lui pose problème n'a pas déjà été rencontrée. Cette vérification pourra être faite en comparant la similarité des énoncés annotés enregistrés dans le fichier, avec l'énoncé courant. L'annotation lui permettra alors de résoudre la contradiction sans avoir à déranger l'utilisateur, à la fois dans le dialogue courant, et dans de futurs dialogues. Dans un deuxième temps, on peut imaginer, grâce aux retours utilisateur et aux statistiques d'accès au fichier, qu'un certain nombre d'énoncés annotés peuvent être considérés comme utiles et fiables. Ces énoncés annotés pourront alors servir d'exemples d'entraînement et le système pourra décider sous certaines conditions de ré-entraîner le modèle de compréhension avec ces nouveaux exemples. Lors du ré-entraînement du système, il faudra prêter attention à respecter les proportions entre les classes (intentions) afin de contrer au maximum les pertes de performances liées à l'ajout de nouveaux exemples d'entraînement (Parisi *et al.*, 2018).
Dans notre cas, où nous générons et annotons automatiquement des énoncés utilisateur pour entraîner le modèle de compréhension, il est intéressant de déduire un patron de l'énoncé utilisateur annoté. Dans le cas de l'exemple présenté en Figure 1, le patron issu de la clarification serait "i'd like to prepare a $recipe". Ce patron nous permettra ensuite de générer et d'annoter automatiquement des énoncés supplémentaires pour ré-entraîner le modèle de compréhension. Ceci nous permettra possiblement d'améliorer de façon plus significative les performances du système de compréhension.

Le système peut aussi être amené à détecter une *nouvelle intention* dans l'énoncé utilisateur, comme présenté en Figure 2. Dans cet exemple, l'utilisateur a choisi une recette en particulier et demande des informations supplémentaires (sous-dialogue). L'enjeu ici, est de réussir à comprendre, à partir de l'énoncé utilisateur, qu'aucune intention recensée jusqu'ici ne correspond à la demande de l'utilisateur. On peut imaginer que cette nouvelle intention peut être détectée, par exemple, en observant les scores associés aux concepts et intentions détectés. Nous remarquons cependant que cette détection de

nouvelles intentions peut être difficile à mettre en place, à cause de la complexité et de la diversité de la langue naturelle, et à cause des diverses nouvelles intentions pouvant apparaître. De plus, si la nouvelle intention doit être liée à des concepts, il faudra être capable de replacer ces concepts aux bons endroits. Dans l'exemple, l'intention n'est associée à aucun concept, ce qui facilite grandement le processus.

	Énoncé utilisateur	Action du système
U	i'd like to cook roast chicken	
S	OK, for roast chicken . Do you want the ingredients?	Entre dans le sous-dialogue
U	What temperature for my oven?	
S	I don't know. Can you give me an example of an appropriate answer?	Demande un indice
U	210°C or 370 degree F	
S	I found "Preheat your oven to 200°C (390°F)" Are you satisfied with this answer?	Cherche dans le document recette la phrase qui comporte le plus de similarité avec l'exemple donné par l'utilisateur et demande un retour utilisateur
U	Yes	Déduit des connaissances

FIGURE 2 – Exemple de dialogue entre le système (S) et un utilisateur (U) présentant une nouvelle intention (sous-dialogue).

Pour être capable de traiter la nouvelle intention, nous pouvons imaginer que le système demande à l'utilisateur un indice, en lui demandant un exemple de réponse qui serait appropriée dans la situation courante. Cette méthode suppose cependant que la réponse ait une forme identique selon les contextes et que celle-ci soit simple. Le système pourrait alors chercher dans sa base de données ou dans des documents externes, des données textuelles similaires à l'exemple donné par l'utilisateur, comme envisagé dans l'exemple de la Figure 2. Ceci constitue une première méthode assez simple qui pourra être améliorée.
Après avoir donné sa réponse, et si l'utilisateur valide la pertinence, le système pourrait de plus entrer dans un processus de complétion automatique de sa base de données. En effet, comme proposé dans (Peñas *et al.*, 2019), le système pourrait vérifier dans sa base de données, si cette nouvelle intention peut aussi être appliquée aux autres recettes. Dans l'exemple de la Figure 2, le processus pourrait permettre la création de l'attribut "oven_temperature", et d'associer pour chacune des recettes nécessitant une cuisson au four, une valeur à cet attribut.

4.2.2 Enrichissement de la base de données

Au cours du dialogue, le chatbot devrait aussi être en mesure de tirer profit de ses interactions avec l'utilisateur, afin d'enrichir sa base de données.
Nous considérons l'exemple de la Figure 3, dans lequel l'utilisateur a choisi une recette spécifique (sous-dialogue). On peut imaginer qu'une information manque dans la base de recettes, comme le temps de préparation, la difficulté ou bien le nombre de parts. Dans ce cas, le système pourrait décider d'interroger l'utilisateur sur cette information manquante afin d'enrichir sa base de données. Cet exemple diffère des autres, dans le sens où la détection de la *nouvelle situation* n'est pas effectuée à partir d'un énoncé utilisateur, mais à partir du système lui-même (apprentissage pro-actif), en considérant que la nouvelle situation se manifeste par le manque d'information.

	Énoncé utilisateur	Action du système
S	Hello, are you hungry?	
U	i'd like to cook basic scones	
S	You're telling me about basic scones . Is that OK?	Demande confirmation
U	Ok	
S	OK, for basic scones . Do you want the ingredients?	
U	Yes	
S	The ingredients are: 1 cup flour (100g), 1 tsp baking powder (5g),1 pinch salt, 2 Tbs butter (50g), 1 Tbs sugar (50g),1 cup milk (125ml) + extra milk for glazing 2. Have you ever cooked basic scones?	Le système a détecté qu'au moins une information manque dans la base de données et décide d'interagir en demandant à l'utilisateur s'il a déjà préparé cette recette.
U	Yes	
S	Do you remember how much time did it take?	Demande l'information manquante
U	I needed approximately 30 min	Ajoute à la base de données "cooking_time = 30 min"
S	OK, thank you very much!	

FIGURE 3 – Exemple de dialogue entre le système (S) et un utilisateur (U) avec un enrichissement de la base de connaissance.

L'apprentissage par renforcement pourrait alors être utilisé afin d'apprendre au système quand il doit demander des informations à l'utilisateur, et sur quelle type d'information l'interroger, comme cela a été fait dans (Li *et al.*, 2017). Le processus s'illustrerait de la manière suivante (cf. Figure 3) :

a. Le système détecte qu'au moins une information est manquante pour la recette courante
b. Pour chacune des informations manquantes, le système estime la récompense cumulée associée au fait de demander cette information. Il estime aussi celle associée au cas où il ne poserait aucune question à l'utilisateur. Ces estimations lui permettent de décider s'il doit interroger l'utilisateur et sur quelle information l'interroger.
c. Si le système a décidé d'interroger l'utilisateur, il commence par lui demander s'il a déjà eu l'occasion de réaliser la recette courante.
d. Si ce n'est pas le cas, le système déduit que l'utilisateur ne pourra lui fournir aucune information et le processus s'arrête là. Dans l'autre cas, le système interroge l'utilisateur sur la question choisie en (b).
e. Si l'utilisateur donne une réponse, le système enregistre celle-ci. Il peut ensuite décider d'interroger l'utilisateur sur une autre information en ayant recours une nouvelle fois à de l'apprentissage par renforcement. Si l'utilisateur n'a pas connaissance de l'information demandée, il utilise aussi la méthode de RL afin de décider s'il interroge l'utilisateur sur une autre information.

La récompense associée au fait d'obtenir une certaine information pourra de plus dépendre du type de l'information et évoluer au cours du temps. Par exemple, si les utilisateurs demandent régulièrement des informations sur le temps de préparation, le système devra associer une plus grande récompense au fait d'obtenir cette information. De même, la fonction de récompense doit s'adapter au niveau de connaissance en cuisine de l'utilisateur courant ; si l'utilisateur connaît peu de chose en cuisine, il est peu intéressant pour le système de l'interroger. Le paramètre de coût pourra aussi évoluer au cours du temps en fonction du comportement de l'utilisateur. Il faudra cependant définir quels

paramètres doivent être partagé entre les différents utilisateurs et ceux qui ne devront être relatifs qu'aux utilisateurs eux-même.
Avant de décider d'ajouter l'information donnée par l'utilisateur à la base de données, le système pourrait la stocker de manière temporaire afin de tester sa fiabilité en questionnant d'autres utilisateurs sur la même recette et sur la même information par exemple.

5 Conclusion et discussion

Dans cet article, nous avons défini ce que pourrait être un système de dialogue orienté-tâche réalisant du LL. Celui-ci doit être en mesure d'apprendre de nouvelles connaissances, après avoir été déployé, et ceci de manière continue grâce à ses interactions avec l'utilisateur. Ensuite, nous avons appliqué cette définition à un un chatbot dédié à la cuisine et montré que celui-ci peut apprendre de deux manières différentes : en améliorant sa compréhension des énoncés utilisateur, et en enrichissant sa base de connaissances. Ces deux axes d'apprentissage pourront de plus être facilement adaptés à d'autres domaines d'applications.

Nous avons ainsi montré que réaliser du LL sur un système de dialogue orienté tâche implique que le système ait la capacité de :

— comprendre à partir des énoncés utilisateur en langage naturel qu'une nouvelle situation se présente, c'est à dire qu'une nouvelle connaissance peut être acquise. Ceci implique des méthodes de détection reposant par exemple sur l'estimation de la satisfaction de l'utilisateur ou de l'incertitude du système.
— décider quand et comment agir en réponse à une nouvelle situation, afin d'en extraire une nouvelle connaissance, tout en ayant recours à des critères évolutifs. Dans ce cas, l'apprentissage par renforcement peut être envisagé afin d'apprendre au système dans quelles conditions il est judicieux pour lui d'interroger l'utilisateur.
— de s'adapter à cette nouvelle connaissance dans le dialogue courant, ainsi que dans les futurs, tout en étant capable de juger de la fiabilité de la nouvelle connaissance. Cette adaptation pourra se traduire dans certains cas par la modification de la base de connaissance du système ou par le ré-entraînement d'un modèle associé au système.

Il faudra de plus prendre en compte le fait que les méthodes citées précédemment impliquent une certaine quantité de données supplémentaires, voire des simulations de données (par exemple des simulations utilisateur). De plus, dans le cadre du LL, ces méthodes et les données associées devront être capable de s'adapter aux nouvelles tâches que le système aura appris à réaliser.

Il faut aussi noter que nous n'avons considéré que des aspects relativement simples du LL appliqué aux systèmes de dialogue orientés tâche, et que nous avons laissé de côté des aspects comme l'amélioration de la qualité du dialogue. Un tel aspect serait par exemple intéressant dans le cas de systèmes uniquement conversationnels.

Remerciements

Cet article est tiré de la traduction de l'article suivant : (Veron *et al.*, 2019). Ce travail a été partiellement financé par le projet LIHLITH (ANR-17-CHR2-0001-03) et soutenu par ERA-Net CHIST-ERA, ainsi que l'Agence Nationale pour la Recherche (ANR).

Références

CHANDRAMOHAN S., GEIST M., LEFEVRE F. & PIETQUIN O. (2011). User simulation in dialogue systems using inverse reinforcement learning. In *Twelfth Annual Conference of the International Speech Communication Association*.

CHEN Z. & LIU B. (2016). *Lifelong Machine Learning*. Morgan & Claypool Publishers.

CHEN Z., LIU B., BRACHMAN R., STONE P. & ROSSI F. (2018). *Lifelong Machine Learning : Second Edition*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers.

HANCOCK B., BORDES A., MAZARE P.-E. & WESTON J. (2019). Learning from dialogue after deployment : Feed yourself, chatbot !

LI J., MILLER A. H., CHOPRA S., RANZATO M. & WESTON J. (2017). Learning through dialogue interactions by asking questions. In *ICLR*.

MALTONI D. & LOMONACO V. (2018). Continuous learning in single-incremental-task scenarios. *arXiv preprint arXiv :1806.08568*.

MAZUMDER S., MA N. & LIU B. (2018). Towards a continuous knowledge learning engine for chatbots. *CoRR*, **abs/1802.06024**.

NEURAZ A., CAMPILLOS LLANOS L., BURGUN A. & ROSSET S. (2018). Natural language understanding for task oriented dialog in the biomedical domain in a low ressources context, nips workshop. In *Machine Learning for Health (ML4H) : Moving beyond supervised learning in healthcare*, Montréal, Québec, Canada.

PAPANGELIS A., KARKALETSIS V. & MAKEDON F. (2012). Evaluation of online dialogue policy learning techniques. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)* : European Language Resources Association (ELRA).

PARISI G. I., KEMKER R., PART J. L., KANAN C. & WERMTER S. (2018). Continual lifelong learning with neural networks : A review. *arXiv preprint arXiv :1802.07569*.

PEÑAS A., VERON M., PRADEL C., OTEGI A., ECHEGOYEN G. & RODRIGO A. (2019). Continuous learning for question answering. In *Tenth International Workshop on Spoken Dialogue Systems Technology (IWSDS)*.

THRUN S. & MITCHELL T. M. (1995). Lifelong robot learning. *Robotics and Autonomous Systems*, **15**, 25–46.

TUR G. & MORI R. D. (2011). *Spoken Language Understanding : Systems for Extracting Semantic Information from Speech*. John Wiley & Sons. Google-Books-ID : RDLyT2FythgC.

VERON M., GHANNAY S., LIGOZAT A.-L. & ROSSET S. (2019). Lifelong learning and task-oriented dialogue system : what does it mean ? In *Tenth International Workshop on Spoken Dialogue Systems Technology (IWSDS)*.

YOUNG S. (2006). Using pomdps for dialog management. In *Spoken Language Technology Workshop, 2006. IEEE*, p. 8–13 : IEEE.

Méthodes de représentation de la langue pour l'analyse syntaxique multilingue

Manon Scholivet
Aix-Marseille Univ, Université de Toulon, CNRS, LIS, Marseille
manon.scholivet@lis-lab.fr

RÉSUMÉ

L'existence de modèles universels pour décrire la syntaxe des langues a longtemps été débattue. L'apparition de ressources comme le *World Atlas of Language Structures* et les corpus des *Universal Dependencies* rend possible l'étude d'une grammaire universelle pour l'analyse syntaxique en dépendances. Notre travail se concentre sur l'étude de différentes représentations des langues dans des systèmes multilingues appris sur des corpus arborés de 37 langues. Nos tests d'analyse syntaxique montrent que représenter la langue dont est issu chaque mot permet d'obtenir de meilleurs résultats qu'en cas d'un apprentissage sur une simple concaténation des langues. En revanche, l'utilisation d'un vecteur pour représenter la langue ne permet pas une amélioration évidente des résultats dans le cas d'une langue n'ayant pas du tout de données d'apprentissage.

ABSTRACT

Language representation methods for multilingual syntactic parsing

The existence of universal models to describe the syntax of languages has been debated for decades. The availability of resources such as the World Atlas of Language Structures and the Universal Dependencies treebanks makes it possible to study the plausibility of universal grammar from the perspective of dependency parsing. Our work consists in studying different language representations in multilingual systems learned on treebanks in 37 languages. We show that representing the language of each word leads to better results compared to simply concatenate all the languages. However, using a vector to represent the language does not increase obviously the results in zero-shot learning cases.

MOTS-CLÉS : Analyse syntaxique multilingue, Représentation de la langue, Traits typologiques, WALS, UD.

KEYWORDS: Multilingual parsing, Language representations, Typological features, WALS, UD.

1 Introduction

L'analyse syntaxique en dépendances requiert, comme de nombreuses autres tâches en TAL, de grandes quantités de données pour apprendre la syntaxe de la langue. Certaines langues comme l'anglais bénéficient d'une grande quantité de données annotées. En revanche, pour de nombreuses langues, il n'en existe actuellement pas. L'analyse syntaxique multilingue est une solution possible pour tenter de tirer profit des langues bien dotées pour apprendre des informations communes entre les langues. Deux problèmes importants se posent alors : comment représenter les données, et comment représenter la langue ?

Avec l'apparition des Universal Dependencies (UD) [1], l'utilisation d'annotations universelles pour la syntaxe permet en partie de répondre au premier problème. En effet, UD propose un ensemble de relations de dépendances, de parties du discours et de traits morphologiques qui se veulent communs à toutes les langues (Nivre *et al.*, 2016). Les corpus arborés de UD sont disponibles pour de nombreuses langues, et sont basés sur un guide d'annotation commun.

Bien que les UD permettent un certain partage entre les langues au niveau des traits universels proposés, il n'existe cependant pas de lexique commun à toutes les langues. Le choix de l'utilisation d'un analyseur delexicalisé (Zeman & Resnik, 2008) a été fait pour ce travail. Cette technique consiste à ignorer le lexique lors de l'entraînement de l'analyseur. Cet appauvrissement des données conduit à des analyseurs moins précis, mais offre une solution simple au problème du lexique [2].

Le problème de représentation de la langue est le cœur de cet article. Trois méthodes de représentations seront présentées. En plus du simple identifiant de la langue, nous verrons comment tenter d'extraire les *paramètres* syntaxiques, premièrement à partir des données, ensuite en utilisant le World Atlas of Language Structures (WALS) [3] (Dryer & Haspelmath, 2013).

Nos contributions consistent à comparer ces différentes méthodes et à évaluer leurs effets dans un cadre d'entraînement multilingue, où le corpus d'entraînement est composé de corpus arboré de 37 langues, puis dans un cadre de *zero-shot*, c'est-à-dire pour une langue n'ayant pas de données d'entraînement (Ammar *et al.*, 2016a; Guo *et al.*, 2015).

Après la présentation de l'état de l'art (Sec. 2), nous décrirons les UD (Sec. 3), nos représentations des langues (Sec. 4), en particulier nous présenterons en détail le WALS (Sec. 4), puis présenterons l'analyseur utilisé (Sec. 5). Le cadre expérimental (Sec. 6) précédera nos résultats (Sec. 7) et nos conclusions (Sec. 8).

2 État de l'art

Ce travail est à l'intersection de trois tendances dans la littérature sur l'analyse syntaxique en dépendances multilingue. La première est le *transfer parsing*, qui consiste à apprendre un analyseur sur une langue (ou un ensemble de langues) puis à le tester sur une autre. La seconde est l' *analyse syntaxique delexicalisée*, qui a pour but de supprimer le lexique. La troisième et dernière tendance est l'utilisation de *ressources typologiques* telles que le WALS.

Le transfer parsing est une bonne solution lorsqu'il faut traiter des langues avec peu de ressources. McDonald *et al.* (2011) décrivent deux types de transfert : la première se base sur des corpus parallèles dont une des deux langues n'a pas, ou pas assez, de données d'entraînement pour apprendre un analyseur, alors que l'autre langue en a. La deuxième, les approches par transfert direct, reposent sur les similarités entre les langues et ne nécessitent pas de corpus parallèles. Par exemple, Lynn *et al.* (2014) proposent un analyseur pour l'irlandais entraîné d'abord sur une autre langue, puis appliqué à l'irlandais. Étonnamment, l'indonésien est la langue donnant les meilleurs résultats, bien qu'elles n'appartiennent pas à la même famille de langues. L'hypothèse des auteurs serait que les dépendances très distantes sont mieux représentées dans l'indonésien que dans les autres langues testées.

1. `http://universaldependencies.org`
2. Une alternative non explorée dans ce travail et que nous comptons explorer à l'avenir est l'utilisation de plongements de mots multilingues (Ammar *et al.*, 2016b)
3. `https://wals.info/`

Les langues pauvrement dotées peuvent parfois avoir une petite quantité de données d'entraînement disponible. En concaténant les corpus d'entraînement de deux langues, on peut obtenir un analyseur bilingue pour vérifier si une amélioration est possible comparé à un analyseur monolingue (Vilares *et al.*, 2015). Les méthodes de transfert direct et les analyseurs bilingues sont proches de cet article, puisqu'on retrouve cette idée de concaténation des corpus d'entraînement. Cependant, dans notre cas, nous combinons les corpus de bien plus de langues (environ 40) et incluons des méthodes de représentation des langues. La combinaison de corpus de multiples langues pour l'entraînement d'un analyseur est facilitée par les récentes avancées sur les standards multilingues et les ressources disponibles, en particulier grâce aux Universal Dependencies pour la syntaxe en dépendances (Nivre *et al.*, 2016). La recherche sur l'analyse syntaxique multilingue est fortement stimulée par des initiatives comme les campagnes d'évaluation CoNLL 2017 et 2018, sur l'analyse syntaxique en dépendances fortement multilingue à partir de textes bruts (Zeman *et al.*, 2017, 2018).

Les analyseurs delexicalisés ignorent la forme superficielle et les lemmes des mots lors de l'analyse d'une phrase, utilisant des traits plus abstrait comme les étiquettes de parties de discours (POS). L'utilisation d'analyseurs delexicalisés est particulièrement pertinente pour l'apprentissage d'analyseurs multilingues, puisque les langues partagent généralement peu de leurs lexiques. L'approche proposée par Zeman & Resnik (2008) consiste à adapter un analyseur pour une nouvelle langue en utilisant soit un corpus parallèle, soit un analyseur delexicalisé. Cette méthode peut être utilisée pour construire rapidement un analyseur si la langue source et la langue cible sont suffisamment proches.

De plus, les traits typologiques comme ceux présents dans le WALS donnent une information sur la structure des langues (Dryer & Haspelmath, 2013). Ces traits peuvent permettre à l'analyseur multilingue d'apprendre des caractéristiques communes à plusieurs langues. Naseem *et al.* (2012) et Zhang & Barzilay (2015) utilisent l'ensemble des traits issus de la catégorie de l'Ordre des mots du WALS qui sont disponibles pour leurs langues. Ponti *et al.* (2018) utilisent plutôt les traits qu'ils jugent pertinents dans les diverses catégories (pas uniquement celle concernant l'ordre des mots).

Täckström *et al.* (2013) utilisent une méthode de transfert multilingue delexicalisée, montrant en quoi le partage de paramètres basé sur des traits typologiques et l'appartenance à une famille de langues peut être utilisé dans un analyseur en dépendances discriminant. Les traits typologiques choisis sont basés sur ceux utilisés par Naseem *et al.* (2012), en retirant deux traits qu'ils ne considèrent pas utiles.

Les travaux les plus proches du notre concatènent des corpus pour entraîner un analyseur multilingue (Ammar *et al.*, 2016a). Les auteurs utilisent un analyseur à transitions S-LSTM similaire au notre (en dehors de l'utilisation de récurrence) entraîné sur un ensemble de traits lexicaux incluant des plongement de mots multilingues, des clusters de Brown, et des étiquettes POS [4] détaillées, alors que nous utilisons seulement des POS plus grossières et des traits morphologiques dans un cadre delexicalisé. Ils utilisent également un vecteur de représentation de la langue encodé en one-hot, l'ensemble de 6 traits issus du travail de Naseem *et al.* (2012), et testent en plus l'utilisation de la matrice complète du WALS. Nous avons testé ces deux premières représentations, ainsi qu'un autre vecteur de dimension 22 issu du WALS également. Leurs expériences ont été réalisées sur sept langues richement dotées alors que nous avons mené les nôtres sur un échantillon beaucoup plus grand de 40 langues. Bien que Ammar *et al.* (2016a) ont montré que, dans un cadre lexicalisé, la concaténation de corpus peut donner des résultats similaires à des analyseurs syntaxiques monolingues, les origines et les limites de ces gains restent floues. Nous explorons entre autres des pistes pour évaluer les avantages des caractéristiques typologiques dans un analyseur delexicalisé.

4. Partie du discours

3 Universal Dependencies

La cohérence de l'annotation des données entre les langues est un problème majeur sur les tâches d'analyse multilingue, la plupart des corpus étant annotés en utilisant différents guides d'annotation et jeux d'étiquettes. L'initiative des Universal Dependencies (UD) a pour but de créer des corpus arborés consistants entre les langues, facilitant ainsi les analyses lors de travaux sur la langue.

Nous utilisons la version 2.0 de UD pour nos corpus d'entraînement de développement [5], et les corpus de tests de la campagne d'évaluation CoNLL 2017 [6]. 64 corpus arborés de UD en 45 langues sont disponibles pour l'entraînement et le développement. Cependant, ces corpus sont de taille très variable (de 529 mots pour le kazakh à 1 1842 867 mots pour le tchèque). Les corpus de tests contiennent au minimum 10 000 mots par langue et sont disponibles pour 49 langues. 4 langues n'ont pas de corpus d'entraînement correspondant.

Les analyseurs delexicalisés, entraînés à partir des corpus de UD, prennent en entrée les parties du discours universelles (UPOS) et les traits morphologiques (FEAT) et prédisent les étiquettes de l'arbre de dépendances (incluant des sous-relations syntaxiques pouvant être spécifique au langage (p. ex., *acl :relcl*)). Les traits morphologiques sont renseignés pour presque tous les corpus, mais présentent de grandes différences. Par conséquent, nous avons fait le choix de ne garder que les traits les plus fréquents, qui apparaissent dans au moins 28 langues. De plus, les traits morphologiques sont représentés comme une liste de paires *(clef, valeur)* que nous avons séparé, afin que chaque paire soit considérée indépendamment, produisant ainsi un ensemble de 16 traits morphologiques par mot.

4 Les différentes représentations de la langue

Dans ce travail, nous utilisons trois méthodes de représentation de la langue. La première (que nous appelerons ID) consiste en un simple identifiant de la langue, sous la forme d'un vecteur one-hot de dimension 37 en entrée de notre analyseur. Les deux autres méthodes sont décrites ci-dessous.

Apprentissage à partir des données Une méthode pour représenter la langue consiste à apprendre un vecteur à partir de nos données d'entraînement, utilisant des statistiques sur les dépendances vues dans les corpus d'entraînement. Deux vecteurs seront appris de cette façon.

Le premier, que nous appellerons W_d, consiste à encoder, pour chaque étiquette des dépendances syntaxiques, la distance moyenne qui sépare les deux éléments de la relation dans une langue. Chaque dépendance d sera encodée sur deux composantes du vecteur W_d : une pour le cas où d est une dépendance droite, et l'autre pour les dépendances gauches.

La deuxième représentation, appelée W_{df}, ajoute à W_d l'information sur la fréquence de chacune de ces dépendances. Ainsi, si il existe n étiquettes de dépendance, W_d sera de taille $2 * n$ et W_{df} de taille $4 * n$. Un exemple de W_d et W_{df} est donné, dans la Table 1.

Ces vecteurs sont appris sur notre corpus d'entraînement (Sec. 6) et ne sont donc pas disponibles pour les langues n'ayant pas de corpus d'entraînement. Ils ne conviennent donc pas aux tests de type zero-shot, contrairement aux vecteurs issus du WALS présentés ci-dessous.

5. http ://hdl.handle.net/11234/1-1983
6. http ://hdl.handle.net/11234/1-2184

6,00	3,36	12,80	12,59	...
L_acl	R_acl	L_advcl	R_advcl	...

6,00	3,36	0,00005	0,02	12,80	...
L_acl	R_acl	freq_L_acl	freq_R_acl	L_advcl	...

TABLE 1 – Exemple de W_d (en haut) et W_{df} (en bas). L_acl représente les dépendances gauches étiquetées acl, dont la distance moyenne entre les éléments et de 6. R_acl représente la même chose, mais pour une dépendance droite. freq_L_acl est la fréquence des dépendances acl gauche dans la langue, et freq_R_acl est la fréquence des dépendances acl droite.

World Atlas of Language Structures Le World Atlas of Language Structures (WALS) est une base de données de propriétés structurelles (phonologiques, grammaticales et lexicales) réunie par 55 auteurs à partir de matériel descriptif comme par exemple des grammaires de référence. Cette base de données nous a permis d'associer à chaque langue des corpus de UD un ensemble de traits décrivant des propriétés pertinentes pour l'analyse syntaxique.

Le WALS décrit 2 676 langues grâce à un ensemble de 192 traits, répartis entre 11 familles (p.ex. Phonologie, Ordre des mots...). Il peut ainsi être représenté via une matrice W de 2 676 lignes et 192 colonnes, où chaque cellule $W(l, f)$ donne la valeur du trait f pour la langue l. Chaque ligne $W(l)$ est le vecteur de trait de la langue l.

Cette matrice a été épurée puis complétée pour correspondre à nos conditions expérimentales. Pour commencer, nous n'avons gardé que les lignes correspondant aux 49 langues de notre corpus de test. En revanche, 4 langues de UD (le Vieux Slave (cu), le Gothique (got), le Grec Ancien (grc), et le Latin (la)) n'apparaissent pas dans le WALS et ont donc été mises de côté. On obtient alors une version réduite de W contenant 45 lignes.

Deux de nos représentations de la langue ont été extraites à partir du WALS. La première, que l'on appellera W_N, est basée sur les travaux de Naseem *et al.* (2012), qui ont sélectionné les 6 traits [7] de la famille de l'Ordre des mots qui étaient entièrement renseignés pour leurs 17 langues cibles. Ces traits couvrent des phénomènes tels que l'ordre verbe-objet ou adjectif-nom, et ont été largement discutés dans la littérature (Täckström *et al.*, 2013; Zhang & Barzilay, 2015; Ammar *et al.*, 2016a). La matrice qui en résulte a 45 lignes (langues) pour 6 colonnes (traits). Cependant, le WALS est une matrice creuse, puisque certains traits ne sont pas renseignés pour certaines langues. C'est pourquoi nous avons fait le choix de ne garder que les langues pour lesquelles au plus la moitié du vecteur n'est pas renseignée, éliminant ainsi 5 langues de plus : le galicien (gl), le haut sorabe (hsb), le kazakh (kk), le slovaque (sk), et le ouïghour (ug). Nos expériences sont effectuées sur cet ensemble de 40 langues.

La deuxième représentation de la langue extraite du WALS, que l'on appellera W_{80}, est une version plus étendue de W_N. Nous nous sommes demandé s'il n'était pas dommage de se restreindre au traits de (Naseem *et al.*, 2012), qui ne gardent que 6 traits. Nous incluons alors dans W_{80} tous les traits renseignés pour au moins 80% de nos 40 langues.

En plus des traits de la famille de l'Ordre des mots, nous avons également inclus ceux de la famille des Propositions simples [8]. Il en résulte une matrice de 40 lignes et 22 colonnes, correspondant à 3 traits de la famille des Propositions simples (101A, 112A, 116A) et 19 traits de celle de l'Ordre des mots (81A, 82A, 83A, 85A, 86A, 87A, 88A, 89A, 90A, 92A, 94A, 95A, 96A, 97A, 144A, 143A, 143E, 143F, 143G).

7. Ces traits sont ceux identifiés par les codes 81A, 85A, 86A, 87A, 88A, 89A dans le WALS.
8. Nous avons également considéré la famille des Phrases Complexes, mais aucun trait ne dépassait le seuil des 80%.

	Romane	Germanique	Slave	Aléatoire
W_N	0.33	1.33	0.67	2.41
W_{80}	4.13	4.47	4.19	10.15

TABLE 2 – MID de chaque famille de langues comparée à l'aléatoire. Aléatoire est la moyenne de 50 000 familles de 6 langues.

Les matrices W_N et W_{80} obtenues ne sont cependant pas complètes : elles contiennent respectivement 4 et 35 valeurs non-renseignées, que nous avons rempli automatiquement. Chaque matrice W (abréviation pour W_N et W_{80}) permet de comparer deux langues l_1 et l_2 de manière simple en utilisant la distance de Hamming [9] entre leur vecteur $W(l_1)$ et $W(l_2)$, noté $d(l_1, l_2)$. Pour remplacer les valeurs manquantes, nous avons sélectionné, pour chaque langue l_1 contenant au moins une valeur non-renseignée ('?'), la valeur correspondante dans le vecteur de la langue l_2 la plus proche qui soit entièrement renseignée, où $l_2 = \underset{l_i \mid \text{“?”} \notin W(l_i)}{\arg\min} d(l_1, l_i)$.

Les matrices W_N et W_{80} ne fournissent qu'une description partielle des langues, fortement biaisée en faveur de l'analyse syntaxique et ignorant les autres aspects (p.ex. la phonologie). Néanmoins, il est tentant de comparer les distances de langues appartenant aux mêmes familles typologiques dans ce mode de représentation. Pour cela, nous nous sommes intéressés à 3 familles présentes dans notre ensemble de 40 langues : les langues Romanes (6 langues), les langues Germaniques (6 langues) et les langues Slaves (7 langues). Nous avons alors calculé la proximité des vecteurs de ces langues. Nous définissons la distance interne moyenne (MID) d'un ensemble de langues $L = \{l_1, \ldots, l_n\}$, comme la moyenne des distances de chaque paire dans L :

$$MID(L) = \frac{1}{n^2 - n} \sum_{\substack{(l_i, l_j) \in L \times L \\ i \neq j}} d(l_i, l_j)$$

Nous avons calculé le MID de chaque famille de langues, et l'avons comparé au MID d'ensembles aléatoires de 6 langues (ce qui correspond au nombre de langues des familles Germanique et Romane). Les résultats de la Table 2 montrent clairement que les vecteurs du WALS permettent de capturer des similarités au sein d'une famille de langues, puisque le MID des vecteurs de langues d'une même famille est nettement inférieur au simple hasard.

5 Analyseur

L'analyseur utilisé dans nos expériences est un analyseur par transition de type *arc-eager* (Nivre, 2008), entraîné avec un oracle dynamique (Goldberg & Nivre, 2012). La prédiction des transitions est faite par un perceptron multi-couches (MLP) similaire au système de Chen & Manning (2014), consistant en une couche d'entrée, une couche cachée et une couche de sortie. Deux ensembles de traits entièrement delexicalisés ont été définis pour la prédiction : BASIQUE et ÉTENDU. BASIQUE est un ensemble classique composé de 9 traits liés au POS, 7 traits syntaxiques, 32 traits morphologiques et un trait de distance (la distance entre la tête et le dépendant). [10] L'ensemble ÉTENDU ajoute à BASIQUE de nouveaux traits correspondant aux vecteurs du WALS W_N et W_{80}, et/ou l'identifiant

9. Le nombre de dimensions pour lesquelles les valeurs diffèrent.
10. Nos corpus, le code source de l'analyseur, les fichiers de configuration et les matrices issues du WALS seront disponibles.

de la langue. Chaque trait est associé à un plongement de taille 3, initialisé à zéro. L'identifiant de la langue (issu de la représentation ID) est un vecteur one-hot de dimension 37 (correspondant aux 37 langues uniques). La couche d'entrée du MLP correspond à la concaténation des plongements des différents traits, dont les dimensions varient de 396 à 465 selon la configuration (avec ou sans vecteurs de langue W_N et W_{80}, ou de l'identifiant de la langue ID). La couche de sortie est composée de 263 neurones, correspondant au nombre de transitions que l'analyseur peut prédire. La couche cachée est de taille 1 000, avec un *dropout* durant l'entraînement de 0.4, le nombre d'itérations est égal à 10, la fonction d'activation est la fonction ReLu, la fonction objectif est un softmax de vraisemblance négative, et l'algorithme d'apprentissage est AMSgrad, utilisant les paramètres par défaut de Dynet (Neubig *et al.*, 2017). [11]

À chaque étape du processus d'analyse syntaxique, l'analyseur prédit une action à réaliser, pouvant aboutir à la création d'une nouvelle dépendance entre deux mots de la phrase. La prédiction des actions est basée sur la valeur des traits donnés au MLP. Dans la configuration BASIQUE, ces traits décrivent différents aspects du gouverneur, du dépendant, et du contexte. Par exemple, si la tête est un verbe et que le dépendant est un nom situé avant le verbe, une dépendance sujet aura une forte probabilité d'être prédite pour les langues utilisant majoritairement l'ordre sujet-verbe (SV). Avec l'ensemble de traits ÉTENDU, l'utilisation de l'ordre SV pour une langue est explicite. Le MLP a donc la possibilité de combiner une *configuration de phrastique* (p.ex., un nom avant un verbe) avec une *configuration de la langue* (p.ex., la langue est SV) quand il prédit une action. Les langues partageant un trait dans W auront la possibilité de générer la même prédiction pour une configuration phrastique correspondant à ce trait (p.ex. le nom précédant le verbe et la langue est de type SV). [12]

6 Cadre expérimental

Corpus Nos expériences ont été réalisées sur les données de la campagne d'évaluation CoNLL 2017 (Zeman *et al.*, 2017), en utilisant la tokenisation de référence et en ignorant les contractions (p.ex. *du*=*de*+*la*). Nos modèles sont évalués individuellement sur chacune des 40 langues pour lesquelles nous avons un vecteur $W(l)$ (Sec. 4), en utilisant les corpus de test de la campagne d'évaluation pour faciliter la comparaison avec les travaux similaires. Le corpus de test pour chaque langue est obtenu en faisant la concaténation de tous les corpus de test disponibles pour cette langue.

Trois langues n'ont pas de corpus d'entraînement ou de développement (bxr, kmr, sme). L'entraînement et le développement se font sur des corpus multilingues (ML) dérivés des 37 langues de UD restantes, que l'on nommera TRAIN-ML et DEV-ML. La taille des corpus de UD peut fortement varier d'une langue à l'autre, allant de 529 mots pour le Kazakh (kk) à 1 842 867 pour le Tchèque (cs). Ainsi, concaténer simplement tous les corpus pour constituer TRAIN-ML et DEV-ML sur-représenterait certaines langues et introduirait un biais en leur faveur. C'est pourquoi nous avons décidé d'équilibrer le nombre de tokens de TRAIN-ML et DEV-ML entre les langues.

La construction des corpus de développement et d'entraînement se fait en deux étapes. Tout d'abord

11. Les valeurs des hyperparamètres ont été définies dans des conditions similaires à $\Sigma\ W_N$, cf. Section 6).

12. Notre analyseur ne peut pas prédire d'arbre non-projectif. La présence d'une dépendence non-projective génère systématiquement une erreur d'analyse lors de la phase de test. Le taux moyen de projectivité du corpus de test est égal à 1%, avec un écart-type de 1% pour les 40 langues. 14 corpus (pour 20 langues) ont un taux de non-projectivité inférieur à 1%, et le taux maximum est de 8% pour le corpus du néerlandais (corpus Lassysmal). Nous avons fait des tests utilisant une transformation en arbre pseudo-projectif (Nivre & Nilsson, 2005), mais l'impact sur les résultats étant négligeable, nous avons choisi de garder l'algorithme projectif original.

la totalité des données des corpus disponibles (entraînement et développement) est divisée en deux à hauteur de 10% pour un sous corpus de développement et le reste pour l'entraînement. Ensuite, pour chacun des sous-corpus précédents, on sélectionne de façon aléatoire des phrases pour chaque langue jusqu'à ce que les corpus finaux atteignent une certaine limite en terme de tokens (respectivement 2 000 pour le DEV et 20 000 pour le TRAIN). Enfin les données ainsi sélectionnées sont mélangées (pour chacun des corpus) afin d'éviter de conserver l'ordre des langues. Avec cette procédure, la même phrase peut apparaître plusieurs fois. Néanmoins, cette approche garantit une représentation équilibrée de chaque langue dans TRAIN-ML et DEV-ML

Métrique La qualité des arbres prédits est évaluée par une mesure standard pour l'analyse syntaxique en dépendance : le score d'attachement de l'étiquette **AN étiqueté**(LAS). [13, 14] On donne le LAS par langue, ainsi que le MACRO-LAS, qui est la macro-moyenne du LAS de toutes les langues qui ont un corpus d'entraînement. Cette mesure est indépendante de la taille du corpus de test de chaque langue, et n'est pas biaisée en faveur des langues sur-représentées dans l'ensemble de test.

Configuration d'entraînement Nos expériences sur plusieurs paires ⟨corpus d'entraînement, représentation de la langue⟩ sont désignées par les codes suivants :

L : Corpus monolingue. Le corpus d'entraînement de la langue l consiste simplement à prendre toutes les phrases de la langue l dans TRAIN-ML. 37 analyseurs avec la configuration BASIQUE ont été entraînés, un pour chaque langue. Cette configuration correspond à la situation standard des expériences en analyse syntaxique : entraînement et test sur une même langue.

Σ : Corpus multilingue. Un analyseur est entraîné sur l'intégralité de TRAIN-ML, sans indication sur la langue. Le modèle est delexicalisé, le corpus ne contient donc que les étiquettes POS de référence, les traits morphologiques de référence, et les relations syntaxiques à apprendre.

Σ ID : Corpus multilingue + ID de la langue. Un analyseur entraîné avec la configuration ÉTENDU sur TRAIN-ML, en utilisant, attaché à chaque mot, l'identifiant de la langue.

$\Sigma\ W_N$, $\Sigma\ W_{80}$: Corpus + WALS. Deux analyseurs entraînés avec l'ensemble de traits ÉTENDU, sur TRAIN-ML. L'entraînement de $\Sigma\ W_N$ (resp. W_{80}) ajoute à chaque mot de Σ un vecteur $W(l)$ issu du WALS, qui correspond à la langue du mot. Un seul modèle est entraîné pour $\Sigma\ W_N$ (resp. W_{80}).

$\Sigma\ W_d$, $\Sigma\ W_{df}$: Corpus + vecteurs de données. Deux analyseurs entraînés avec l'ensemble de traits ÉTENDU, sur TRAIN-ML, avec W_d (respectivement W_{df}) attaché à chaque mot. Cet ajout est fait de la même façon que pour les tests $\Sigma\ W_N$ et $\Sigma\ W_{80}$.

$\overline{L}$ **:** *Zero-shot*. Dans cette configuration, 37 corpus d'entraînements sont dérivés de TRAIN-ML : pour chaque langue l, un corpus d'entraînement est construit à partir de toutes les phrases de TRAIN-ML, sauf celles appartenant à la langue l. Cette configuration représente la situation où un analyseur dans la configuration BASIQUE est entraîné pour une langue pour laquelle aucune donnée d'entraînement

13. Nous avons omis le score UAS car le UAS et le LAS sont étroitement corrélés ($r = 0.98$).
14. Nous avons utilisé le script d'évaluation de la campagne d'évaluation CoNLL 2017.

n'est disponible, comme dans les méthodes de transfert direct.

$\overline{L}\ W_{80}$ **:** $\overline{L}\ W_{80}$ ajoute à $\overline{L}$ l'indication de la langue de chaque mot à travers son vecteur du WALS, de la même manière que $\Sigma\ W_{80}$ faisait avec Σ.

7 Résultats et analyse

Nos expériences ont été réalisées dans les configurations décrites ci-dessus. Le LAS est donné pour chaque langue, ainsi que la macro-moyenne du LAS (MACRO), dans la Table 3. Nous commentons ci-dessous les résultats pour *L*, et comparons les résultats de certaines expériences (voir Table 4).

***L* :** Les résultats de l'expérience *L* montrent une importante variation des performances selon les langues. Le LAS varie de 46,78 pour le turc à 81,44 pour l'italien. Plus d'expériences seraient nécessaires pour expliquer les raisons d'une telle variabilité, mais n'entrent pas dans le cadre de ce travail. Nous pouvons tout de même émettre certaines hypothèses. Tout d'abord, certaines particularités spécifiques aux langues, comme l'équilibre entre les marqueurs morphologiques et syntaxiques (p.ex. les langues morphologiquement riches sont probablement favorisées dans notre configuration, puisque l'analyse morphologique est donnée en entrée de l'analyseur). D'autres sont spécifiques au genre textuel. Bien que la delexicalisation permette de neutraliser certains biais de genre, le genre peut aussi influencer la syntaxe, notamment la longueur des phrases (or, les phrases longues sont généralement plus dures à analyser), ou encore la proportion de constructions difficiles, comme les prépositions ou les coordinations ambiguës. Enfin, l'hétérogénéité de la qualité des annotations selon les langues peut également expliquer la variabilité du LAS.

***L* vs Σ :** Une chute attendue des performances est observée lorsque l'on passe de *L* à Σ. Le MACRO LAS chute de 5,68 points. L'hypothèse principale pour expliquer cette chute est le bruit qui est introduit par le mélange des langues. Concaténer toutes les phrases de plusieurs langues sans préciser l'identité de la langue introduit du bruit dans l'analyseur. Par exemple, la configuration phrastique associée à une dépendance sujet dans une langue SV ou VS sera très différente, et l'analyseur n'est pas capable de faire la distinction entre ces langues et voit donc des contradictions. La chute du LAS est cependant très variable selon les langues, allant même jusqu'à une augmentation plutôt qu'une chute dans le cas de l'espagnol (+0,51 points). Nous n'avons pas actuellement d'explication pour ce résultat, tout au plus une intuition qui serait que Σ apprendrait implicitement une langue moyenne (bruitée) qui serait plus proche de l'espagnol que du chinois par exemple (dont le LAS chute de 14,37 points), les langues composant Σ étant plus proches de l'espagnol que du chinois en moyenne.

Σ vs $\Sigma\ W_{80}$, $\Sigma\ W_d$, ID :[15] Nous obtenons ici le premier résultat important de ce travail : lors de l'ajout d'une représentation de la langue à l'analyseur, la MACRO LAS augmente de 4 à 4,75 points comparé à Σ. Le LAS augmente pour toutes les langues, pour les trois expériences. Ajouter l'information ID à Σ permet une augmentation des résultats MACRO de 4,60 points. Cette augmentation était attendue puisque dans ce test, les configurations phrastique sont associées à l'ID de la langue, ce qui aide à diminuer le bruit dans les données. Deux interprétations sont possibles pour ΣW_N : celle optimiste serait que les vecteurs de représentations de la langue aident à diminuer le bruit

15. W_N et W_{df} sont analysés plus tard.

L	Σ	Σ ID	$\Sigma\, W_N$	$\Sigma\, W_{80}$	$\Sigma\, W_d$	$\Sigma\, W_{df}$	$\overline{L}$	$\overline{L}\, W_{80}$	Lang.	
65,89	60,59	64,04	63,15	64,38	64,08	63,29	27,50	34,34	ar	
78,59	74,32	79,25	76,26	77,47	78,28	78,94	67,05	63,73	bg	S
77,18	72,76	76,63	73,03	76,27	76,90	77,17	70,48	68,88	ca	R
68,92	68,01	69,41	68,72	69,61	69,75	69,67	62,08	59,47	cs	S
73,62	67,38	70,26	70,19	70,25	70,57	72,19	61,56	63,87	da	G
71,07	63,76	70,36	69,18	69,22	70,37	70,44	59,65	60,51	de	G
77,11	71,26	76,16	73,29	75,84	76,11	76,90	63,74	65,96	el	
70,05	66,02	71,17	69,91	70,19	70,78	71,21	60,87	62,11	en	G
71,47	71,98	73,83	72,29	73,22	73,89	74,12	71,40	70,76	es	R
66,98	63,76	68,41	65,75	67,79	67,94	69,08	58,52	58,77	et	
63,26	55,76	60,11	60,22	59,39	60,00	60,31	33,50	31,68	eu	
72,85	66,02	69,50	69,63	70,00	69,67	68,02	31,25	34,27	fa	
60,97	56,29	59,65	57,37	59,28	59,99	59,92	50,69	48,72	fi	
75,74	74,25	76,16	74,79	75,82	75,80	76,50	72,68	71,97	fr	R
66,55	60,41	66,86	64,68	65,96	66,02	66,30	43,51	42,75	ga	
70,21	63,03	67,97	66,09	67,45	67,19	68,33	51,36	52,98	he	
78,91	73,86	74,86	75,77	74,45	76,66	76,52	54,23	57,33	hi	
71,03	67,49	71,37	70,00	70,40	71,42	71,28	62,88	65,71	hr	S
67,08	62,55	66,84	67,19	67,51	67,79	68,14	48,24	49,62	hu	
68,64	58,38	63,98	62,61	64,57	64,31	64,57	45,57	43,03	id	
81,44	76,45	80,56	76,97	79,83	80,45	81,73	75,82	77,04	it	R
78,26	68,22	75,81	74,85	75,56	76,59	76,89	7,64	28,53	ja	
47,68	37,17	38,61	38,07	39,66	41,11	39,36	17,99	23,18	ko	
59,89	54,11	59,98	58,23	60,17	59,78	60,86	44,41	47,38	lv	
62,56	57,21	59,28	58,13	58,59	59,11	60,00	51,11	48,98	nl	G
74,59	73,19	76,95	73,51	75,93	77,67	76,75	53,09	55,49	no	G
81,24	74,24	80,52	74,78	79,02	80,44	80,06	69,15	70,72	pl	S
72,00	65,74	69,83	68,74	69,86	69,88	69,96	62,85	65,86	pt	R
70,99	67,72	71,47	70,38	70,61	71,26	71,19	55,84	61,01	ro	R
74,06	61,35	74,72	68,65	74,45	74,92	75,06	55,09	55,50	ru	S
67,10	64,12	66,44	64,75	66,36	66,63	66,69	60,52	63,26	sl	S
72,05	69,97	72,55	70,71	71,88	73,21	73,25	64,80	67,05	sv	G
46,78	41,01	43,16	43,26	41,62	45,73	43,57	29,46	30,33	tr	
71,60	69,40	75,30	69,77	72,81	74,05	74,56	67,08	64,92	uk	S
74,35	69,15	70,93	70,71	70,76	72,41	71,47	58,93	59,41	ur	
54,40	42,42	53,75	51,94	51,72	52,81	52,42	25,86	41,07	vi	
59,83	45,46	58,48	53,42	54,87	56,95	59,19	22,24	24,48	zh	
69,32	63,64	68,25	66,41	67,64	68,39	68,54	51,86	53,80	MACRO	
-	33,32	34,10	30,37	28,49	-	-	-	-	bxr	
-	40,34	37,20	41,41	44,04	-	-	-	-	kmr	
-	47,34	45,60	47,63	42,38	-	-	-	-	sme	

TABLE 3 – LAS pour chaque langue, et MACRO LAS, pour les 9 configurations. Les langues suivies d'un S appartiennent à la famille des langues Slave, G appartiennent à la famille des langues Germaniques et R appartiennent à la famille des langues Romanes.

X	Y	$\overline{X-Y}$	σ	min	max
L	Σ	5.68	3.32	-0.51 es	14.37 zh
$\Sigma\ W_{80}$	Σ	4.00	2.58	0.59 hi	13.10 ru
Σ ID	Σ	4,60	2,91	1,00 hi	13,37 ru
$\Sigma\ W_d$	Σ	4,75	2,57	1,55 fr	13,57 ru
$\Sigma\ W_{80}$	$\Sigma\ W_N$	1.24	1.45	-1.64 tr	5.80 ru
Σ ID	$\Sigma\ W_{80}$	0,61	0,91	-1,05 ko	3,61 zh
$\Sigma\ W_d$	Σ ID	0,14	0,89	-1,53 zh	2,57 tr
$\Sigma\ W_{df}$	$\Sigma\ W_d$	0,15	0,89	-2,16 tr	2,24 zh
L	$\overline{L}$	17,47	13,57	0,07 es	70,62 ja
$\overline{L}\ W_{80}$	$\overline{L}$	1,95	4,60	-3,32 bg	20,89 ja

TABLE 4 – Différence entre les configurations X et Y : moyenne ($\overline{X-Y}$), écart type (σ), minimum et maximum avec la langue correspondante.

introduit par le mélange des langues dans Σ en "expliquant" certaines informations contradictoires dans les données grâce à l'utilisation des traits linguistiques encodés dans le WALS pour W_{80}, et grâce aux informations de distance contenues dans W_d. L'interprétation pessimiste consiste à dire que ces vecteurs sont un simple encodage arbitraire des langues. Dans ce cas, le MLP de l'analyseur apprendrait à associer les configurations phrastiques à certaines langues spécifiquement, et apprendrait alors différents modèles pour différentes langues, ce qui reviendrait à l'expérience ID. Plus d'expériences sont prévues pour comprendre comment le MLP utilise les vecteurs W.

$\Sigma\ W_N$ **vs** $\Sigma\ W_{80}$ **:** Les vecteurs W_N et W_{80} n'ont pas le même impact lorsqu'on les ajoute à Σ. L'ajout de W_{80} permet l'augmentation de 4 points de la mesure MACRO tandis que l'ajout de W_N augmente le score MACRO par rapport au MACRO obtenu avec Σ de 2,77 points seulement. L'analyseur est donc capable de tirer profit d'une description des langues plus riche lors de son apprentissage. Ce résultat pourrait indiquer que les résultats décevants sur l'analyse syntaxique rapportés par Ammar *et al.* (2016a), qui utilisaient le vecteur W_N, pourraient venir des traits extraits du WALS qui n'étaient pas suffisamment riches pour expliquer à l'analyseur les différences syntaxiques importantes entre les langues.

Σ ID **vs** $\Sigma\ W_{80}$ **:** Malgré de meilleurs résultats obtenus en utilisant W_{80} plutôt que W_N, ce nouveau vecteur n'est toujours pas capable de surpasser les performances de l'utilisation d'un simple identifiant de la langue. Mais l'augmentation de la mesure MACRO entre W_N et W_{80} peut laisser supposer que le choix des traits du WALS influence fortement les résultats. Plus d'expériences sont nécessaires pour confirmer cette hypothèse. On pourra également se poser la question de l'influence de la méthode de remplacement des valeurs inconnues.

Σ ID **vs** $\Sigma\ W_d$ **:** Le résultat de ces expériences représentent le deuxième résultat majeur de ce travail : le vecteur W_d permet de battre les résultats de Σ ID. Même si l'hypothèse qu'une partie du vecteur W_d sert a représenter la langue s'avère juste, une partie du modèle a été capable d'apprendre des informations supplémentaires, partagées entre les différentes langues. Ce résultat est tout de même à nuancer, puisque le LAS de $\Sigma\ W_d$ n'est supérieur que de 0,14 points à celui de Σ ID.

$\Sigma\ W_d$ **vs** $\Sigma\ W_{df}$ **:** Les résultats de $\Sigma\ W_d$ et de $\Sigma\ W_{df}$ sont assez proches (+0,15 points pour la mesure MACRO de W_{df}). L'information sur la fréquence des dépendances permet donc l'apprentissage de

connaissances supplémentaires pour l'analyseur. Cependant, le temps d'apprentissage de $\Sigma\, W_{df}$ étant assez conséquent, nous avons fait le choix de nous concentrer sur les résultats de $\Sigma\, W_d$. Le vecteur W_{df} étant deux fois plus gros que W_d, on peut également supposer que la taille de la couche cachée du MLP de l'analyseur n'est pas suffisante pour utiliser toutes les informations du vecteur W_{df}.

***L* vs $\overline{L}$:** $\overline{L}$ correspond à des conditions extrêmes mais également plus réalistes. La situation simulée est celle où aucune donnée d'entraînement n'est disponible pour la langue l. La chute des performances comparée à *L* était prévisible, mais n'en reste pas moins dramatique : la MACRO LAS chute de 17,47 points. Cette chute varie énormément selon les langues (cela se ressent sur l'écart type qui atteint 13,57 points). Certaines langues ne sont presque pas affectées, comme l'espagnol qui ne perd que 0,07 point. En revanche, le japonais perd 70,62 points de LAS, la plus grosse chute toutes expériences et langues confondues. Les langues les plus isolées sont celles qui souffrent le plus du passage à $\overline{L}$. Les familles de langues sont moins affectées, avec une chute de "seulement" 6,65 points en moyenne pour les langues romanes par exemple.

$\overline{L}$ vs $\overline{L}\ W_{80}$: Cette comparaison constitue notre troisième et dernier résultat majeur. Cette expérience consiste à observer si l'ajout d'un vecteur issu du WALS permet de limiter la chute des score LAS de l'expérience $\overline{L}$. Comme on peut le voir dans les Tables 3 et 4, l'ajout de W_{80} permet d'augmenter la MACRO LAS de 1,95 points, ce qui pourrait laisser penser que W_{80} permet bien d'aider le modèle à gérer les langues inconnues. Cependant, on remarque que l'écart type est assez élevé (4,60 points). En effet, lorsque l'on regarde le détail langue par langue, 11 de nos 37 langues ont de moins bons résultats avec le modèle $\overline{L}\ W_{80}$ qu'avec $\overline{L}$. Le bulgare perd même 3,32 points de LAS. Les langues ayant des résultats extrêmement bas dans $\overline{L}$ ont réussi à tirer profit du vecteur W_{80}, comme le japonais qui, après son importante chute au passage à $\overline{L}$, remonte de 20,89 points avec $\overline{L}\ W_{80}$. Le vecteur W_{80} reste une description extrêmement partielle de la langue, et il est probable qu'il ne soit pas suffisant pour compenser l'absence de corpus d'entraînement. La question précédemment soulevée se pose à nouveau : le vecteur sert il simplement d'identifiant de la langue ? Si l'analyseur utilise, ne serait-ce qu'en partie, W_{80} pour détecter la langue de la phrase analysée, l'absence de cette langue dans le corpus d'entraînement rend la tâche impossible.

8 Conclusions et travaux futurs

Dans ce travail, nous avons analysé des représentations possibles d'une langue dans le cadre d'une analyse en dépendances multilingue. Les meilleurs résultats sont obtenus avec les représentations apprises sur les données (W_d,W_{df}), bien qu'ils dépassaient de peu la représentation par un simple identifiant (ID). Mais ces représentations avaient l'inconvénient de ne pas être utilisables dans le cadre d'une langue pour laquelle il n'existe pas de données d'apprentissage, contrairement aux représentations utilisant le WALS. L'utilisation de ce vecteur ne permettant cependant pas une amélioration nette des résultats pour toutes les langues lors d'un d'apprentissage de type *zero-shot*.

Pour les travaux futurs, nous prévoyons de faire une analyse des poids du réseau afin de tester l'hypothèse selon laquelle W est utilisé comme un simple identifiant de la langue. Nous aimerions également tester d'autres méthodes de remplacement des valeurs non-renseignées du WALS, ainsi que la création d'un W_{de}, qui utiliserait l'écart type plutôt que la fréquence. Une autre étape sera l'introduction de plongements de mots multilingues pour donner des informations lexicales à l'analyseur.

Références

AMMAR W., MULCAIRE G., BALLESTEROS M., DYER C. & SMITH N. A. (2016a). Many Languages, One Parser. *arXiv :1602.01595 [cs]*. arXiv : 1602.01595.

AMMAR W., MULCAIRE G., TSVETKOV Y., LAMPLE G., DYER C. & SMITH N. A. (2016b). Massively Multilingual Word Embeddings. *arXiv :1602.01925 [cs]*. arXiv : 1602.01925.

CHEN D. & MANNING C. (2014). A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, p. 740–750 : Association for Computational Linguistics.

M. S. DRYER & M. HASPELMATH, Eds. (2013). *WALS Online*. Leipzig : Max Planck Institute for Evolutionary Anthropology.

GOLDBERG Y. & NIVRE J. (2012). A dynamic oracle for arc-eager dependency parsing. *Proceedings of COLING 2012*, p. 959–976.

GUO J., CHE W., YAROWSKY D., WANG H. & LIU T. (2015). Cross-lingual Dependency Parsing Based on Distributed Representations. In *ACL (1)*, p. 1234–1244.

LYNN T., FOSTER J., DRAS M. & TOUNSI L. (2014). Cross-lingual Transfer Parsing for Low-Resourced Languages : An Irish Case Study. In *Proceedings of the First Celtic Language Technology Workshop*, p. 41–49, Dublin, Ireland : Association for Computational Linguistics and Dublin City University.

MCDONALD R., PETROV S. & HALL K. (2011). Multi-source transfer of delexicalized dependency parsers. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, p. 62–72, Edinburgh, Scotland, UK. : Association for Computational Linguistics.

NASEEM T., BARZILAY R. & GLOBERSON A. (2012). Selective sharing for multilingual dependency parsing. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 629–637, Jeju Island, Korea : Association for Computational Linguistics.

NEUBIG G., DYER C., GOLDBERG Y., MATTHEWS A., AMMAR W., ANASTASOPOULOS A., BALLESTEROS M., CHIANG D., CLOTHIAUX D., COHN T., DUH K., FARUQUI M., GAN C., GARRETTE D., JI Y., KONG L., KUNCORO A., KUMAR G., MALAVIYA C., MICHEL P., ODA Y., RICHARDSON M., SAPHRA N., SWAYAMDIPTA S. & YIN P. (2017). Dynet : The dynamic neural network toolkit. *arXiv preprint arXiv :1701.03980*.

NIVRE J. (2008). Algorithms for deterministic incremental dependency parsing. *Computational Linguistics*, **34**(4), 513–553.

NIVRE J., DE MARNEFFE M.-C., GINTER F., GOLDBERG Y., HAJIC J., MANNING C. D., MCDONALD R., PETROV S., PYYSALO S., SILVEIRA N., TSARFATY R. & ZEMAN D. (2016). Universal dependencies v1 : A multilingual treebank collection. In N. C. C. CHAIR), K. CHOUKRI, T. DECLERCK, S. GOGGI, M. GROBELNIK, B. MAEGAARD, J. MARIANI, H. MAZO, A. MORENO, J. ODIJK & S. PIPERIDIS, Eds., *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France : European Language Resources Association (ELRA).

NIVRE J. & NILSSON J. (2005). Pseudo-projective dependency parsing. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, p. 99–106 : Association for Computational Linguistics.

PONTI E. M., REICHART R., KORHONEN A. & VULIĆ I. (2018). Isomorphic Transfer of Syntactic Structures in Cross-Lingual NLP. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 1531–1542, Melbourne, Australia : Association for Computational Linguistics.

TÄCKSTRÖM O., MCDONALD R. & NIVRE J. (2013). Target language adaptation of discriminative transfer parsers. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 1061–1071 : Association for Computational Linguistics.

VILARES D., GÓMEZ-RODRÍGUEZ C. & ALONSO M. A. (2015). One model, two languages : training bilingual parsers with harmonized treebanks. *arXiv :1507.08449 [cs]*. arXiv : 1507.08449.

ZEMAN D., POPEL M., STRAKA M., HAJIC J., NIVRE J., GINTER F., LUOTOLAHTI J., PYYSALO S., PETROV S. & POTTHAST M. (2017). CoNLL 2017 shared task : multilingual parsing from raw text to universal dependencies. *Proceedings of the CoNLL 2017 Shared Task : Multilingual Parsing from Raw Text to Universal Dependencies*, p. 1–19.

ZEMAN D., POPEL M., STRAKA M., HAJIC J., NIVRE J., GINTER F., LUOTOLAHTI J., PYYSALO S., PETROV S., POTTHAST M., TYERS F., BADMAEVA E., GOKIRMAK M., NEDOLUZHKO A., CINKOVA S., HAJIC JR. J., HLAVACOVA J., KETTNEROVÁ V., URESOVA Z., KANERVA J., OJALA S., MISSILÄ A., MANNING C. D., SCHUSTER S., REDDY S., TAJI D., HABASH N., LEUNG H., DE MARNEFFE M.-C., SANGUINETTI M., SIMI M., KANAYAMA H., DEPAIVA V., DROGANOVA K., MARTÍNEZ ALONSO H., ÇÖLTEKIN , SULUBACAK U., USZKOREIT H., MACKETANZ V., BURCHARDT A., HARRIS K., MARHEINECKE K., REHM G., KAYADELEN T., ATTIA M., ELKAHKY A., YU Z., PITLER E., LERTPRADIT S., MANDL M., KIRCHNER J., ALCALDE H. F., STRNADOVÁ J., BANERJEE E., MANURUNG R., STELLA A., SHIMADA A., KWAK S., MENDONCA G., LANDO T., NITISAROJ R. & LI J. (2018). CoNLL 2018 Shared Task : Multilingual Parsing from Raw Text to Universal Dependencies. In *Proceedings of the CoNLL 2017 Shared Task : Multilingual Parsing from Raw Text to Universal Dependencies*, p. 1–19, Vancouver, Canada : Association for Computational Linguistics.

ZEMAN D. & RESNIK P. (2008). Cross-Language Parser Adaptation between Related Languages. In *Proceedings of the IJCNLP-08 Workshop on NLP for Less Privileged Languages* : Association for Computational Linguistics.

ZHANG Y. & BARZILAY R. (2015). Hierarchical low-rank tensors for multilingual transfer parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, p. 1857–1867, Lisbon, Portugal : Association for Computational Linguistics.

Parsing des textes journalistiques en serbe par le logiciel *Talismane*

Dusica Terzic[1]
(1) Faculté de philologie, Université de Belgrade, Studentski trg 3, 11 000 Belgrade, Serbie
duterzic@gmail.com

RÉSUMÉ

Cet article présente la création d'un treebank journalistique serbe, *ParCoJour*. Il est composé de 30K tokens et doté de trois couches d'annotation : étiquetage morphosyntaxique, lemmatisation et annotation syntaxique. Une fois construit, *ParCoJour* a été utilisé dans trois expériences afin d'évaluer l'impact du domaine textuel sur le parsing du serbe en comparant les performances de *Talismane*, un système par apprentissage automatique, sur deux types de corpus, journalistique et littéraire : 1) parsing du corpus journalistique avec un modèle entraîné sur le corpus journalistique ; 2) parsing du corpus journalistique avec un modèle entraîné sur le corpus littéraire ; 3) parsing du corpus littéraire avec un modèle entraîné sur le corpus journalistique. Les résultats sont comparés à ceux où les deux corpus relevaient du domaine littéraire. Le changement de domaine textuel dans la deuxième et la troisième expérience entraîne une baisse des performances, mais les résultats de parsing restent satisfaisants.

ABSTRACT

Parsing of newspaper texts in Serbian using *Talismane*

This article presents the creation of a newspaper treebank for Serbian. The corpus contains 30K tokens and it is tagged, lemmatised and parsed. After its creation, the corpus was used in three experiments the goal of which was to evaluate the impact of text domain on the automatic processing of Serbian by comparing the performance of *Talismane*, a supervised machine learning system, on two types of corpora, journalistic and literary: 1) parsing of the newspaper corpus with the model trained on the same corpus; 2) parsing of the newspaper corpus with the model trained on the literary corpus; 3) parsing of the literary corpus with the model trained on the newspaper corpus. The results are compared with the results of the experiment in which both corpora were literary. The change of domain in the second and third experiment resulted in a drop of the parsing results, but they remained satisfactory.

MOTS-CLÉS : Parsing, corpus d'entraînement, serbe, adaptation de domaine.

KEYWORDS: Parsing, training corpus, Serbian, domain adaptation.

1 Introduction

Le serbe, la langue de notre étude, fait partie des langues slaves du sud. Il est parlé en Serbie et dans les pays voisines par presque 9 millions de personnes (Fagard et al., 2017). C'est une langue à ordre de mots flexible et à morphologie flexionnelle riche. Il figure dans le groupe des langues

relativement peu dotées en ressources du TAL. Certaines de ces ressources sont dotées de trois couches d'annotation : l'annotation morphosyntaxique, la lemmatisation, l'annotation syntaxique. Néanmoins, les ressources analysées au niveau syntaxique sont les moins nombreuses. Pour contribuer au développement des ressources serbes dotées d'annotation syntaxique, nous décidons de mettre accent sur le parsing dans notre recherche. L'état de l'art en parsing du serbe est atteint avec *Talismane,* un logiciel statistique à base de transitions qui traite les arbres en dépendances. Pour cette raison, dans cet article, l'accent est mis sur les systèmes par apprentissage automatique bien que d'autres techniques de traitement automatique des langues naturelles existent, et qu'elles donnent parfois de bons résultats en étiquetage morphosyntaxique et en lemmatisation du serbe (voir Gesmundo, Samardžić, 2012).

Les ressources existantes et librement disponibles relèvent d'un domaine textuel restreint, à savoir littéraire (*1984* (Krstev et al., 2004) ; *ParCoTrain-Synt* (Miletic, 2018)) et journalistique (*SETimes.SR* (Samardžić et al., 2017)). Ce manque de diversité de corpus disponibles affecte la qualité des résultats obtenus dans le cas d'apprentissage supervisé. Afin de contribuer à la diversification des corpus disponibles, nous avons défini notre premier objectif : constituer une nouvelle ressource du TAL pour le serbe, enrichie d'annotations linguistiques à plusieurs niveaux et provenant d'un domaine textuel sous-représenté pour le moment, à savoir le domaine journalistique. Or, la création des ressources du TAL exige des investissements humains et matériels considérables. Pour faciliter la création de la nouvelle ressource, nous nous sommes servi des ressources existantes. Le point de départ dans la constitution du corpus *ParCoJour* est le treebank littéraire *ParCoTrain-Synt* (Miletic, 2018). *ParCoTrain-Synt* est doté de trois couches d'annotation, il est librement disponible et l'état de l'art de parsing du serbe est atteint en l'utilisant en tant que corpus d'entraînement (LAS[1]=87,48, UAS[2]=91,22). Nous exploitons donc les modèles de traitement entraînés sur ce corpus pour effectuer un prétraitement de nos données, suivi d'une correction manuelle.

Notre deuxième objectif est d'effectuer une première évaluation des effets des domaines textuels différents sur les résultats du parsing du serbe. Malgré les performances élevées qu'il permet d'atteindre en parsing, *ParCoTrain-Synt* reste limité à un seul domaine textuel – la littérature. Cela remet en question son utilité lors du traitement de textes provenant d'autres domaines, comme des textes journalistiques ou scientifiques. En effet, des expériences en adaptation de domaine montrent que les modèles de parsing entraînés sur des corpus mono-genre sont particulièrement peu robustes lors du passage à un nouveau type de texte (Nivre et al. 2007a, Gildea 2001). Les expériences d'Agić et al. (2013) sur l'étiquetage et la lemmatisation du serbe et du croate montrent des pertes plus importantes lors du changement de domaine (journalistique *vs* encyclopédique) que lors du changement de langue (croate *vs* serbe). Le travail d'Agić & Ljubešić (2015) sur le parsing de ces deux langues aboutit aux mêmes observations.

Il faut cependant noter que dans toutes les expériences citées ci-dessus le corpus d'entraînement relevait du domaine journalistique. Le choix de ce domaine pour le nouveau corpus du serbe est, entre autres, motivé par le but de mener les expériences dans le sens inverse. Dans nos expériences, nous effectuons donc des évaluations dans les deux sens (entraînement sur le littéraire et évaluation sur le journalistique et *vice versa*). Notre intuition linguistique nous mène à considérer les textes littéraires comme syntaxiquement plus complexes que les textes journalistiques. Par conséquent, nous nous demandons si la baisse lors de l'adaptation de domaine serait plus importante dans le cas où le corpus d'entraînement serait du domaine littéraire et le corpus d'évaluation du domaine

1 Le score de LAS (score de rattachement labellisé) représente le pourcentage des tokens pour lesquels le parser a bien déterminé le gouverneur aussi bien que la fonction.

2 Le score de UAS (score de rattachement non labellisé) équivaut au pourcentage des gouverneurs bien identifiés. Pour calculer le score de UAS, l'identification de la fonction n'est pas prise en compte.

journalistique[3]. Autrement dit, l'hypothèse de notre recherche serait la suivante : un système d'apprentissage automatique entraîné sur un corpus littéraire atteint de meilleurs scores dans le parsing des textes journalistiques que l'inverse.

Dans la suite de l'article, nous présentons l'état de l'art du TAL pour le serbe dans la partie 2. La méthodologie de la constitution du nouveau corpus ainsi que le corpus créé dans le cadre de cette recherche sont décrits dans la section 3. Les expériences de l'adaptation de domaine que nous avons menées et leurs résultats sont détaillés dans la section 4.

2 Le serbe en traitement automatique des langues

Pour justifier notre choix du treebank de départ, nous présentons d'abord les ressources qui existent déjà pour le traitement automatique du serbe. Nous nous focalisons sur la disponibilité des ressources. Dans un deuxième temps, nous présentons l'état de l'art dans le parsing du serbe pour avoir un point de référence dans l'analyse des résultats de nos expériences.

2.1 Disponibilité des ressources

Ressource	Type de ressource	Annotation	Taille (en tokens)
corpus *NETK* et *SrpKor2003*	corpus de la langue serbe contemporaine	non annotés	22 millions
SrpKor2013	corpus du serbe contemporain de domaines diversifiés	bibliographique et morphologique	122 millions
SrpLemKor[4]	sous-corpus journalistique du corpus *SrpKor2013*	lemmatisation	3,7 millions
Corpus du serbo-croate de Henning Moerk de l'Université d'Aarhus	corpus des textes de prose publiés entre 1955 et 1990		
Crise éléctorale de l'année 2000	corpus journalistique		
	corpus des proverbes de Vuk Karadžić		

TABLE 1 : Ressources du TAL pour le serbe – indisponibles

Parmi les ressources auxquelles nous n'avons pas réussi à accéder figurent les bases textuelles citées sur le site[5] de la Faculté des mathématiques de l'Université de Belgrade. Elles sont énumérées dans le tableau 1 et accompagnées d'informations disponibles sur le site. Nous n'avons pas trouvé le lien pour télécharger les ressources, même si certaines sont censées être disponibles sous la licence

3 Il faut mentionner que des techniques d'adaptation des outils à un nouveau domaine autre que l'entraînement d'un outil sur un corpus des textes de ce domaine existent. Nous avons cependant choisi la technique décrite pour rendre les résultats de nos expériences comparables aux résultats des expériences antérieures et pour essayer de répondre à la question posée.

4 D'après les informations sur le site, *SrpLemKor* peut être téléchargé depuis l'adresse suivante : http://www.korpus.matf.bg.ac.rs/SrpLemKor/. Dernier accès : le 6 janvier 2019. Pourtant, nous n'avons pas réussi à le faire.

5 http://korpus.matf.bg.ac.rs/prezentacija/korpusi.html. Dernier accès : le 6 janvier 2019.

CC_BY-NC. D'après les informations sur le site, il est possible d'effectuer la recherche en ligne dans certains corpus, mais il faut avoir le compte utilisateur que nous n'avons pas réussi à créer.

Ressource	Type	Annotation	Taille
Lexiques			
lexique de MULTEXT-East (Krstev et al., 2004)	NA	NA	16 907 formes fléchies
wikimorph-sr (Miletic, 2017)	NA	NA	1 226 638 formes fléchies
Corpus			
1984 (Krstev et al., 2004) [6]	traduction du roman *1984*	morphosyntaxique	environ 100 000 mots
ParCoTrain (Miletic, 2013)	corpus de domaines divers	morphosyntaxique et lemmatisation	environ 150 000 tokens
ParCoTrain-Synt, (Miletic, 2018)	treebank littéraire	morphosyntaxique, syntaxique, lemmatisation	la première version de 81 000 tokens
SETimes.SR[7] (Samardžić et al., 2017)	treebank journalistique	morphosyntaxique, syntaxique, lemmatisation[8]	86 726 tokens

TABLE 2 : Corpus du TAL pour le serbe – disponibles

Le tableau 2 regroupe les ressources serbes du TAL que nous avons réussi à télécharger. Les premières ressources du TAL pour le serbe ont été développées à l'issu du projet MULTEXT-East (Dimitrova et al., 1998). Il s'agit du corpus *1984* et du lexique composé à partir des mots de ce corpus (Krstev et al., 2004) qui contient la traduction serbe du roman *1984* de George Orwell. Toutefois, dans le développement des ressources du TAL pour le serbe, le projet *ParCoLab*[9] (Miletic et al., 2017) représente un point crucial surtout concernant la disponibilité. *ParCoLab* est un corpus parallèle de textes en serbe, anglais et français. Chaque langue du corpus est représentée aussi bien à travers des traductions que des originaux. Les textes qui y figurent proviennent majoritairement du domaine littéraire. Néanmoins, les domaines se diversifient au fur et à mesure que le corpus s'enrichit. Il est désormais composé de 11 359 761 mots et il est accompagné de ressources pour le traitement automatique du serbe.

En plus des ressources développées directement pour le serbe, il est important de mentionner celles développées pour le croate et, dans la plupart des cas, adaptées au serbe par une équipe de chercheurs de l'Université de Zagreb (Agić et al., 2013 ; Ljubešić et al., 2016 ; Agić, Ljubešić, 2015). Ils énumèrent les corpus suivants : *Croatia Weekly*, corpus parallèle journalistique anglo-croate (100 000 tokens) ; *hrWaC* (1,9 milliard de tokens) et *srWaC* (894 millions de tokens), corpus collectés sur Internet ; *hr500k* (500 000 tokens), corpus provenant de domaines diversifiés ; *Setimes.HR* (90 000 tokens), corpus journalistique. Certains de ces corpus sont dotés d'annotation morphosyntaxique, de lemmatisation et d'annotation syntaxique. Quant aux lexiques

6 CC BY-NC-SA 4.0 (https://creativecommons.org/licenses/by-nc-sa/4.0/

7 Téléchargeable depuis l'adresse du projet Universal Dependencies : https://universaldependencies.org/. Dernier accès : le 18 février 2019.

8 Les annotations ont été effectuées conformément à la méthode du projet Universal Dependencies (McDonald et al., 2013).

9 http://parcolab.univ-tlse2.fr/en/about/resources/ Dernier accès : le 15 octobre 2018.

morphologiques, ils présentent *Hrvatski morfološki leksikon* (lexique morphologique croate) de 3,8 millions de formes fléchies et les lexiques *hrLex* et *srLex* construits à partir d'un lexique serbo-croato-bosniaque d'Apertium[10], un système de traduction automatique de langues proches. Toutes ces ressources développées par l'équipe de l'Université de Zagreb sont librement disponibles[11].

Ce survol bref des ressources pour le traitement automatique du serbe montre qu'elles ne sont pas nombreuses. La plupart de ces ressources sont citées dans la littérature et énumérées sur les sites, mais elles ne sont pas téléchargeable. Autrement dit, bien qu'elles soient créées, elles ne peuvent pas être effectivement utilisées. D'autres défauts des ressources serbes du TAL existent. Par exemple, la plupart des corpus serbes sont des bases textuelles non annotées. Par conséquent, ils ne peuvent pas servir de corpus d'apprentissage pour des systèmes d'apprentissage automatique. En revanche, des corpus serbes qui sont annotés ne contiennent pas de textes de domaines diversifiés et se limitent le plus souvent au domaine littéraire. En conclusion, il existe un grand besoin de développement de nouvelles ressources annotées, provenant d'autres domaines textuels, afin de favoriser le progrès du TAL pour le serbe.

2.2 État de l'art

	Étiquetage morphosyntaxique fin (%)		**Lemmatisation (%)**	
	Croate	Serbe	Croate	Serbe
Agić et al. (2013), corpus d'entraînement - *SETimes*, lemmatiseur - *CST*, étiqueteur - *HunPOS*	87,72	85,56	97,78	96,30
Ljubešić et al. (2016), corpus d'entraînement - *hr500k*, *CRF*, lexique - hrLex	92,53	92,33	NA	NA
Gesmundo & Samardžić (2012), corpus d'entraînement *1984*, *BTagger*	NA	86,65	NA	97,72
Miletic (2018), corpus d'entraînement *ParCoTrain-Synt*, lemmatiseur – *CST*, étiqueteur – *HunPOS*, lexique *ParCoLex*	NA	85,00	NA	96,50

TABLE 3 : État de l'art de l'étiquetage morphosyntaxique fin et de la lemmatisation du serbe

Le tableau 3 contient les taux d'exactitude les plus élevés atteints dans la lemmatisation et dans l'étiquetage du serbe et du croate que nous avons trouvés dans la littérature. Agić et al. (2013)[12] effectuent les expériences qui montrent que *BTagger* est beaucoup plus chronophage que *HunPOS* (Halácsy et al., 2007) en tant qu'étiqueteur et que *CST* (Jongejan, Dalianis, 2009) en tant que lemmatiseur même si l'étiqueteur et lemmatiseur *BTagger* donne les résultats les plus élevés (Gesmundo, Samardžić, 2012). L'état de l'art actuel en parsing du serbe et du croate est présenté dans le tableau 4. Étant donné que les schémas d'annotation dans le cadre de ces deux travaux diffèrent, les résultats ne sont pas directement comparables. Néanmoins, ce sont les résultats de Miletic (2018) qui dépassent largement l'état de l'art antérieur.

10 L'interface se trouve à l'adresse suivante https://www.apertium.org/index.eng.html?dir=cat-arg#translation. Dernier accès : le 22 septembre 2018.

11 Depuis les adresses suivantes : https://github.com/ffnlp/sethr et https://www.clarin.si/repository/xmlui/. Dernier accès : le 18 février 2019.

12 L'apprentissage sur un même corpus prend 6 heures avec *BTagger*, alors qu'avec *HunPOS* il ne prend qu'une seconde.

	Croate		Serbe	
	UAS	LAS	UAS	LAS
Agić & Ljubešić (2015), parser *Mate*, corpus *Setimes.hr*	86,9	81,5	86,0	81,5
Miletic (2018), parser *Talismane*, corpus *ParCoTrain-Synt*	NA	NA	91,22	87,48

TABLE 4 : État de l'art dans le parsing du serbe et du croate

Une des conclusions de Nivre et al. (2007a) est que les schémas d'annotation et les principes adoptés lors de l'adaptation de domaine doivent être pareils si l'on veut profiter de cette démarche. Nous avons ainsi décidé d'adopter la méthodologie de Miletic (2018) dans la création de notre ressource. Nous nous servons donc de *ParCoTrain-Synt* pour faciliter ce processus et pour rendre les résultats de nos expériences comparables à ceux de Miletic (2018).

3 Création d'un treebank littéraire serbe, *ParCoJour*

Pour réaliser notre premier objectif, nous avons procédé à la création d'un treebank journalistique, *ParCoJour*. Il est composé de 37 articles journalistiques de 30 566 tokens au total. Les articles ont été tirés de deux journaux nationaux, *Danas*[13] (quotidien) et *NIN*[14] (hebdomadaire). Pour essayer d'éviter tout biais, nous avons choisi les textes de manière aléatoire. La longueur des textes varie de 251 à 15 291 tokens. Les articles de *NIN* couvrent la période de 2003 à 2017 et ceux de *Danas* de 2007 à 2017. Les articles ont été écrits par 28 auteurs différents. Après les avoir collectés, nous avons procédé à l'annotation en suivant la méthode de Miletic (2018). Nous présentons d'abord cette méthode. Ensuite, nous présentons le travail pratique effectué.

3.1 Méthodologie de création de *ParCoJour*

La méthode globale préconisée par Miletic (2018) est la suivante : nous effectuons l'étiquetage morphosyntaxique détaillé (incluant des traits morphosyntaxiques fins, comme le cas, le nombre, le genre, etc.), la lemmatisation et le parsing. Ces niveaux de traitement sont effectués en cascade. Pour chaque niveau, nous nous servons des outils entraînés dans le cadre de (Miletic 2018) afin d'effectuer une préannotation automatique de nos données. Plus précisément, nous utilisons *HunPOS* (Halácsy et al., 2007), pour l'étiquetage morphosyntaxique, *CST* (Jongejan, Dalianis, 2009) pour la lemmatisation et *Talismane* (Urieli, 2013) pour le parsing. Nous adoptons également les mêmes jeux d'étiquettes et schémas d'annotation. La préannotation automatique est ensuite corrigée manuellement. Dans la correction manuelle des annotations automatiques, nous avons suivi les instructions des trois guides d'annotation qui figurent dans les annexes de Miletic (2018). Il s'agit de *Guide d'annotation morphosyntaxique, Guide de lemmatisation* et *Guide d'annotation syntaxique*. Lorsque le *Guide de lemmatisation* n'offrait pas la solution aux problèmes rencontrés, nous avons consulté le dictionnaire de référence de (Miletic 2018), à savoir *Srpski elektronski rečnik* de Milorad Simić. Cette approche de préannotation automatique qui est ensuite corrigée manuellement s'est montrée comme moins chronophage que l'annotation manuelle (Miletic et al., 2016 ; Fort, Sagot, 2010).

13 https://www.danas.rs/ Dernier accès : le 17 janvier 2019.
14 http://www.nin.co.rs/ Dernier accès : le 17 janvier 2019.

Le choix d'effectuer une analyse morphosyntaxique détaillée est motivé par les observations que les scores de parsing d'un même texte en serbe aussi bien qu'en croate sont meilleurs si ce texte est doté au préalable d'une annotation morphosyntaxique détaillée (voir Agić, Ljubešić, 2015 ; Miletic, 2018). Cela est dû au fait que l'ordre des constituants en serbe est flexible et que ce sont les indices morphosyntaxiques qui nous informent sur les fonctions syntaxiques.

En ce qui concerne le parsing, nous adoptons l'analyse syntaxique en dépendances. *Talismane* traite les arbres en dépendances et le treebank que nous utilisons dans le processus est composé d'arbres syntaxiques de ce type. Ce choix a été motivé par la nature du serbe, qui est une langue à ordre de mots flexible.

3.2 Travail pratique de création du corpus *ParCoJour*

Le corpus textuel avant annotation a été encodé au format XML. Chaque élément <text> contient un article du corpus. Différentes métadonnées sont stockées au niveau de cet élément en tant qu'attributs : l'id (le numéro de l'article dans le corpus), la source (soit « nin » soit « danas »), la date de publication de l'article, l'auteur et l'adresse URL à partir de laquelle il a été récupéré. Afin d'être exploitable dans les tâches d'étiquetage morphosyntaxique, de lemmatisation et d'annotation syntaxique, le corpus a dû être transformé en d'autres formats, accessibles aux outils automatiques sélectionnés. Les formats exploités par des outils que nous avons utilisés sont présenté dans le tableau 5.

Outil	*HunPOS* (Halácsy et al., 2007)	*CST* (Jongejan, Dalianis, 2009)	*Talismane* (Urieli, 2013)
Format d'entrée	document XML segmenté et tokénisé	[token]\t[tag]	CoNLL-X
Format de sortie	[token]\t[tag]	[token]\t[tag]\t[lemme]	CoNLL-X
Format de correction manuelle	CSV	CSV	*standoff* de *Brat* (Stenetorp et al., 2012)
Corpus d'entraînement	*ParCoTrain-Synt*[15] (Miletic, 2018)	Le modèle[16] créé par Miletic (2018)	Le modèle le plus performant de Miletic (2018)[17]

TABLE 5 : Formats exploités par outils choisis

Il n'est pas facile de corriger manuellement les annotations dans les formats exploités par les outils. Tableau 5 contient également les informations sur les formats de corpus sous lesquels la correction manuelle a été effectuée. Pour faciliter la correction de l'annotation morphosyntaxique et de la lemmatisation pour un annotateur humain, nous avons converti la sortie de *HunPOS* et de *CST* en format CSV. Figure 2 illustre une phrase lemmatisée sous format CSV. Le token se trouve dans la première colonne, l'étiquette morphosyntaxique dans la deuxième et le lemme dans la troisième.

Seule l'auteure a participé dans la correction manuelle des préannotations automatiques. La correction manuelle de l'annotation syntaxique automatique a cependant été vérifiée par l'auteure de *ParCoTrain-Synt.* La vitesse moyenne de correction de l'annotation morphosyntaxique était de 430

15 Il est disponible sous la licence Creative Commons BY-NC-SA 3.0 et il peut être téléchargé à partir de l'adresse suivante : http://parcolab.univ-tlse2.fr/en/about/resources/. Dernier accès : le 15 octobre 2018.

16 Librement disponible à l'adresse : https://github.com/aleksandra-miletic/serbian-nlp-resources/tree/master/Models/Lemmatisation. Dernier accès : le 15 octobre 2015.

17 Ce modèle peut être téléchargé à partir de l'adresse suivante : https://github.com/aleksandra-miletic/serbian-nlp-resources/tree/master/Models/Parsing/*Talismane*. Dernier accès : le 15 octobre 2018.

tokens par heure, ce qui est moins que la vitesse de l'annotateur expérimenté dans le cadre de Miletic (2018) (800 tokens par heure), mais plus que la vitesse de l'annotateur novice (325 tokens par heure). La comparaison de notre vitesse avec celle de l'annotation manuelle sans pré-annotation automatique d'un annotateur novice dans le cadre de Miletic (2018) (80 tokens) justifie notre décision d'effectuer la correction d'une annotation automatique au lieu d'annoter tout le corpus manuellement. Notre vitesse moyenne de la correction manuelle de lemmatisation était de 1500 tokens par heure, ce qui est de nouveau plus vite que l'annotation manuelle sans pré-annotation automatique dans le cadre de Miletic (2018) (825 tokens par heure).

52	Lideri	N_com_nom_pl_m	Lider
53	EU	N_prop_gen_sg_f	eu
54	pristali	V_main_partact_-_pl_m_-	pristati
55	na	Prep	na
56	kompromis	N_com_acc_sg_m	kompromis
57	,	Z	,
58	ali	C_coord	ali
59	ne	Part	ne
60	odustaju	V_main_pres_3_pl_-_-	odustajati
61	od	Prep	od
62	sopstvenih	A_qual_gen_pl_m_pos	sopstven
63	vojnih	A_qual_gen_pl_m_pos	vojni
64	planova	N_com_gen_pl_m	plan
65			

FIGURE 1: Extrait de *ParCoJour* étiqueté et lemmatisé en format CSV

Pour corriger manuellement le parsing, nous nous sommes servi de l'outil *Brat*[18] de Stenetorp et al. (2012). L'interface graphique de cet outil (voir Figure 2) facilite la correction de l'annotation syntaxique étant donné que la représentation graphique dans cet outil permet de visualiser l'arbre. Les phrases sont présentées linéairement avec l'étiquette morphosyntaxique détaillée en dessous de chaque token.

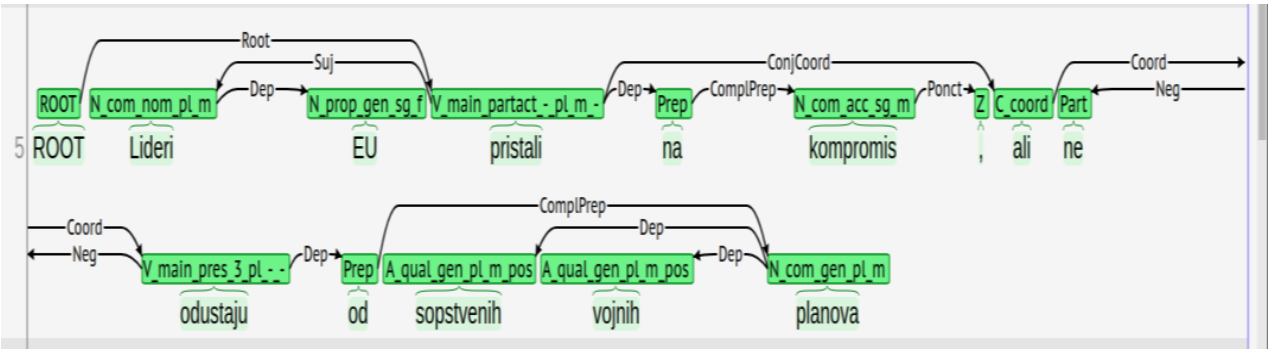


FIGURE 2: Interface graphique de *Brat*

18 Il peut être téléchargé à partir de l'adresse suivante : http://brat.nlplab.org/. Dernier accès : le 15 octobre 2018.

Ces trois processus effectués, nous avons atteint notre premier but : nous avons obtenu un corpus de textes journalistiques, *ParCoJour*, composé de 30 566 tokens et doté d'annotations syntaxique, morphosyntaxique et de lemmatisation. La suite de l'article porte sur notre objectif méthodologique. La méthode adoptée dans les expériences, l'application de cette méthode et les réponses à nos questions qui en découlent sont détaillées dans la section suivante.

4 Adaptation au domaine

Les corpus à notre disposition (*ParCoJour* et *ParCoTrain-Synt*) relevant des domaines restreints (journalistique et littéraire respectivement), nous pouvons les utiliser dans des expériences dont le but est d'évaluer l'impact des domaines textuels des corpus d'entraînement et d'évaluation sur le parsing. Pour ce faire, nous avons mené trois expériences. Nous les présentons dans cette section aussi bien que les résultats de ces expériences. Ils sont commentés et confrontés aux scores des deux études antérieures du parsing du serbe que nous avons présentées (Miletic, 2018 ; Agić, Ljubešić, 2015).

4.1 Expériences d'adaptation au domaine avec *ParCoJour*

Pour effectuer des évaluations sur notre corpus, nous l'avons divisé en trois parties : *dev* (1 596 tokens), *train* (30 063 tokens), *test* (3 036 tokens). La section *test* est parsée automatiquement et comparée à l'annotation *gold* lors de l'évaluation. Lorsque nous parlons des corpus *ParCoJour* et *ParCoTrain-Synt* dans l'analyse des résultats des expériences, nous nous référons aux sections test des corpus respectifs.

Talismane est doté d'un sytème d'évaluation et nous l'exploitons dans nos expériences. Lors de l'évaluation, *Talismane* crée deux fichiers, un fichier TXT et un fichier CSV. Le fichier TXT contient l'annotation que *Talismane* a produite sous le format conforme aux propositions des campagnes CoNLL-X. Le fichier CSV contient les résultats quantitatifs de la comparaison sous forme d'une matrice de confusion accompagnée de scores standard de parsing. Il s'agit du nombre total d'occurrences d'une étiquette dans le corpus *test*, ainsi que la précision, le rappel et la f-mesure liés à cette étiquette. Ce fichier contient également le score LAS, alors que le score UAS doit être calculé ultérieurement.

Pour comparer les performances de *Talismane* lorsqu'il est entraîné sur un corpus littéraire à ses performances lorsque le corpus d'entraînement est un corpus journalistique nous avons effectué trois expériences : 1) parsing de *ParCoJour* avec le modèle entraîné sur *ParCoJour* ; 2) parsing de *ParCoJour* avec le modèle entraîné sur *ParCoTrain-Synt* ; 3) parsing de *ParCoTrain-Synt* avec le modèle entraîné sur *ParCoJour*. Dans chaque expérience, l'évaluation a été effectuée sur la section *test* des corpus correspondant. Les résultats obtenus dans les expériences menées sont ensuite comparés aux résultats des expériences de Miletic (2018) où le modèle entraîné sur *ParCoTrain-Synt* a été évalué sur la section *test* du même corpus.

La configuration optimale proposée par Miletic (2018) comprend le paramètre *cutoff* de 3 et le paramètre *beam width* de 5. L'augmentation de la valeur de *beam width* entraîne l'augmentation du temps nécessaire d'effectuer l'apprentissage. Ici, nous nous éloignons de la configuration optimale : le temps étant un facteur important dans notre étude, nous effectuons nos expériences en utilisant le *beam width* de 1. En conséquence, nous comparons nos résultats avec les résultats obtenus par Miletic (2018) avec la même valeur de ce paramètre (voir Table 7). La comparaison des résultats dans les conditions optimales reste une piste à suivre dans les travaux futurs.

Miletic (2018) évalue aussi l'effet du type d'annotation morphosyntaxique qui précède le parsing. Les résultats montrent qu'une annotation manuelle donne de meilleurs résultats qu'une annotation automatique non corrigée. Le but de notre recherche étant l'évaluation de l'effet du genre sur le parsing, nous n'analysons pas les effets du type d'annotation qui précède, mais nous utilisons les conditions d'apprentissage concernant l'étiquetage qui se sont montrées comme les meilleures dans le cadre de Miletic (2018). L'analyse des effets de type de l'étiquetage morphosyntaxique sur le parsing du corpus journalistique *ParCoJour* peut être envisagée dans des travaux ultérieurs.

4.2 Résultats des expériences d'adaptation au domaine

Les tableaux 6 et 7 contiennent les résultats quantitatifs globaux des expériences menées dans le cadre de notre travail. Dans le tableau 6 figurent les résultats des deux premières expériences où le corpus d'évaluation était la section test du corpus *ParCoJour* et le corpus d'entraînement changeait. Le tableau 7 donne les résultats du parsing avec le corpus *ParCoTrain-Synt* comme le corpus d'évaluation. Pour le cas où le modèle entraîné sur *ParCoTrain-Synt* est évalué sur le corpus *test* de *ParCoTrain-Synt* nous reprenons les résultats de Miletic (2018) avec le paramètre *beam width* de 1 et où l'étiquetage moprhosyntaxique était manuel (voir Table 7).

		Corpus d'évaluation *ParCoJour*	
	Corpus d'entraînement	UAS	LAS
Expérience 1	*ParCoJour*	89,72	85,08
Expérience 2	*ParCoTrain-Synt*[19]	87,15	80,86
	différentiel de performance dans l'expérience avec changement de domaine de corpus d'entraînement	-2,57	-4,22

TABLE 6 : Résultats du parsing dans les expériences avec le corpus *ParCoJour* comme corpus d'évaluation

		Corpus d'évaluation *ParCoTrain-Synt*	
	Corpus d'entraînement	UAS	LAS
Expérience de Miletic (2018)	*ParCoTrain-Synt*	91,19	87,83
Expérience 3	*ParCoJour*	85,75	81,24
	différentiel de performances dans l'expérience avec changement de domaine de corpus d'entraînement	-5,44	-6,59

TABLE 7 : Résultats du parsing dans les expériences avec le corpus *ParCoTrain-Synt* comme corpus d'évaluation

Dans les cas où le domaine des corpus d'entraînement et d'évaluation ne change pas (expérience 1 et expérience de Miletic (2018)), les meilleurs scores sont atteints dans le parsing du corpus littéraire

19 Résultats du parsing de *ParCoTrain-Synt* si beam width = 1 (Miletic, 2018, p. 215).

(voir Table 6 et 7). Étant donné que les conditions d'apprentissage, le schéma d'annotation et le jeu d'étiquettes ne diffèrent pas, nous supposons que les scores moins bons dans le parsing des textes journalistiques d'après le modèle entraîné sur le corpus journalistique viennent de la différence des tailles des deux corpus analysés. En effet, le corpus *ParCoJour* contient environ 30 000 tokens, alors que la version de *ParCoTrain-Synt* que nous utilisons ici en contient environ 81 000. Pour tester cette hypothèse, nous proposons d'enrichir *ParCoJour* de nouveaux textes journalistiques ou de limiter la taille de *ParCoTrain-Synt* à 30 000 tokens dans les travaux à suivre.

Pour tester les résultats du changement de domaine, nous confrontons ensuite les résultats de la deuxième et de la troisième expérience. Les deux expériences confirment une chute des résultats lors de ce processus. Les dernières lignes des tableaux 6 et 7 contiennent le différentiel des performances dans ces deux expériences. En analysant ces baisses, nous essayons de déterminer s'il est plus utile d'avoir à disposition un corpus d'entraînement littéraire ou un corpus journalistique si l'on doit traiter les deux genres textuels. Les résultats ont baissé de 2,57 en UAS et de 4,22 en LAS si le corpus d'évaluation était *ParCoJour* et de 5,44 en UAS et de 6,59 en LAS si le corpus d'évaluation était *ParCoTrain-Synt* (voir Table 6 et 7), ce qui est moins que ce qui a été observé auparavant dans le cadre de (Nivre et al., 2007a). Néanmoins, les résultats ne sont pas directement comparables étant donné que les schémas d'annotation et les tailles de corpus diffèrent.

Dans le cadre de notre recherche, la baisse des scores est plus importante si le corpus à parser est le corpus littéraire. Cette observation correspond aux résultats de Gildea (2001). Un parser est, donc, plus performant sur les textes journalistiques que sur les textes littéraires. Nous pouvons conclure que le corpus d'entraînement littéraire est plus utile dans le parsing des textes journalistique que le corpus journalistique ne l'est dans le parsing des textes littéraires. Il faut, toutefois, prendre en considération les tailles différentes des corpus d'entraînement dans nos expériences. D'autre part, le score de UAS est meilleur de 1,4 dans l'expérience où le corpus d'entraînement était le corpus *ParCoTrain-Synt* et le corpus d'évaluation *ParCoJour*. Le score de LAS est, en revanche, meilleur de 0,38 dans la troisième expérience où le corpus d'entraînement était *ParCoJour* et le corpus d'évaluation *ParCoTrain-Synt*. Le fait que les résultats ne sont pas toujours meilleurs avec un corpus littéraire comme corpus d'entraînement, et cela dans le cas où ce corpus est plus petit, confirment l'importance d'avoir les corpus de genres diversifiés.

Il serait intéressant d'entrer dans une analyse plus profonde des différences linguistiques entre ces deux types de corpus. Cette analyse pourrait expliquer la cause des différences dans les performances d'un parser sur ces deux types de corpus. Faute de place, nous sommes obligée de laisser cette analyse pour un prochain travail.

Dans l'introduction, nous avons présenté les scores du parsing du serbe et du croate dans le cadre du travail d'Agić et Ljubešić (2015). Avant Miletic (2018), c'était Agić & Ljubešić (2015) qui avaient atteint l'état de l'art dans le parsing du serbe. Le score de LAS était de 81,5 et le score de UAS de 86,0. Les scores dans le cadre de notre étude sont inférieurs à ceux de Miletic (2018), mais dépassent les scores d'Agić et Ljubešić (2015). Ceci est vrai même dans les scénarios avec changement de domaine. Il faut, cependant, noter que le schéma d'annotation d'Agić et Ljubešić (2015) diffère du schéma dans le cadre de notre recherche. Par conséquent, les résultats ne sont pas directement comparables. Néanmoins, les scores atteints dans le cadre de notre étude justifient le parti pris dans notre travail de suivre la méthodologie adoptée par Miletic (2018) et illustrent également l'utilité de la diversification des genres dans la constitution des corpus annotés.

5 Conclusion et pistes

Deux grands objectifs de notre travail étaient de créer un treebank journalistique pour le serbe et de faire des expériences en adaptation du domaine sur cette langue. Par le premier objectif, nous avons envisagé de contribuer à la diversification de *ParCoTrain-Synt* (Miletic, 2018), un treebank littéraire pour le serbe. L'objectif méthodologique était censé contribuer à la compréhension du mécanisme d'adaptation de domaine. Le corpus *ParCoJour* sera librement disponible sur le site du projet *ParCoLab*.

Lors de la création de notre treebank, nous nous sommes servi de *ParCoTrain-Synt*, un treebank existant. Le corpus *ParCoJour* que nous avons créé est composé de 37 articles journalistiques de 30 566 tokens au total. *ParCoJour* est doté de l'annotation morphosyntaxique, de la lemmatisation et de l'annotation syntaxique en dépendances suivant les principes et les schémas d'annotation présentés dans (Miletic, 2018).

Nous utilisons le corpus *ParCoTrain-Synt* pour mener des expériences censées donner des résultats qui répondraient à quel point la baisse des performances est due au domain concret des corpus d'entraînement et si la baisse serait importante au cas où le corpus d'entraînement relevait du domaine littéraire. Les limites de ce travail ne nous permettant pas d'analyser le traitement à tous les niveaux possibles, nous avons décidé de mener les expériences concernant le parsing.

Dans le parsing du corpus journalistique, *Talismane*, l'outil que nous avons utilisé, atteint le score de 85,08 en LAS et de 89,72 en UAS si entraîné sur le corpus du même domaine. Ces scores sont inférieurs à ceux de Miletic (2018), mais dépassent les scores d'Agić et Ljubešić (2015). Ceci est vrai même pour les scores dans les scénarios avec changement de domaine. Même si le schéma d'annotation d'Agić et Ljubešić (2015) diffère du schéma dans le cadre de notre recherche et même si les résultats ne sont pas directement comparables, les scores atteints dans le cadre de notre étude justifient le parti pris dans notre travail de suivre la méthodologie adoptée par Miletic (2018). Ils confirment aussi que l'utilisation d'un corpus littéraire, quoique du domaine restreint, donne des résultats satisfaisants dans le parsing des textes journalistiques.

D'autre part, le score de UAS est meilleur de 1,4 dans l'expérience où le corpus d'entraînement était le corpus *ParCoTrain-Synt* et le corpus d'évaluation *ParCoJour*. Le score de LAS est, en revanche, meilleur de 0,38 dans l'expérience où le corpus d'entraînement était *ParCoJour* et le corpus d'évaluation *ParCoTrain-Synt*. Le fait que les résultats ne sont pas toujours meilleurs avec un corpus littéraire comme corpus d'entraînement confirment l'importance d'avoir des corpus de genres diversifiés et justifie le choix de créer un corpus journalistique pour le serbe. Ainsi, nous avons contribué à l'enrichissement des ressources du TAL pour la langue serbe.

Le travail fait dans le cadre de cette recherche peut être poursuivi dans le but d'améliorer les résultats obtenus et d'approfondir l'analyse de l'adaptation de domaine. D'abord, la piste la plus importante serait d'augmenter la taille du corpus *ParCoJour* pour rendre les résultats des expériences plus immédiatement comparables à ceux des expériences menées sur *ParCoTrain-Synt*. L'autre possibilité est d'entraîner l'outils sur seulement une partie du corpus *ParCoTrain-Synt* dont la taille serait comparable à celle de *ParCoJour*. Ensuite, il serait bien d'évaluer l'exactitude de la lemmatisation et de l'étiquetage morphosyntaxique lors du changement de domaine. Un autre scénario d'évaluation possible comprendrait un entraînement de l'outil effectué dans des conditions définies comme optimales dans le cadre de Miletic (2018), à savoir, *cutoff* de 3, *beam width* de 5. Une autre piste possible serait d'évaluer l'effet de l'étiquetage automatique au lieu de l'étiquetage *gold* sur le parsing du corpus journalistique. Enfin, il serait également utile d'évaluer les performances de l'outil si entraîné sur un corpus mixte où une moitié serait composée des textes d'un genre, et l'autre moitié des textes de l'autre. Ceci nous permettrait d'évaluer pleinement l'intérêt de la diversification de treebank et l'apport du contenu journalistique au parsing du serbe.

Références

Agić Ž., Ljubešić N., Merkler D. (2013). Lemmatization and morphosyntactic tagging of Croatian and Serbian. Actes de *The 4th Biennial International Workshop on Balto-Slavic Natural Language Processing (BSNLP 2013)*, 48-57.

Agić Ž., Ljubešić N. (2015). Universal Dependencies for Croatian (that work for Serbian, too). Actes de *5th Workshop on Balto-Slavic Natural Language Processing (BSNLP 2015)*, 1-8.

Ljubešić N., Klubička F., Agić Ž, Jazbec I.-P. (2016). New inflectional lexicons and training corpora for improved morphosyntactic annotation of Croatian and Serbian. Actes de *The Tenth International Conference on Language Resources and Evaluation (LREC 2016)*.

Dimitrova L., Erjavec T., Ide N., Kaalep H. J., Petkevič V., Tufiş D. (1998). Multext-East: Parallel and Comparable Corpora and Lexicons for Six Central and Eastern European Languages. Actes de *COLING-ACL '98.*

Fagard B., Stosic D., Cerruti M. (2017). Within-type variation in Satellite-framed languages: The case of Serbian. *STUF, Akademie Verlag*, 70 (4), 637-660.

Fort K., Sagot B. (2010). Influence of Pre-annotation on POS-tagged Corpus Development. *The Fourth ACL Linguistic Annotation Workshop*, 56-63.

Gesmundo A., Samardžić T. (2012). Lemmatising Serbian as category tagging with bidirectional sequence classification. Acte de *The 8th Language Resources and Evaluation Conference (LREC 2012)*, 2103-2106.

Gildea D. (2001). Corpus Variation and Parser Performance. Actes de *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Halácsy P., Kornai A., Oravecz C. (2007). *HunPOS* : an open source trigram tagger. Actes de *The 45th annual meeting of the ACL on interactive poster and demonstration sessions*, 209-212.

Jongejan B., Dalianis H. (2009). Automatic training of lemmatization rules that handle morphological changes in pre-, in- and suffixes alike. Actes de *The Joint Conference of the 47th Annual Meeting of ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP.*,145-153.

Krstev C., Vitas D., Erjavec T. (2004). MULTEXT-East resources for Serbian. Actes B de *The 7th international multiconference information society: Language technologies,*108-114.

Kübler S., McDonald R., Nivre J. (2009). *Dependency parsing*. Morgan & Claypool Publishers.

McDonald R., Nivre J., Quirmbach-Brundage Y., Goldberg Y., Das D., Ganchev K., Hall K., Petrov S., Zhang H., Täckström O., Bedini C., Bartomeu Castelló N., Lee J. (2013). Universal dependency annotation for multilingual parsing. Actes de *The 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 92-97.

Miletic A. (2017). Building a morphosyntactic lexicon for Serbian using Wiktionary. Actes de *Sixièmes Journées d'études Toulousaines (JéTou2017)*, 30-34.

Miletic A., Stosic D., Marjanović S. (2017). ParCoLab : A Parallel Corpus for Serbian, French and English. *Text, Speech and Dialogue 2017, LNAI 10415*, 156-164.

Miletic A., Fabre C., Stosic D. (2016). Mise au point d'une méthode d'annotation morphosyntaxique fine du serbe. Actes de *Traitement Automatique des Langues Naturelles(TALN 2016)*, 506-514.

Miletic A. (2013). *Annotation morphosyntaxique semi-automatique d'un corpus littéraire serbe*. Master's thesis. Université Charles de Gaulle - Lille 3.

Miletic A. (2018). *Un treebank pour le serbe : constitution et exploitation.* PhD thesis. Université Toulouse-Jean Jaurès.

Nivre J., Hall J., Kübler S., McDonald R., Nilsson J., Riedel S., Yuret D. (2007). The CoNLL 2007 shared task on dependency parsing. Actes de *CoNLL shared task session of EMNLP-CoNLL*, 915-932.

Samardžić T., Agić Ž., Starović M., Ljubešić N. (2017). Universal Dependencies for Serbian in Comparison with Croatian and Other Slavic Languages. Actes de *The 6th Workshop on Balto-Slavic Natural Language Processing* 39-44.

Simić M. (2005). *Srpski elektronski rečnik*.

Stenetorp P., Pyysalo S., Topić G., Ohta T., Ananiadou S., Tsujii J. (2012). BRAT : a web-based tool for NLP-assisted text annotation. Actes de *The Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics,* 102-107.

Urieli A. (2013). *Robust French syntax analysis: reconciling statistical methods and linguistic knowledge in the Talismane toolkit*. PhD thesis, Université Toulouse le Mirail - Toulouse II.

Vers la traduction automatique d'adverbiaux temporels du français en langue des signes française

Sandra Bellato[1]
(1) MoDyCo, 200 Avenue de la République, 92001 Nanterre, France
bellato.sandra@parisnanterre.fr

RÉSUMÉ

Nous présentons ici de premiers travaux abordant la question de règles de passage entre deux formalismes décrivant la sémantique d'adverbiaux temporels respectivement pour le français et pour la Langue des Signes Française (LSF). Ces travaux prennent place dans une visée de traduction automatique d'une langue vers l'autre. Nous nous appuyons sur un corpus rassemblant 95 adverbiaux temporels du français traduits par trois locuteurs de la LSF.

ABSTRACT

Towards the automatic translation of temporal adverbials from French to French sign language.
The present paper is an initial study addressing the question of rules of passage between two formalisms describing the semantics of temporal adverbials respectively for French and for French Sign Language (FSL). This research takes place in an objective of automatic translation from one language to the other. We rely on a corpus of 95 French temporal adverbials translated by three FSL speakers.

MOTS-CLÉS : adverbiaux temporels, français, LSF, traduction

KEYWORDS : temporal adverbials, French, French Sign Language, translation

1 Introduction

Dans cet article, nous abordons la problématique de la traduction automatique des adverbiaux temporels du français vers la langue des signes française (dorénavant LSF). Dans le domaine de la traduction automatique, deux types de méthodes sont principalement utilisées : la méthode fondée sur des règles dites expertes qui modélisent un savoir linguistique permettant le traitement de données d'une source vers des représentations plus abstraites puis vers des données d'une langue cible ; la méthode par apprentissage automatique fondée sur des récurrences observées dans les données source et dans les données cible. Le manque de données accessibles pour la LSF, et en particulier de données bilingues de bonne qualité français-LSF qui sont essentielles pour un apprentissage statistique, nous a conduits à choisir pour cette étude une approche de traduction automatique fondée sur des règles expertes. Dans cet article, nous revenons dans un premier temps sur ce à quoi renvoie la notion d'adverbial temporel en français ainsi que sur les modes d'expression de la temporalité en LSF, et présentons quelques avancées actuelles en matière de traduction. Dans un second temps, après avoir expliqué notre méthodologie et présenté les différents formalismes que nous avons utilisés lors de

cette étude, nous présentons le corpus choisi pour cette recherche. Pour finir, nous détaillons les résultats obtenus grâce aux différentes étapes de formalisation et proposons en particulier trois règles de généralisation permettant la traduction automatique d'adverbiaux temporels du français vers la LSF.

2 Problématique

La temporalité est un champ d'analyse linguistique complexe, qui nécessite de s'intéresser à de nombreux marqueurs entrant en interaction (verbes, morphèmes de temps verbaux, connecteurs temporels, semi-auxiliaires, adverbes temporels). Dans le domaine du TAL, les adverbes temporels ont reçu une attention toute particulière au cours des quinze dernières années en réponse à des attentes pour les systèmes d'extraction d'information (Pustejovsky et al., 2003).
Pour le français, les travaux de Battistelli et al. (2008) ou Ehrmann et Hagège (2009) ont montré l'intérêt d'une approche fine de cette catégorie de marqueurs temporels qui s'avère plus complexe qu'elle n'en a l'air au premier abord, notamment parce qu'elle renvoie à une définition qui ne fait pas consensus en linguistique. La catégorisation de ces adverbiaux, souvent par une fonction et non par une nature grammaticale bien définie (Wilmet, 2003), ainsi que leur mobilité dans la phrase pouvant impliquer des différences sémantiques (Le Goffic, 1994), sont des exemples de cette complexité. Suivant les principes décrits dans Teissèdre (2012) à la suite de Battistelli et al. (2008), nous retiendrons le terme d'adverbiaux de localisation temporelle (dorénavant ALT) pour désigner les éléments de cette catégorie, laquelle sera entendue comme regroupant aussi bien des adverbes de temps (*depuis*), des syntagmes prépositionnels *(en été 2018*), des subordonnées circonstancielles de temps (*depuis qu'elle est arrivée*), que des syntagmes nominaux se comportant comme des compléments adverbiaux (*la veille*).
Si les travaux sur les adverbiaux en français sont nombreux, il n'en existe pas à notre connaissance en LSF, ou alors ils sont intégrés à une analyse plus générale de l'expression du temps en LSF.
Différents travaux (Maeder, 1994 ; Ivani Fusellier-Souza et Jimmy Leix, 2005 ; Sinte, 2015) ont montré que les signeurs, s'exprimant dans différentes langues des signes, possèdent plusieurs moyens à leur disposition pour rendre compte de notions temporelles. Ces moyens renvoient à :

– De nombreux signes lexicaux : ils peuvent être considérés comme équivalents à des noms communs (par exemple des signes désignant les jours de la semaine, les mois de l'année, …), à des adverbes ou prépositions (par exemple les signes AVANT, APRÈS, DEPUIS, …), ou encore comme pouvant traduire des notions aspectuelles (par exemple les signes FINI, ENCORE, …).

Figure 1 : Captures d'image d'une traduction observée de *lundi* dans une vidéo du corpus

– Des lignes de temps : elles permettent de donner une signification particulière à des signes lexicaux ayant à la base une forme dite neutre, mais qui sont flexibles dans l'espace (Sinte, 2015). La LSF utilise quatre lignes de temps, les lignes A et B sont porteuses de notions de temps (comme

le passé ou le futur) et sont situées sur le côté du corps, sur un axe avant-arrière, tandis que la ligne C est porteuse de notions d'aspect (comme la régularité, la continuité ou la durée) et est située devant le corps sur un axe droite-gauche. La ligne D quant à elle est utilisée pour indiquer une idée de croissance, qu'elle soit physique ou au niveau temporel, et n'est pas toujours attestée comme une ligne de temps (Maeder, 1994).

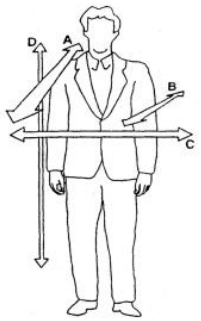


Figure 2 : Schéma des lignes de temps en LSF (Maeder, 1994)

– Des composants non manuels (clignements des yeux, expressions faciales, positions du visage et du corps, …) : ils servent à affiner le sens, de nuancer certains signes temporels (Filhol et al., 2014). Une ouverture des yeux peut par exemple signifier une accentuation du signe effectué.
– Un ordre dans lequel les événements et les signes sont signés. Les événements, par exemple, sont généralement signés dans l'ordre chronologique dans lequel ils se sont déroulés (Maeder, 1994).

Soulignons que tous ces moyens ne sont pas exclusifs les uns des autres mais apparaissent de manière combinée.
Il est à noter que les langues des signes, dont la LSF, sont des langues encore très peu outillées. Dans le domaine de la traduction automatique quelques efforts ont cependant été initiés (Morrissey, 2008 ; Kervajan, 2011), et les nouvelles avancées technologiques, notamment concernant l'animation d'avatars (Wolfe et al., 2016), permettent également d'envisager de grandes avancées dans ce domaine.

3 Méthodologie

Afin d'établir des règles de traduction automatique, il nous a fallu tout d'abord analyser des ALT en français ainsi que leur traduction en LSF. Pour ce faire, nous nous sommes appuyés sur deux formalismes : l'un dédié aux ALT en français (Teissèdre, 2012), et l'autre à la formalisation de la LSF (Filhol & Hadjadj, 2016). Nous avons appliqué ces formalismes sur un corpus français-LSF en nous concentrant sur les parties correspondant aux ALT en français. Nous avons ainsi pu observer certaines régularités dans les traductions en LSF selon différents critères sémantiques et syntaxiques. À partir de ces observations, nous avons ensuite proposé des règles générales de traduction qui permettraient le passage d'un formalisme vers l'autre.

3.1 Les formalismes d'analyse sémantique des ALT

3.1.1 Formalisation des ALT en français

C'est l'approche de Teissèdre (2012), déployée à partir de Battistelli et al. (2008), que nous avons décidé d'adopter pour décrire la sémantique des ALT en français. Cette approche est particulièrement intéressante pour notre étude car elle permet de décrire et de formaliser tous les ALT quelles que soient leur nature grammaticale et leur type d'ancrage – aussi appelé 'base' - sur un référentiel temporel. Cette base peut ainsi être de nature calendaire (e.g. *le 1er février 2003*), déictique (e.g. *la semaine prochaine*), anaphorique (e.g. *la veille*) ou relative à un événement (e.g. *dès l'élection présidentielle*). Ce formalisme distingue trois types d'opérateurs :

– L'opérateur de régionalisation : il permet de spécifier la région pointée par un ALT sur un axe temporel par rapport à une base. Elle est souvent dénotée en français à l'aide de prépositions ou de locutions prépositionnelles (vers, depuis, avant, bien après, …). L'opération de régionalisation peut prendre comme valeur ID (e.g. *en 2003*), DEPUIS (e.g. *depuis peu*), JUSQUE (e.g. *jusqu'à aujourd'hui*), AVANT (e.g. *bien avant*), APRÈS (e.g. *postérieurement à 2003*) ou VERS (e.g. *aux alentours du 1er février 2003*).
– L'opérateur de focalisation permet, quant à lui, d'effectuer un zoom par rapport à un repère temporel noyau. La focalisation peut être de différentes natures. Elle peut tout d'abord être granulaire, c'est-à-dire que l'on se focalise sur la granularité décrite par le repère temporel noyau (e.g. *ce jour-là*). La focalisation peut aussi se faire par subdivision, ce qui permet de zoomer sur une portion spécifique de la zone temporelle désignée (par exemple le début de la semaine dans *en début de semaine*). Cette focalisation par subdivision peut être numérique (quart, tiers, moitié) ou qualitative (début, milieu, fin). Il peut enfin s'agir d'une focalisation par sélection lorsqu'il s'agit de sélectionner une partie de la zone temporelle désignée (e.g. *dès le premier jour de l'hiver*). Cette focalisation par sélection peut être formée d'un rang et d'une unité calendaire.
– L'opérateur de déplacement : il permet d'effectuer un déplacement par rapport à une zone temporelle. Cet opérateur peut être formé d'un rang (e.g. *pendant le dernier mois*), d'une durée (e.g. *deux mois avant la fin*) ou d'une unité calendaire ordinale (e.g. *jusqu'au vendredi qui suivra son arrivée*). L'opérateur de déplacement a également une valeur liée à l'orientation du déplacement à effectuer, qu'elle soit égale à avant, après ou à ID si l'opération de déplacement est nulle.

Ces opérateurs s'appliquent dans l'ordre suivant :

OpRegionalisation (OpFocalisation / OpDéplacement (Base))

(1) *Quelque 50.000 personnes se sont rassemblées jeudi soir à Podgorica, la capitale du Monténégro, pour exprimer leur volonté que cette république devienne indépendante, trois jours avant la tenue d'un référendum historique sur cette question.* (phrase 1G)

Pour exemple, le groupe nominal « *jeudi soir* » présent dans la phrase (1) extraite de notre corpus serait ainsi formalisé :
Base : Déictique (dépend de la situation d'énonciation)
Régionalisation : ID
Focalisation : Soir (focalisation par subdivision)
Déplacement : Orientation : avant (non explicite dans l'ALT mais déduit grâce à la conjugaison du verbe *rassembler*) ; Unité calendaire ordinale : jeudi
Formalisation : ID (Soir / Orientation : avant ; Unité Calendaire Ordinale : jeudi (Déictique))

3.1.2 Formalisation en LSF

Afin de formaliser les traductions en LSF du corpus, nous nous sommes appuyés sur les travaux de Filhol & Hadjadj (2016). Le formalisme AZee décrit dans ces recherches a comme avantage la prise en considération de facteurs non manuels, comme les expressions du visage, la spatialisation, les durées entre éléments séquentiels et les synchronisations entre ceux simultanés, qui sont très importants dans les langues des signes. Filhol et al. (2016) partent du postulat qu'une production langagière est un ensemble de formes observables et de fonctions sémantiques interprétables. La modélisation d'un lien systématique entre une forme (articulée, ou à articuler) et une interprétation (sémantique) donne lieu à une règle de production, et un ensemble (système) de telles règles pouvant couvrir tous les phénomènes linguistiques en LSF est postulé. Nous avons par exemple utilisé les règles suivantes :

- règle Context :

Fonction : L'item 1 décrit le temps, la situation, l'endroit, pendant laquelle l'item 2 se déroule ou est vrai.
Forme : juxtaposition des signes.
Exemple : item 1 = « *aujourd'hui, le 3 mai* » ; item 2 = « *élection présidentielle* » ; interprétation combinée = « *élections présidentielles d'aujourd'hui, le 3 mai* »

- règle Category :

Fonction : L'item 1 indique comment il faut interpréter l'item 2.
Forme : Le menton et/ou les sourcils sont levés au début de l'item 1, avec la tête sensiblement penchée. Il y a également une séparation courte (approximativement 100ms) entre l'item 1 et l'item 2.
Exemple : item 1 = « *pays* » ; item 2 = « *Monténégro* » ; interprétation combinée = « *Monténégro* »

- règle Side-info :

Fonction : L'item 2 donne des informations supplémentaires sur l'item 1, mais le focus de l'information est sur l'item 1.
Forme : Le menton et/ou les sourcils sont levés au début de l'item 2. La transition est minimale entre l'item 1 et l'item 2.
Exemple : item 1 = « *ville* » ; item 2 = « *le pouvoir est le tourisme* » ; interprétation combinée = « *Ville touristique* »

- règle Info-about :

Fonction : L'item 2 est l'information donnée sur l'item 1. Il y a cette fois un focus sur l'item 2.
Forme : Séparation plus longue entre les deux items (environ 340ms) et un clignement des yeux à la fin de l'item 2.
Exemple : item 1 = « *pouvoir* » ; item 2 = « *tourisme* » ; interprétation combinée = « *le pouvoir c'est le tourisme* »

- règle Séquence temporelle (SeqTemp) :

Fonction : Permet de formaliser une chronologie d'événements. Cette fonction n'est pas nécessairement binaire.
Forme : Juxtaposition des événements.
Exemple : item 1 = « *un homme entre dans un bureau de poste* » ; item 2 = « *5 personnes prises en otage pendant 3 heures* » ; interprétation combinée = « *Un homme entre dans un bureau de poste puis 5 personnes sont prises en otage pendant 3h* »

- règle SeqRes :

Fonction : Cette fonction est binaire et prend en argument deux événements E1 et E2. Cette règle est utilisée quand l'événement E2 suit immédiatement l'événement E1, ou quand l'événement E1 a été le déclencheur de l'événement E2. Il y a ici une relation de cause à conséquence implicite.

Forme : l'événement E2 est signé immédiatement derrière l'événement E1, il y a un rythme rapide. De plus, il y a un abaissement durant E2 de la tête et des sourcils, suspendus durant E1.
Exemple : item 1 = « *pluies abondantes* » ; item 2 = « *éboulement de terrain* » ; interprétation combinée = « *Glissement de terrain provoqué par des pluies abondantes* »

– règle Finger-spell :

Fonction : Dénommer un endroit, une personne ou un concept en épelant son nom lettre par lettre.
Forme : Il y a un clignement des yeux avant le début de l'épellation, et le regard est fixé sur la main qui épelle.
Exemple : item 1 = P ; item 2 = O ; item 3 = D ; item 4 = G ; item 5 = O ; item 6 = R ; item 7 = I ; item 8 = C ; item 9 = A ; interpretation combinée = "*Podgorica*"

La combinaison récursive de telles règles peut être représentée par un arbre, comme par exemple avec la fig. 3 représentant une des traductions observées de l'ALT « *jeudi soir* » présent dans la phrase (1) donnée précédemment.
L'imbrication empile, par construction, les formes associées à chacune de ces règles et détermine ainsi les formes à produire en LSF pour exprimer la combinaison sémantique ainsi représentée.
La partie notée à l'aide du symbole […] correspond à un sous-arbre dans la phrase (1). Elle décrit l'événement se déroulant le « *jeudi soir* » mais son détail n'est pas donné ici.

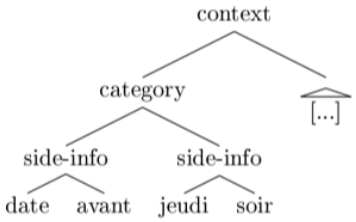


Figure 3 : Arbre représentant une des traductions observées de *jeudi soir* et son lien avec le reste de la phrase (1).

3.2 Description du corpus

Les données bilingues alignant français écrit et LSF sont très rares. Le corpus utilisé dans cette étude a été constitué par Filhol & Tannier (2013) et est le premier corpus français écrit-LSF créé dans un but de recherche académique. Il est composé de 40 brèves journalistiques en français qui ont ensuite été signées par trois traducteurs professionnels natifs en LSF, constituant ainsi un total de 120 vidéos (soit une heure de LSF journalistique). Chacune des vidéos propose une vue de face et de profil, permettant de saisir au mieux les aspects articulatoires, et donc linguistiques, de la LSF. Les traductions n'ont pas été réalisées de manière instantanée, et sont le résultat d'une préparation en amont de la part des traducteurs. Ces 40 brèves journalistiques ont été sélectionnées parmi un ensemble d'environ 980 phrases grâce à la prise en compte de quatre critères en lien avec l'analyse de la temporalité linguistique : i) la précédence chronologique entre des événements ; ii) la durée (d'un événement ou séparant deux événements distincts) ; iii) la relation de causalité entre des événements ; iv) et la notion de répétition pour un événement.

4 Formalisation et observations

En appliquant ces formalismes sur le corpus des 40 brèves, nous disposons au total de 95 ALT formalisés en français, dont 7 sont de nature calendaire (e.g. *le 1er février 2003*), 51 sont de nature déictique (e.g. *mercredi matin*), 14 sont de nature anaphorique (e.g. *la veille*), 21 sont relatifs à un événement (e.g. *Dix ans après l'évacuation musclée de l'église Saint-Bernard*).[1]

Pour les données en LSF, en prenant en compte les traductions des trois locuteurs, nous avons pu identifier un peu plus de 1 200 nœuds participant aux traductions clairement identifiables de 83 ALT de la partie écrite (française) du corpus. Les 12 traductions d'ALT non repérées sont essentiellement les ALT constitués d'une simple préposition, telle que « *après* » ou « *lors* », pour lesquels les traductions sont difficilement repérables. En prenant en compte le fait que, dans ces 83 ALT traduits, il y a eu également 8 fois un oubli de la part d'un des traducteurs, cela nous fait un total de 241 traductions d'ALT observables et une moyenne d'environ 5 nœuds pour la traduction d'un ALT par locuteur.

À partir de ces formalisations, nous avons pu faire quelques observations, que nous pourrons illustrer avec des traductions des phrases ci-dessous issues du corpus :

(2) *Le processus de sortie de coma d'Ariel Sharon se poursuivait progressivement mercredi, alors que son équipe soignante le dit hors de danger "immédiat" une semaine après son attaque cérébrale.* (phrase 1B)
(3) *Les partis de gauche se retrouvent mercredi à la Mutualité pour leur premier sommet depuis 2002, où la préparation d'actions communes prendra le pas sur celle de l'alternance en 2007, au lendemain de la journée de mobilisation contre le Contrat première embauche (CPE).* (phrase 2J)
(4) *Le Français Bernard Planche, 52 ans, retenu en otage en Irak pendant 35 jours, a retrouvé samedi la liberté près de Bagdad, et devrait très prochainement regagner la France.* (phrase 1R)
(5) *Le chef des soldats rebelles du Timor oriental, Alfredo Reinado, a opposé une fin de non recevoir mercredi aux mesures présidentielles destinées à rétablir le calme dans le pays, rendant ainsi peut probable un dénouement rapide d'une crise qui perdurent depuis plusieurs semaines.* (phrase 1I)

Parmi les 241 traductions d'ALT obtenues, 200 ont été signés en tant qu'item 1 d'une règle *context* avec l'événement décrit en tant qu'item 2 ; 30 ont été signés en tant qu'item 2 d'une règle *side-info* avec l'événement décrit en tant qu'item 1 ; et 11 ont un placement autre par rapport à l'événement qu'ils décrivent. Ce choix de placement de l'élément temporel vis-à-vis de l'événement décrit ne diffère entre les signeurs pour la traduction d'un même ALT que pour 5 ALT déictiques, 1 ALT calendaire, et 3 ALT relatifs à un événement.
Enfin, parmi ces 83 ALT traduits, 20 ont été signés de la même manière par au moins deux signeurs. Les différences notables trouvées dans ses traductions sont, par exemple, l'ajout ou non d'un signe non obligatoire tel qu'un catégorisateur comme le signe DATE, la mobilité apparente de certains signes comme par exemple le signe U-N (Until-Now / du temps s'écoule jusqu'à ce moment), ou encore des oublis partiels de la part des traducteurs.

[1] Le corpus d'origine contenait également 2 adverbes de répétition (e.g. *pour la quatrième fois*) difficilement analysables avec le formalisme de Teissèdre (2012), et que nous avons ainsi décidé de ne pas prendre en compte pour le reste de l'analyse.

Nous pouvons observer tout d'abord que l'élément temporel se place très majoritairement en tant qu'item 1 d'une règle *context*, avec l'évènement décrit en tant qu'item 2. Un exemple de ce premier placement de l'élément temporel peut se trouver dans la figure 4 qui est un exemple de traduction de l'ALT « *une semaine après* » issu de la phrase (2).

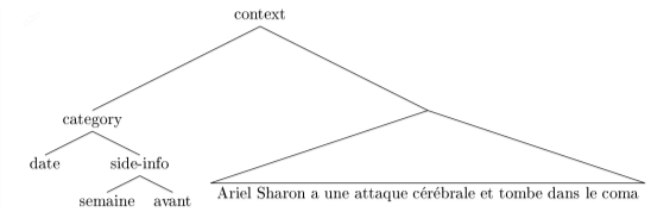


Figure 4 : Arbre représentant une des traductions observées de « *une semaine après* » et son lien avec le reste du contenu.

Le placement de l'élément temporel en tant qu'item 2 d'une règle *side-info*, avec l'événement décrit en tant qu'item 1, a aussi pu être observé. Un exemple de ce deuxième placement de l'élément temporel peut se trouver dans la figure 5 qui est un exemple de traduction de l'ALT « *depuis 2002* » issu de la phrase (3).

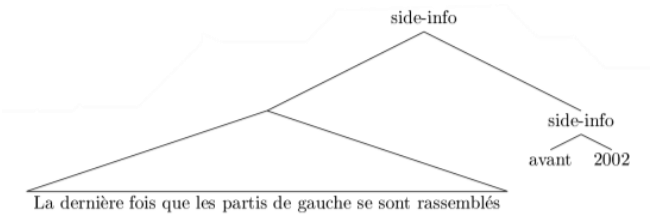


Figure 5 : Arbre représentant une des traductions observées de « *depuis 2002* » et son lien avec le reste du contenu.

Le placement des éléments temporels semble stable en LSF. En analysant plus en détail le placement de l'élément temporel en tant qu'item 2 d'une règle *side-info*, nous avons pu nous nous apercevoir que ce placement de l'élément temporel avait été utilisé pour les traductions en LSF des trois signeurs des 3 ALT du corpus à base relative à un événement ayant pour forme en français « pendant + durée ». Ce propos peut être illustré grâce à une des traductions observées du groupe prépositionnel « *pendant 35 jours* » de la phrase (5) du corpus.

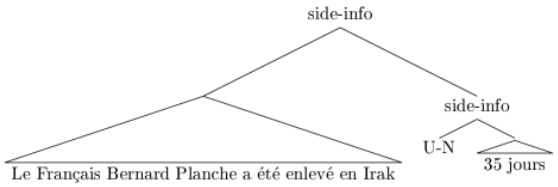


Figure 6 : Arbre représentant une des traductions observées de « *pendant 35 jours* » et son lien avec le reste du contenu.

Pour la traduction d'un petit nombre d'ALT, nous avons pu néanmoins observer le placement de l'élément temporel en tant qu'item 1 d'une règle *context* ou en tant qu'item 2 d'une règle *side-info*

indifféremment selon les signeurs, comme par exemple pour les traductions de l'ALT « *depuis plusieurs semaines* » de la phrase (5) du corpus illustrées dans les figures 7 et 8.

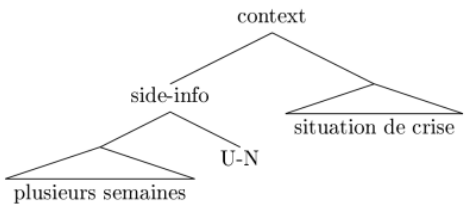


Figure 7 : Arbre représentant une première traduction observée de « *depuis plusieurs semaines* » et son lien avec le reste du contenu (placement de l'élément temporel en tant qu'item 1 d'une règle *context*).

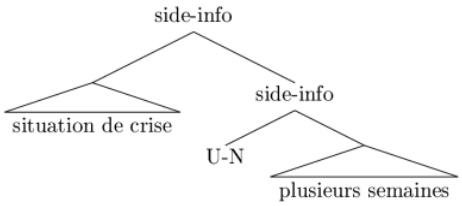


Figure 8 : Arbre représentant une seconde traduction observée de « *depuis plusieurs* semaines » et son lien avec le reste du contenu (placement de l'élément temporel en tant qu'item 2 d'une règle *side-info*).

5 Résultats

Ces observations nous ont conduits à proposer 3 types de généralisations (fig.9).

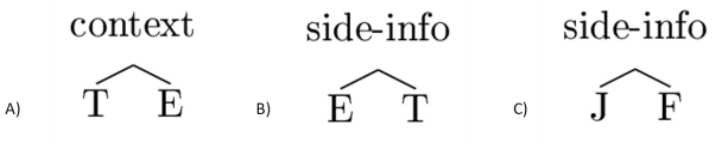


Figure 9 : Schémas des généralisations.

D'après nos résultats de formalisation, les éléments temporels en LSF - et cela peu importe leur base ou la syntaxe d'origine en français - se placent très majoritairement de deux manières distinctes vis-à-vis de l'événement qu'ils décrivent. Nous pouvons ainsi proposer deux premiers types de généralisations :

Généralisation A) : l'élément temporel, que nous noterons T, se place en tant qu'item 1 d'une règle *context* avec l'événement E qu'il décrit en tant qu'item 2. Nous avons pu voir cette généralisation avec les figures 4 et 7.

Généralisation B) : l'élement temporel T se place en tant qu'item 2 d'une règle side-info et l'événement E en tant qu'item 1. Nous avons pu voir cette généralisation avec les figures 5, 6 et 8.

Cette généralisation est utilisée pour traduire les ALT français à base relative à un événement ayant pour forme « pendant + durée » (fig. 6).

Nous avons également remarqué que pour les 5 ALT du corpus en français composé d'un jour de la semaine J ainsi que d'une focalisation concernant une partie du jour F, cela a été traduit 12 fois par l'élément J placé en tant qu'item 1 d'une règle side-info, et l'élément F en tant qu'item 2. Nous avons pu observer ceci d'ailleurs dans la figure 1. Cela correspond à la troisième généralisation que nous proposons, la généralisation C). Dans les cas où cette règle n'est pas observable, c'est qu'il y a eu un oubli partiel de la focalisation de la part d'un des traducteurs.

Les résultats sous la forme de seulement 3 types de généralisations que nous avons pu obtenir sont au moins en partie en lien avec la taille réduite et relativement peu diversifié du corpus du point de vue des types d'ALT présents. Certaines formes d'ALT ne sont ainsi pas présentes dans un nombre d'occurrences satisfaisant dans le corpus pour pouvoir faire des généralisations. Par exemple, nous ne trouvons dans le corpus que trois occurrences d'ALT possédant un opérateur de régionalisation et un opérateur de focalisation ayant une valeur différente de ID. L'impact que ces deux opérateurs peuvent avoir l'un sur l'autre ne peut donc pas être analysé de manière très approfondie.

6 Conclusion

L'objectif de cette étude était d'effectuer un premier pas vers la traduction automatique des ALT du français vers la LSF. La formalisation d'ALT issues d'un corpus en français grâce aux travaux de Teissèdre (2012), ainsi que la formalisation de leurs traductions en LSF grâce aux travaux de Filhol & Hadjadj (2016), nous ont permis d'obtenir des données formalisées que nous avons pu comparer afin de pouvoir ensuite proposer trois premières règles générales de passage d'un formalisme à l'autre permettant un début de traduction automatique d'ALT du français vers la LSF.
Ces règles de traduction proposées sont un premier apport à la question de la traduction automatique du français vers la LSF car il n'existe pas de règles déjà existantes de traductions automatiques pour la LSF. Il n'existe également pas de telles règles pour d'autres langues des signes nous permettant de vérifier si ces règles se retrouvent dans différentes langues des signes.
Ces règles restent cependant très générales et mériteraient dans le futur d'être améliorées, par exemple avec l'aide de nouvelles analyses sur de nouveaux corpus.
Il reste que cela serait grâce à ce genre de règles que nous pourrons par la suite passer à des animations à l'aide d'avatar. Ces animations pourront alors être proposées à des locuteurs natifs en LSF, ce qui permettra alors de modifier ou d'affiner d'avantage ces premières règles de traduction.

Remerciements

Ce travail s'insère dans le cadre du projet DGLFLF « *temporalité linguistique en LSF et en français écrit : création d'un outil d'aide à l'acquisition de marqueurs temporels chez l'enfant sourd signeur* » (2016-2018).

Références

BATTISTELLI D., COUTO J., MINEL J.-L., SCHWER S. (2008). Représentation algébrique des expressions calendaires et vue calendaire d'un texte. Actes de *TALN'08*, Avignon.

BATTISTELLI D. (2011). *Linguistique et recherche d'information : la problématique du temps*. Lavoisier.
EHRMANN M., HAGEGE C. (2009). Proposition de caractérisation et de typage des expressions temporelles en contexte. Actes de *TALN'09*, Senlis.
FILHOL M., HADJADJ M.N., TESTU B. (2013). A rule triggering system for automatic Text-to-Sign translation. *Universal access in the information society* 15, 487-498.
FILHOL M., HADJADJ M. N., CHOISIER A. (2014). Non-manual features: the right to indifference. In *6th Workshop on the Representation and Processing of Sign Languages: Beyond the manual channel, Language resource and evaluation*, Reykjavik, Iceland.
FILHOL M., TANNIER X. (2014). Construction of a French–LSF corpus. In *Language Resources and Evaluation Conference, Building and Using Comparable Corpora*.
FILHOL M., HADJADJ M. N. (2016). Juxtaposition as a form feature - syntax captured and explained rather than assumed and modelled. In *Language resources and evaluation conference (LREC), Representation and processing of Sign Languages.*
FUSELLIER-SOUZA I., LEIX J. (2005). L'expression de la temporalité en Langue des Signes Française (LSF). Actes du *colloque Conceptualisation et Surdité*, 207-230.
GOFfiC P. L. (1994). *Grammaire de la phrase française*. Hachette.
KERVAJAN L. (2011). *Contribution à la traduction automatique Français/Langue des Signes Française (LSF) au moyen de personnages virtuels*. PhD Linguistique, Université de Provence - Aix-Marseille.
MAEDER C. (1994). *Espace, temps et relations temporo-logiques chez le sujet sourd : étude comparative de sujets sourds et entendants dans le maniement des marqueurs spatio-temporels en LSF et en français*. PhD thesis. Université de Nancy II - U.F.R des Sciences du Langage, France.
MORRISSEY S. (2008). *Data-driven Machine Translation for Sign Languages*. PhD thesis, Dublin City University, Dublin, Ireland.
PUSTEJOVSKY P., CASTAÑO J., INGRIA R., SAURI R., GAIZAUSKAS R., SETZER A., KATZ G. (2003). TimeML: Robust Specification of Event and Temporal Expressions in Text. In *Proceedings of the Fifth International Workshop on Computational Semantics (IWCS-5)*, Tilburg.
SINTE A. (2015). *Le temps en langue des signes*. Presses universitaires de Rennes.
TEISSEDRE C. (2012). *Analyse sémantique automatique des adverbiaux de localisation temporelle : application à la recherche d'information et à l'acquisition de connaissances*. PhD thesis. Université Paris X – Nanterre, France.
WILMET M. (2003). *Grammaire critique du français*. Duculot.
WOLFE R., ETHIMIOUS E., GLAUERT J., HANKE T., MCDONALD J., SCHNEPP J. (2016). Special issue on recent advances in sign language translation and avatar technology. *Universal Access in the Information Society* 15, 485–486.