Notes de lecture

Rubrique préparée par Denis Maurel

Université de Tours, LIFAT (Laboratoire d'informatique fondamentale et appliquée)

Kyle GORMAN, Richard SPROAT. Finite-State Text Processing. *Morgan & Claypool publishers*. 2021. 158 pages. ISBN : 9-781-63639-115-1.

Lu par **Gabriel BERNIER-COLBORNE**

Conseil national de recherches du Canada

Kyle Gorman (City University of New York ; Google) et Richard Sproat (Google) nous offrent cet ouvrage qui se veut une nouvelle référence sur le formalisme des transducteurs finis pondérés (TFP), couramment utilisé pour le traitement et la génération du texte et de la parole. Il présente également Pynini, une bibliothèque pour Python développée chez Google qui permet de construire, de combiner, d'optimiser et d'appliquer des TFP. L'ouvrage contient notamment de nombreux trucs du métier qui en font une référence unique à ce sujet.

Contrairement à beaucoup d'ouvrages dans la collection *Synthesis Lectures on Human Language Technologies*, celui-ci n'est pas axé sur de nouvelles méthodes ou de nouvelles applications du traitement automatique des langues (TAL), mais cherche plutôt à fournir une introduction et une référence pratique au sujet d'un formalisme qui joue depuis longtemps un rôle important dans le développement de technologies langagières.

Cet ouvrage vise principalement les ingénieurs et les linguistes ; à notre avis, il pourrait également servir de support pour un cours, en raison du mélange habile de théorie et d'applications concrètes qu'il présente. Les lecteurs intéressés doivent posséder des connaissances de base en linguistique, en particulier en morphologie et en phonologie, et une maîtrise de base du langage Python, ainsi qu'une certaine familiarité avec la notation phonétique.

Le premier chapitre introduit des concepts fondamentaux et leur notation : ensembles, relations, fonctions, chaînes de caractères, langages, etc. Il explique notamment la notion de demi-anneau, un concept essentiel pour maîtriser la théorie des automates, quoique souvent négligé dans la littérature existante, selon les auteurs. C'est certainement l'un des concepts les plus difficiles à appréhender au début, mais les auteurs ont tout à fait raison de l'inclure.

Une fois cette base théorique établie, les auteurs expliquent les conventions de la bibliothèque Pynini pour Python, utilisée par la suite pour illustrer divers algorithmes sur les automates. Ils présentent d'abord la bibliothèque sous-jacente

OpenFST pour C++. Pynini comprend plusieurs algorithmes qui sont absents de cette dernière (et de pywrapfst, qui permet d'utiliser l'interface de script de OpenFST en Python). Elle fournit des méthodes pour la conversion entre automates et chaînes de caractères, des opérateurs pour le produit et la répétition bornée (en anglais, *range concatenation* ; par exemple, l'expression régulière *[0-9]{m,n}* représente une séquence qui contient entre *m* et *n* chiffres), des routines pour optimiser les automates, c'est-à-dire minimiser le nombre d'états, et des méthodes pour compiler un transducteur à partir d'un ensemble de règles de réécriture. Ces ajouts facilitent le développement de grammaires et d'automates de toutes sortes : plutôt que de manipuler directement les états et transitions d'un automate, on utilise des opérateurs de plus haut niveau, tels que l'union et la composition, éventuellement suivis d'une optimisation (ou de la simple suppression de transitions vides).

Les chapitres 3 et 4 introduisent des opérations et des algorithmes sur les automates, tels que la concaténation, la fermeture de Kleene et l'union, ainsi que divers algorithmes utilisés pour combiner et optimiser les automates, et pour y chercher les plus courts chemins, ce qui permet de décoder les TFP utilisés pour différentes applications du TAL, telles que la reconnaissance automatique de la parole. La recherche des plus courts chemins permet également d'inspecter des exemples de chaînes reconnues par un accepteur ou transformées par un transducteur. Pour chaque opération ou algorithme, on nous fournit une caractérisation algébrique, des extraits de code qui illustrent sa mise en application et une description de l'automate résultant.

Pour bien évaluer cet ouvrage et maîtriser son contenu, je me suis efforcé de reproduire au moyen de Pynini tous les automates présentés comme exemples, pour ensuite exécuter les extraits de code illustrant les opérations ou algorithmes sur ces automates. J'y suis arrivé sans trop de difficultés dans presque tous les cas, mais je note qu'il aurait été utile d'expliquer comment utiliser la méthode *draw* de la classe *Fst* pour sauvegarder une représentation textuelle (en langage DOT) du graphe dirigé correspondant à un automate donné, pour ensuite le dessiner au moyen de la commande *dot* de Graphviz (ou d'autres outils permettant de visualiser des graphes). Cette méthode est très utile pour s'assurer que l'on construit correctement un automate donné ou pour vérifier le résultat d'une opération sur des automates.

Les chapitres 5 à 7 sont consacrés aux règles de réécriture et à leur application à divers problèmes du TAL. On apprend comment compiler un transducteur à partir d'un ensemble de règles de réécriture, construire une cascade de règles, appliquer les règles à des chaînes de caractères et construire des paradigmes morphologiques. Puis, on apprend comment ces méthodes sont utilisées dans le cadre d'applications concrètes : la conversion entre graphèmes et phonèmes, la génération et l'analyse morphologiques, la recherche approximative de chaînes de caractères, la normalisation de texte, le décodage de séquences numériques encodées au moyen du système de saisie T9, etc. En ce qui concerne les paradigmes morphologiques, on apprend comment traiter systématiquement l'accord des noms en russe, le focus agent (ou acteur) des verbes en tagalog et l'aspect verbal en yawelmani.

Le dernier chapitre explore l'avenir des technologies basées sur les transducteurs finis pondérés dans un monde où les réseaux de neurones occupent de plus en plus une position dominante. Les pistes de recherche évoquées comprennent le développement de technologies hybrides qui combinent transducteurs et réseaux de neurones.

À la fin de chaque chapitre, on trouve des recommandations de lectures complémentaires pour les lecteurs intéressés. La bibliographie est vaste et détaillée, les entrées comprenant un DOI lorsque celui-ci est disponible. Enfin, les annexes expliquent comment installer Pynini et décrivent différents modules compris dans cette bibliothèque.

Il faut reconnaître que la durée de vie de cet ouvrage dépendra en partie de la stabilité et du niveau d'adoption de la bibliothèque Pynini. Cette dernière a atteint un certain niveau de maturité, mais continuera d'évoluer, on l'espère, ce qui rendra forcément certaines parties de cet ouvrage désuètes. Les auteurs sont conscients de ce problème, mais osent espérer que cet ouvrage constituera tout de même une référence utile et durable, grâce au mélange de théorie, d'algorithmes, d'applications et de code qu'il présente.

J'ai choisi de lire cet ouvrage parce que je désirais mieux maîtriser ce formalisme et son application à divers problèmes concrets du TAL, et j'y ai trouvé le contenu que je recherchais, expliqué d'une façon claire et concise, et illustré au moyen d'exemples pertinents. Je n'hésiterais donc pas à le recommander à quiconque s'intéresse aux transducteurs finis et à leur application aux problèmes du traitement automatique des langues.

Mohammad Taher PILEHVAR, Jose CAMACHO-COLLADOS. Embeddings in Natural Language Processing: Theory and Advances in Vector Representations of Meaning. ***Morgan & Claypool publishers*****. 2020. 175 pages.**
ISBN : 9-781-63639-023-9.

Lu par **Caio Filippo CORRO**

Université Paris-Saclay – CNRS, Laboratoire Interdisciplinaire des Sciences du Numérique (LISN)

Cet ouvrage propose une vue d'ensemble sur les différentes techniques d'apprentissage de représentations lexicales.

Contenu de l'ouvrage

L'apprentissage de représentations lexicales sous forme de plongements de mots connaît une grande popularité en traitement automatique des langues (TAL), en partie car ces derniers peuvent être utilisés pour améliorer les performances des systèmes. En effet, les classificateurs neuronaux prenant en entrée du texte utilisent des tables de plongements pour associer une représentation continue à chaque mot (voire parfois en utilisant une segmentation plus fine). Ces plongements peuvent être appris de bout en bout sur les données d'apprentissage de la tâche cible. Cependant, les données annotées sont limitées car souvent coûteuses à obtenir. Au contraire, il est aisé de récupérer de grands corpus de textes bruts grâce aux archives de livres numériques et au *Web scraping*. Ces données peuvent être utilisées pour de l'apprentissage par transfert : les plongements lexicaux sont appris *via* un problème auxiliaire (voir ci-dessous), puis sont utilisés pour initialiser les plongements lexicaux des réseaux de neurones utilisés pour la tâche cible. Ceux-ci peuvent, soit rester fixés, soit être ajustés.

Ces dernières années, la notion de plongement contextuel a été largement popularisée : au lieu d'apprendre seulement des représentations de mots fixes, c'est-à-dire des tables de vecteurs, ce sont des couches entières de réseaux de neurones qui sont apprises (LSTM ou réseaux attentionnels). Ces couches sont ensuite « connectées » au réseau utilisé pour la tâche cible. En regardant les publications des dernières conférences (ACL, EMNLP, etc.), on observe qu'une grande partie des travaux de la communauté se focalise sur ces représentations contextuelles, que ce soit en proposant de nouvelles architectures et des adaptations à des langues, en appliquant ces réseaux à diverses tâches ou encore en développant des analyses des représentations apprises. Cet ouvrage, qui retrace l'historique des différentes méthodes d'apprentissage de représentations lexicales, permet d'avoir un panorama sur ces recherches. Autre point important et utile : chaque chapitre s'intéressant à un type de représentation se termine par un rappel des méthodes d'évaluation intrinsèques et extrinsèques.

Contenu de l'ouvrage

L'ouvrage commence par une courte introduction au concept de représentation lexicale (chapitre 1) et à l'apprentissage profond pour le TAL ainsi qu'une rapide liste des bases de données lexicales disponibles (chapitre 2).

Le chapitre 3 se concentre sur les plongements de mots non contextualisés appris sur des données textuelles. Les auteurs commencent par une présentation des méthodes préneuronales dites méthodes par comptage (*count-based models*, parfois aussi appelées modèles creux) fondée sur l'hypothèse distributionnelle : les mots apparaissant dans des contextes similaires tendent à avoir un sens similaire. Les méthodes par comptage fonctionnent par construction d'une matrice de cooccurrences (combien de fois tel mot apparaît avec tel autre mot dans son contexte), suivie de plusieurs étapes de post-traitement. En particulier, notons l'application de méthodes de réduction de la dimensionnalité qui rend denses ces représentations parfois qualifiées de creuses. Les méthodes par prédiction (*word2vec*, *skip-gram*) sont ensuite traitées très brièvement.

Le chapitre 4 traite des plongements de graphes et plus particulièrement des plongements de nœuds. En TAL, ce problème est présent lorsque l'on veut apprendre des représentations de concepts ou de mots à partir de bases de connaissances où ils sont représentés par des nœuds dans un graphe. À la différence de l'utilisation de textes bruts, les graphes de connaissances contiennent explicitement des relations entre les mots. En d'autres termes, deux noms de villes vont apparaître avec le même genre de relations voisines (pays, région, code postal, nombre d'habitants…), qui peuvent être comprises comme la similarité de contexte. Cependant, l'apprentissage de représentations à partir de graphes diffère de plusieurs façons :

- le nombre de relations lorsque l'on regarde plus loin que simplement les voisins de premier niveau devient très rapidement ingérable. Pour s'attaquer à ce problème, il est possible, par exemple, d'extraire des caractéristiques de façon stochastique avec des marches aléatoires ;
- du cadre neuronal, il faut utiliser des architectures neuronales capables de prendre en compte la structure de graphe de l'entrée. Une méthode populaire est l'utilisation de réseaux convolutifs sur graphe ;
- un concept n'apparaît qu'une fois dans un graphe, cela peut mener à des problèmes de surapprentissage avec l'utilisation des réseaux de neurones.

Le chapitre fait un exposé bref mais clair de ces différents sujets en renvoyant vers des articles de la littérature.

Le chapitre 5 traite du problème de la désambiguïsation lexicale. En effet, l'apprentissage de représentations lexicales se heurte au problème de la polysémie : un même mot peut véhiculer plusieurs sens différents en fonction du contexte dans lequel il est utilisé (le terme « avocat » peut, soit référer au fruit, soit au métier). Il n'est donc pas évident qu'il faille se limiter à apprendre une unique représentation par mot, peut-être faudrait-il plutôt apprendre une représentation par sens. Les auteurs commencent par décrire les approches non supervisées, c'est-à-dire les approches qui n'utilisent ni d'informations *a priori* sur le nombre de sens qu'un mot particulier peut avoir, ni d'annotations de sens dans les corpus. Dans le cas d'apprentissage à partir d'un corpus monolingue, cette tâche s'apparente à un problème de *clustering* : arriver à différencier de façon non supervisée le sens véhiculé par les différentes occurrences d'un même mot. Une autre approche décrite par les auteurs est l'utilisation de corpus multilingues qui permet d'extraire des informations sur le sens à partir d'alignements. Par exemple, le mot anglais *crane* peut être traduit en *grulla* ou *grúa* en espagnol. Cette information d'alignements permet donc d'identifier les différents sens de *crâne*. La dernière partie du chapitre décrit les méthodes d'apprentissage à partir de bases de connaissances.

Les plongements contextualisés sont traités dans le chapitre 6, chapitre le plus long de l'ouvrage. Après une courte introduction, les réseaux attentionnels et les *transformers* sont présentés (ceux-ci ne sont pas décrits dans le chapitre 2). Ensuite, les auteurs proposent un court historique sur les différents modèles de plongements contextualisés, suivi de deux sections consacrées à BERT et ses extensions. Enfin, les auteurs font une brève description des méthodes d'analyse de ces réseaux préentraînés.

Les chapitres 7 et 8, très courts, se focalisent sur les plongements de phrases et de documents et sur les problèmes d'éthique et de biais, respectivement.

Conclusion

L’ouvrage propose un panorama sur les différentes méthodes de construction de représentations lexicales. Il sera utile à des chercheuses et chercheurs voulant rapidement trouver des pointeurs vers la littérature. Je tiens cependant à préciser que le choix fait par les auteurs a été de maximiser la couverture des sujets traités plutôt que d’approfondir une direction de recherche, contrairement, par exemple, à un ouvrage précédent de la collection[1] qui se focalise uniquement sur les plongements multilingues, sujet traité ici en moins de quatre pages. Par exemple, si on s’intéresse à l’apprentissage de représentations non contextuelles, aucun recul n’est pris les travaux, aucune tentative de créer une typologie ou proposer un cadre général pour la construction de ces représentations. Les travaux de Omer Levy et Yoav Goldberg sont simplement cités dans l’ouvrage alors qu’ils ont permis par le passé de comprendre le lien entre les méthodes par comptage et par prédiction. Les nombreux travaux théoriques et pratiques sur les fonctions de perte utilisées dans les méthodes par prédiction qui permettent de mieux comprendre l’objectif atteint par des approximations du type *negative sampling* ne sont pas expliqués en détail. L’analyse des réseaux de neurones, souvent considérés comme des boîtes noires, ainsi que les questions d’éthique et de biais sont devenues des problématiques de recherche importantes dans le domaine, mais sont à peine traitées par les auteurs.

Lucia SPECIA, Carolina SCARTON, Gustavo Henrique PAETZOLD. Quality Estimation for Machine Translation. *Morgan & Claypool publishers*. 2018. 148 pages. ISBN : 978-1-68173-375-3.

Lu par **Rémi CARDON**

Cental – Université Catholique de Louvain

Après une brève introduction qui dessine les contours du sujet traité, cet ouvrage dresse un panorama des méthodes d’estimation de qualité (désormais QE pour « quality estimation *») des systèmes de traduction automatique. Les chapitres 2 à 4 décrivent ces méthodes pour trois unités de traitement : le niveau sous-phrastique, le niveau de la phrase et le niveau du document. Le cinquième chapitre évoque les méthodes de QE explorées dans des domaines connexes : la simplification automatique, le résumé automatique, la correction d’erreurs grammaticales, la reconnaissance vocale et la génération automatique de texte. L’ouvrage se conclut par une brève évocation des directions à venir ainsi que par une liste de ressources et outils disponibles pour la QE.*

L’ouvrage traite de la QE appliquée à des systèmes qui produisent du texte et se concentre principalement sur le domaine de la traduction automatique. Les méthodes d’évaluation des systèmes de traduction automatique reposent le plus souvent sur l’utilisation de métriques comparant la sortie des systèmes à des jeux de données de référence (des corpus parallèles). En quelques mots, la recherche en QE s’attache à

1 Celui d’Anders Søgaard *et al.*

concevoir et évaluer des méthodes qui s'affranchissent de la nécessité de disposer de corpus de référence. Le chapitre d'introduction part de cette description et évoque les principes et les motivations de la QE. Ce chapitre est bref, les principes et motivations sont exposés sans être discutés. Il semble également manquer une discussion autour des autres paradigmes d'évaluation des systèmes de traduction automatique. Notamment, il est curieux de ne pas trouver une seule mention de BLEU dans un livre qui évoque l'évaluation des systèmes de traduction automatique, ne serait-ce que pour montrer comment s'en distinguent les méthodes de QE que le livre décrit. Après la définition du sujet, l'introduction établit un historique de la QE pour la traduction automatique à travers de nombreuses citations de travaux de recherche et la présentation de *shared tasks* dédiées au sujet, notamment dans le cadre de plusieurs éditions du *workshop* WMT. Enfin, les différents niveaux de traitement de la QE sont exposés (mot/syntagme, phrase, document). Cette catégorisation sert de plan pour les trois chapitres suivants.

Les chapitres 2 (niveaux lexical et syntagmatique), 3 (niveau de la phrase) et 4 (niveau du document) traitent du sujet principal du livre : la QE pour la traduction automatique. Ils sont tous les trois organisés de la même manière et contiennent les parties suivantes : une introduction, un historique des applications, une description des *labels* utilisés pour la réalisation de la tâche, un inventaire des descripteurs utilisés pour la réalisation de la tâche, une présentation de différentes architectures utilisées, une description des méthodes d'évaluation des approches présentées et une présentation des approches et résultats correspondant à l'état de l'art. Cette structure commune aux trois chapitres aide à la clarté du propos.

Le chapitre 5 établit un panorama des pratiques de QE pour d'autres tâches du traitement automatique des langues : la simplification automatique, le résumé automatique, la correction d'erreurs grammaticales, la reconnaissance vocale et la génération automatique de texte. La simplification automatique de textes occupe une grande partie de ce chapitre, alors que les autres sont traitées plus rapidement (douze pages pour la simplification automatique et d'une demi-page à cinq pages pour chaque autre tâche). L'importance donnée à cette partie sur la simplification automatique peut s'expliquer par le fait que les trois auteurs de l'ouvrage en sont des spécialistes. Cette partie est structurée de la même manière que les chapitres 2 à 4. Cette reproduction du traitement appliqué aux chapitres sur la traduction automatique semble cohérente car la simplification automatique est le plus souvent explorée comme une tâche de traduction monolingue, où la langue source est la langue complexe et la langue cible la langue simple. Il est surprenant de voir si peu de place accordée à la QE pour la reconnaissance vocale, car dans le chapitre 2 les auteurs expliquent à plusieurs reprises que la QE au niveau lexical pour la traduction automatique s'inspire grandement de la QE appliquée à cette tâche.

Le dernier chapitre s'articule en deux parties : une brève conclusion qui résume le contenu des chapitres précédents et une liste de ressources et outils disponibles en QE pour la traduction automatique.

Tout au long de l'ouvrage, les auteurs s'appuient sur de nombreux schémas explicatifs pour illustrer les concepts et systèmes présentés. Sur les aspects techniques abordés à propos des outils utilisés pour la QE, certains passages s'adressent à un lecteur novice. Par exemple dans la section 2.5.2 (sur les approches séquentielles pour la QE au niveau lexical et syntagmatique) les auteurs présentent les CRF, définissent les réseaux de neurones et décrivent le fonctionnement des réseaux de neurones récurrents, à l'aide de schémas et en incluant des références pertinentes. Cependant, quelques pages plus loin (p. 33) les auteurs présentent un réseau de neurones tiré de l'état de l'art qui comporte une couche de plongements de mots et une couche d'attention. Ces deux derniers éléments ne sont pas présentés par les auteurs et aucune référence n'est présente pour aider le lecteur à les élucider spécifiquement. Néanmoins, les notions qui relèvent spécifiquement de la QE sont systématiquement définies et des références sont fournies au lecteur.

D'une façon générale, cet ouvrage représente un inventaire structuré des pratiques, ressources et outils en QE pour la traduction automatique. Il est en cela une ressource utile pour quiconque ayant des connaissances préalables en TAL et qui voudrait se renseigner sur le sujet abordé. L'aspect d'inventaire s'illustre le mieux avec les parties qui présentent les différents descripteurs utilisés par les systèmes de QE. Il s'agit de listes à puces, non commentées et sans références. Dans les autres parties, les références sont nombreuses et les auteurs décrivent en détail les éléments qu'ils abordent (*labels*, applications, systèmes…). L'ouvrage regorge donc d'informations pertinentes, cependant ces informations ne sont pas discutées ou mises en perspective. Aucune orientation ou prise de position ne vient accompagner ces informations brutes.

Pour conclure, notons que le livre est sorti en 2018 et que cette note de lecture a été rédigée en 2021. Les travaux les plus récents en QE pour la traduction automatique (notamment dans le cadre du *workshop* WMT qui continue de proposer des *shared tasks* sur le sujet) intègrent les dernières avancées du TAL et notamment l'utilisation de modèles *transformers* tels que BERT. L'ouvrage présenté ici n'est pas obsolète pour autant. Bien entendu, les parties décrivant les systèmes de l'état de l'art doivent être lues en gardant à l'esprit l'année de publication du livre, mais tout ce qui concerne les *labels*, les domaines d'application, les descripteurs et l'évaluation reste d'actualité.