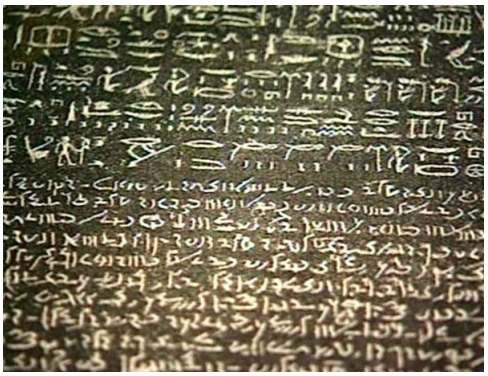
TextMine '22

Atelier sur la Fouille de Textes



Organisateurs :

Pascal Cuxac (INIST - CNRS),
Cédric Lopez (Emvista),
Vincent Lemaire (Orange Labs),

Organisé conjointement à la conférence EGC
(Extraction et Gestion des Connaissances)
le 25 janvier 2022 à Blois

Editeurs :

Pascal Cuxac - INIST - CNRS
2 rue Jean Zay, CS 10310, 54519 Vandoeuvre les Nancy Cedex
Email : pascal.cuxac@inist.fr

Vincent Lemaire - Orange Labs
2 avenue Pierre Marzin, 2300 Lannion
Email : vincent.lemaire@orange.com

Cédric Lopez - Emvista
Cap Oméga, Rond-Point Benjamin Franklin, CS 39521, 34960 Montpellier Cedex 02
Email : cedric.lopez@emvista.com

———

Publisher:

Vincent Lemaire, Pascal Cuxac, Cédric Lopez
2 avenue Pierre Marzin
22300 Lannion

Lannion, France, 2022

PRÉFACE

C'est une évidence que de dire que nous sommes entrés dans une ère où la donnée textuelle sous toute ses formes submerge chacun de nous que ce soit dans son environnement personnel ou professionnel : l'augmentation croissante de documents nécessaires aux entreprises ou aux administrations, la profusion de données textuelles disponibles via Internet, le développement des données en libre accès (OpenData), les bibliothèques et archives en lignes, les medias sociaux ne sont que quelques exemples illustrant l'évolution de la notion de texte, sa diversité et sa prolifération.

Face à cela les méthodes automatiques de fouille de données (data mining), et plus spécifiquement celles de fouille de textes (text mining) sont devenues incontournables. Récemment, les méthodes de deep learning ont créées de nouvelles possibilités de recherche pour traiter des données massives et de grandes dimensions. Cependant, de nombreuses questions restent en suspens, par exemple en ce qui concerne la gestion de gros corpus textuels multi-thématiques. Pouvoir disposer d'outils d'analyse textuelle efficaces, capables de s'adapter à de gros volumes de données, souvent de nature hétérogène, rarement structurés, dans des langues variées, des domaines très spécialisés ou au contraire de l'ordre du langage naturel reste un challenge.

La fouille de textes couvre de multiples domaines comme, le traitement automatique des langues, l'intelligence artificielle, la linguistique, les statistiques, l'informatique et les applications sont très diversifiées, que ce soit la recherche d'information, le filtrage de spam, le marketing, la veille scientifique ou économique, la lutte antiterroriste...

Le but de cet atelier est de réunir des chercheurs sur la thématique large de la fouille de textes. Cet atelier vise à offrir une occasion de rencontres pour les universitaires et les industriels, appartenant aux différentes communautés de l'intelligence artificielle, l'apprentissage automatique, le traitement automatique des langues, pour discuter des méthodes de fouille de textes au sens large et de leurs applications.

P. Cuxac — INIST-CNRS
C. Lopez — Emvista
V. Lemaire — Orange Labs

Membres du comité de lecture

Le Comité de Lecture est constitué de:

Vincent Claveau (IRISA, Rennes)

Kevin Cousot (Emvista, Montpellier)

Amael Dago (INIST-CNRS, Nancy)

Nicolas Dugué (LIUM, Le Mans)

Natalia Grabar (STL - Lille3, Lille)

Jean-Charles Lamirel (LORIA, Nancy)

Sonia Le Meitour (Orange Labs, Lannion)

Denis Maurel (Lifat, Université F. Rabelais, Tours)

David Reymond (Université de Toulon, Toulon)

Wissam Siblini (Wordline, Lyon)

Kai Song (Dalian University of Technology, Dalian, Chine)

TABLE DES MATIÈRES

Vers des approches de plongements interprétables ?

Nicolas Dugué
Le Mans Université, LIUM, EA 4023, Laboratoire d'Informatique de l'Université du Mans

nicolas.dugue@univ-lemans.fr

Résumé :

Les approches récentes d'apprentissage de plongements lexicaux ont mis l'accent sur les résultats, souvent au détriment de l'interprétabilité et de la complexité algorithmique. Pourtant, l'interprétabilité est un pré-requis nécessaire à la mise en œuvre de telles technologies lorsqu'elles sont au service de domaines sensibles comme le domaine juridique ou la médecine. Par ailleurs, les impératifs écologiques créent une urgence à réfléchir à des systèmes performants et économes en calculs. Nous proposons dans le cadre de l'ANR DIGING de développer une nouvelle approche performante et économe en calculs pour la construction de plongements lexicaux interprétables basée sur la théorie des réseaux complexes. Nous discuterons d'une première méthode développée dans ce cadre, SINr (Sparse Interpretable Node Representation) qui propose une méthode unifiée pour l'apprentissage de plongements de graphes et de mots dans un espace aux dimensions tangibles, en complexité quasi-linéaire.

Classification interprétable de documents à l'aide d'un réseau de neurones opérant sur des graphes

Adrien Guille, Hugo Attali

Université de Lyon, Lyon 2, ERIC UR 3083
adrien.guille@univ-lyon2.fr, hugo.attali@univ-lyon2.fr

Résumé. Diverses architectures de réseaux de neurones ont été explorées, comme les réseaux convolutifs et récurrents, et récemment le Transformer, pour la classification de documents. En parallèle, les réseaux de neurones sur données graphes ont largement progressé. Dans cet article, nous proposons d'encoder les documents comme des graphes orientés acycliques décrivant la structuration en phrases des textes. Nous présentons DocGat, un réseau de neurones basé sur le mécanisme d'attention et opérant sur ces graphes pour la classification de documents. Les expériences menées sur des jeux de données variés montrent que DocGat atteint des résultats concurrençant l'état de l'art. Nous montrons aussi qu'en raison du nombre relativement faible de paramètres, DocGat est avantageux en pratique puisqu'il peut facilement être entraîné sur de grands corpus, ou être appliqué à de longs documents. Enfin, nous mettons en avant l'interprétabilité de ses prédictions, une propriété utile dans de nombreux scénarios.

1 Introduction

De nombreuses architectures ont été étudiées pour résoudre la classification de documents : réseaux de neurones convolutifs (Kim, 2014; Liu et al., 2017), réseau de neurones récurrents (Cho et al., 2014; Adhikari et al., 2019) et plus récemment le Transformer (Vaswani et al., 2017). Alors que des modèles de langage profonds pré-entraînés basés sur le Transformer, comme BERT (Devlin et al., 2019), peuvent être spécialisés sur une tâche de classification de documents, le fait d'avoir de l'ordre de 100 millions de paramètres les rend parfois trop coûteux voire impossibles à mettre en place. De plus leurs prédictions sont difficilement interprétables en raison du mécanisme d'attention multi-têtes et au grand nombre de couches successives. En parallèle, les progrès des réseaux de neurones sur les graphes les rendent utiles à la résolution d'une grande variété de tâches (Kipf et Welling, 2017; Veličković et al., 2018). Dans ce papier nous nous appuyons sur les récentes avancées dans ce domaine pour proposer une architecture nécessitant l'estimation d'un nombre comparativement restreint de paramètres, permettant de classifier les documents de manière inductive et interprétable.

En supposant que l'ordre des mots n'est pas important, mais que la structuration en phrases d'un document doit être préservée, nous considérons un document comme un sac de phrases et chaque phrase comme un sac de mots. Nous représentons ainsi un document comme un graphe dirigé acyclique où un sommet document est connecté a des sommets phrases, eux-mêmes connectés à des sommets mots. Pour résoudre la tâche de classification de documents,

nous proposons une nouvelle architecture opérant sur ces graphes nommée DocGAT. Ce réseau consiste essentiellement en 4 couches basée sur une tête d'attention (*i.e.* GAT) pour apprendre les représentations des mots, de phrases et des documents de façon hiérarchique. Pour atténuer le problème de sur-lissage qui survient généralement lors de l'empilement de plusieurs couches GAT, nous implémentons deux types de connexions résiduelles : une connexion résiduelle sur les poids d'attention d'une couche à l'autre et une connexion résiduelle sur les représentations en entrée et sortie des couches. Nous adoptons également une conception spécifique pour la dernière couche GAT, afin que sa sortie soit naturellement interprétable. Dans l'ensemble, cette architecture présente un nombre de paramètres comparables aux techniques CNN les plus simples, ce qui facilite son utilisation même dans des scénarios où les ressources sont limitées.

Nous étudions les performances de DocGAT sur 5 jeux de données couvrant des documents de natures différentes et montrons qu'il obtient des résultats qui concurrencent l'état de l'art. Nous révélons également que DocGAT converge beaucoup plus rapidement que les méthodes existantes comparables pour la classification de documents. Nous mettons également en lumière sa capacité à sélectionner les phrases et les mots importants dans les documents.

Pour favoriser l'utilisation de DocGAT par les praticiens et les chercheurs, nous mettons notre code en libre accès : `https://github.com/adrienguille/docgat`. De plus, les jeux de données utilisés dans nos expériences et le code de toutes les méthodes comparées sont librement disponibles en ligne, ce qui rend nos résultats entièrement reproductibles.

2 Travaux connexes

Réseaux de neurones convolutifs. Dans un article fondateur, Kim propose d'effectuer des convolutions 1D sur des séquences de représentations de mots, suivies d'un sous-échantillonnage *max pooling* global pour résoudre la classification des documents. (Kim, 2014). XML-CNN remplace le sous-échantillonnage global par un sous-échantillonnage dynamique afin de capturer des informations localisées dans les documents (Liu et al., 2017). Comme cela conduit à une représentation du document en bien plus grande dimension, il inclut également une couche dense supplémentaire pour en réduire la dimension avant la classification.

Réseaux de neurones récurrents. Bien qu'efficaces, les réseaux convolutifs sont limités dans la mesure où la convolution ne permet de saisir que les relations à courte distance (quelques mots) dans le texte. Les architectures récurrentes comme GRU (Cho et al., 2014) et LSTM (Adhikari et al., 2019) peuvent capturer des informations à plus longue distance en propageant des états cachés tout le long du texte. Yang *et al.* proposent l'approche HAN, qui traite le texte de manière hiérarchique, en encodant d'abord chaque phrase avec un GRU, puis en transmettant ces encodages des phrases à un autre GRU pour apprendre une représentation du document pour la classification (Yang et al., 2016).

Transformer. La nature récurrente des architectures GRU et LSTM rend leur entraînement et leur exécution inefficaces dans beaucoup de situations. Pour les remplacer, Vaswani *et al.* ont introduit le Transformer, un réseau profond à propagation avant, basé sur l'attention (Vaswani et al., 2017). Cela a conduit au développement de modèles de langage pré-entraînés de très

grande taille, tels que BERT (Devlin et al., 2019), avec des centaines de millions de paramètres, qui peuvent être affinés pour des tâches comme la classification de documents.

Réseaux de neurones sur les graphes. En parallèle, d'importants progrès ont été réalisés dans le domaine de l'apprentissage sur les graphes. La convolution sur des données structurées en graphe peut être efficacement approchée linéairement dans le domaine spectral avec le GCN (Kipf et Welling, 2017). L'approche GAT remplace les poids de convolution fixés du GCN par des paramètres estimés via un mécanisme d'attention multi-têtes (Veličković et al., 2018). Yao *et al.* proposent de résoudre la classification de documents de manière transductive en appliquant le GCN à un unique graphe encodant tout un corpus, avec des sommets à la fois pour les documents et pour les mots (Yao et al., 2019), tel que les mots soient reliés aux documents dans lesquels ils apparaissent et que les mots soient reliés entre eux en fonction des cooccurrences. L'approche MPAD (Nikolentzos et al., 2020) permet quant à elle la classification inductive. Un document est codé comme un graphe, avec un sommet pour chaque mot distinct, relié aux autres par des arêtes dirigées et pondérées en fonction de la cooccurrence, plus un sommet principal relié à tous les mots (avec un poids de 1). Les représentations des mots sont apprises itérativement par passage de messages, avec un perceptron multi-couches et un GRU (Cho et al., 2014) pour combiner les représentations entre chaque itération. Un mécanisme d'attention agrège finalement toutes les représentations des sommets pour former une représentation globale du document pour la classification.

Dans cet article, nous nous appuyons sur les récentes avancées en matière de réseaux de neurones sur les graphes pour concevoir une nouvelle architecture, interprétable et efficace pour, permettant la classification inductive de documents.

3 Proposition

D'abord, nous expliquons comment encoder les documents sous forme de graphes. Ensuite, nous décrivons les composants du réseau de neurones que nous proposons pour apprendre les représentations des mots, des phrases et des documents de façon hiérarchique. Enfin, nous montrons comment estimer ses paramètres pour la classification de documents.

3.1 Modélisation du document

Considérons un document D, composé de n_s phrases et de n_w mots distincts. Supposant que l'ordre des mots n'est pas utile mais que la structuration en phrases doit être préservée, nous codons ce document sous la forme d'un graphe acyclique dirigé, $G = (V, E)$. Il comporte $N = 1 + n_s + n_w$ sommets, à savoir un sommet pour le document, un sommet pour chaque phrase et un sommet pour chaque mot distinct. Le sommet du document (que l'on numérote 0) a des arcs vers tous les sommets des phrases (numérotés de 1 à n_s), tandis que chaque phrase a des arcs vers tous les mots qu'elle comporte. Ce graphe est caractérisé par une matrice d'adjacence non symétrique $\mathbf{A}\{0,1\}^{N\times N}$. La figure 1 illustre un tel graphe.

Nous couplons ce graphe avec les attributs des sommet $\mathbf{X} \in \mathbb{N}^{N\times 1}$, qui sont les identifiants qui les relient aux représentations initiales en entrée du réseau de neurones. Le sommet du document a un identifiant spécial, 0, et tous les sommets des phrases ont le même identifiant spécial, 1.

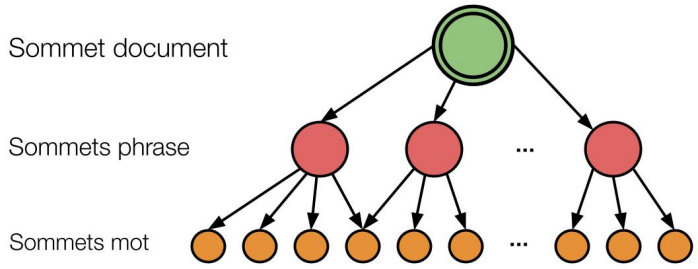


FIG. 1 – *Modélisation du document : un Graphe acyclique dirigé.*

3.2 Architecture du modèle

DocGAT opère sur les deux entrées que nous venons de décrire, à savoir une matrice d'adjacence $\mathbf{A} \in \{0;1\}^{N\times N}$ et une matrice d'attributs $\mathbf{X} \in \mathbb{R}^{N\times 1}$, et vise à apprendre les représentations des mots, des phrases et du document de manière hiérarchique. L'architecture consiste en une couche de représentations initiales suivie de L couches reposant sur une seule tête d'attention (*i.e.* couches GAT) (Veličković et al., 2018). Pour atténuer le phénomène de surlissage qui se produit lors de l'empilement de plusieurs couches GAT (Chen et al., 2020), nous implémentons deux types de connexions résiduelles : une connexion résiduelle sur les poids d'attention et une connexion résiduelle sur les représentations des sommets. La Fig. 2 (page 6) montre un réseau DocGAT à 4 couches, comme celui utilisé dans nos expériences. Dans ce qui suit, nous décrivons la conception générale de nos couches de type GAT et nous concluons en décrivant la conception spécifique de la dernière couche afin d'assurer l'interprétabilité de la sortie du réseau.

Attention contrainte par le graphe. Nous considérons la forme d'attention la plus simple, l'attention à une tête, car nous n'avons empiriquement trouvé aucun avantage à utiliser plusieurs têtes d'attention. La l-ème couche reçoit la matrice d'adjacence $\mathbf{A}$, dont nous fixons la diagonale à 1 pour forcer des boucles sur tous les sommets, et les représentations des sommets d'entrée $\mathbf{H}^{(l-1)}$; elle calcule les représentations des sommets mises à jour $\mathbf{H}^{(l)}$. Lorsque $l = 1$, l'entrée $\mathbf{H}^{(0)}$ corresponds aux représentations initiales.

Tout d'abord, nous calculons les scores d'attention entre les sommets. Si $a_{ij} = 0$, le score d'attention α_{ij} est fixé à 0, sinon, il est calculé comme suit :

$$\alpha_{ij}^{(l)} = \frac{\exp\left(\text{LeakyReLU}(z^{(l)} \cdot [h_i^{(l-1)}\mathbf{W}^{(l)} || h_j^{(l-1)}\mathbf{W}^{(l)}])\right)}{\sum_{k\in\mathcal{N}_i} \exp\left(\text{LeakyReLU}(z^{(l)} \cdot [h_i^{(l-1)}\mathbf{W}^{(l)} || h_k^{(l-1)}\mathbf{W}^{(l)}])\right)}, \tag{1}$$

où $||$ représente la concaténation et $\mathcal{N}_i$ est l'ensemble des voisins sortants du sommet i. Ce score est paramétré par $z^{(l)} \in \mathbb{R}^{2d_{\text{out}}}$ et $\mathbf{W}^{(l)} \in \mathbb{R}^{d_{\text{in}}\times d_{\text{out}}}$, respectivement un vecteur de poids et une transformation linéaire. L'activation de LeakyReLU est calculée avec une pente négative de 0,2.

Les représentations sont mises à jour $\mathbf{H}'^{(l)} \in \mathbb{R}^{N \times d_{\text{out}}}$ et sont calculées en fonction des scores d'attention. Pour le sommet i, il est calculé comme suit :

$$h_i'^{(l)} = \sum_{j \in \mathcal{N}_i} \alpha_{ij}^{(l)} (h_j^{(l-1)} \mathbf{W}^{(l)}). \tag{2}$$

Connexion résiduelle sur les poids d'attention. Dans la lignée des travaux récents de He *et al.* montrant que les résidus sur les poids d'attention améliorent la stabilité du Transformer (He et al., 2021), et des travaux de Lv *et al.* montrant que cela pourrait également profiter à l'apprentissage sur les graphes (Lv et al., 2021), nous remplaçons l'équation 2 par l'équation 3 ci-dessous lorsque $l > 1$ pour calculer les nouvelles représentations de sortie :

$$h_i'^{(l)} = \sum_{j \in \mathcal{N}_i} \Big(\frac{1}{2} \big)\big(\alpha_{ij}^{(l)} + \alpha_{ij}^{(l-1)}\big) h_j^{(l-1)} \mathbf{W}^{(l)} \Big). \tag{3}$$

Ainsi, nous prenons la moyenne arithmétique entre les poids d'attention calculés dans cette couche et les poids d'attention calculés dans la couche précédente.

Connexion résiduelle sur les représentations. Nous implémentons également une connexion résiduelle plus conventionnelle en sommant les représentations de sortie avec les représentations d'entrée, avant activation, pour obtenir les représentations de sortie définitives, $\mathbf{H}^{(l)} \in \mathbb{R}^{N \times d_{\text{out}}}$:

$$h_i^{(l)} = \sigma\big(h_i'^{(l)} + h_i^{(l-1)}\big), \tag{4}$$

où σ est la fonction d'activation SELU (Klambauer et al., 2017) dans nos expériences. Notons que cela nécessite d'avoir $d_{\text{in}} = d_{\text{out}}$. Si l'on souhaitait faire varier la dimension au fil des couches, on pourrait, par exemple, transformer linéairement $\mathbf{H}^{(l-1)}$ pour assurer la connexion résiduelle dans la bonne dimension. Nous ne détaillons pas cette modification car nous avons trouvé empiriquement que garder la même dimension dans les $L-1$ premières couches conduisait aux meilleurs résultats.

Interprétabilité de la dernière couche. Pour que les prédictions soient interprétables, nous adoptons une conception spécifique pour la dernière couche. Premièrement, celle-ci reçoit la matrice d'adjacence $\mathbf{A}$ sans boucles. Deuxièmement, la matrice des poids d'attention calculée dans l'avant-dernière couche est "masquée" avant d'appliquer la connexion résiduelle, sa diagonale étant fixée à 0. Troisièmement, la connexion résiduelle sur les représentations n'est pas appliquée. Ainsi, la représentation du document calculée dans la dernière couche est seulement une moyenne pondérée des représentations des phrases, tandis que les représentations des phrases sont seulement des moyennes pondérées des représentations des mots. Ainsi, les poids d'attention dans la dernière couche permettent d'identifier directement les phrases importantes et les mots importants dans celles-ci.

3.3 Estimation des paramètres

Nous extrayons la représentation du sommet document calculé dans la dernière couche, $h_0^{(L)}$, et nous la passons dans une couche dense avec activation softmax pour la classification :

$$\hat{y} = \text{softmax}(h_0^{(L)} \mathbf{C} + b) \tag{5}$$

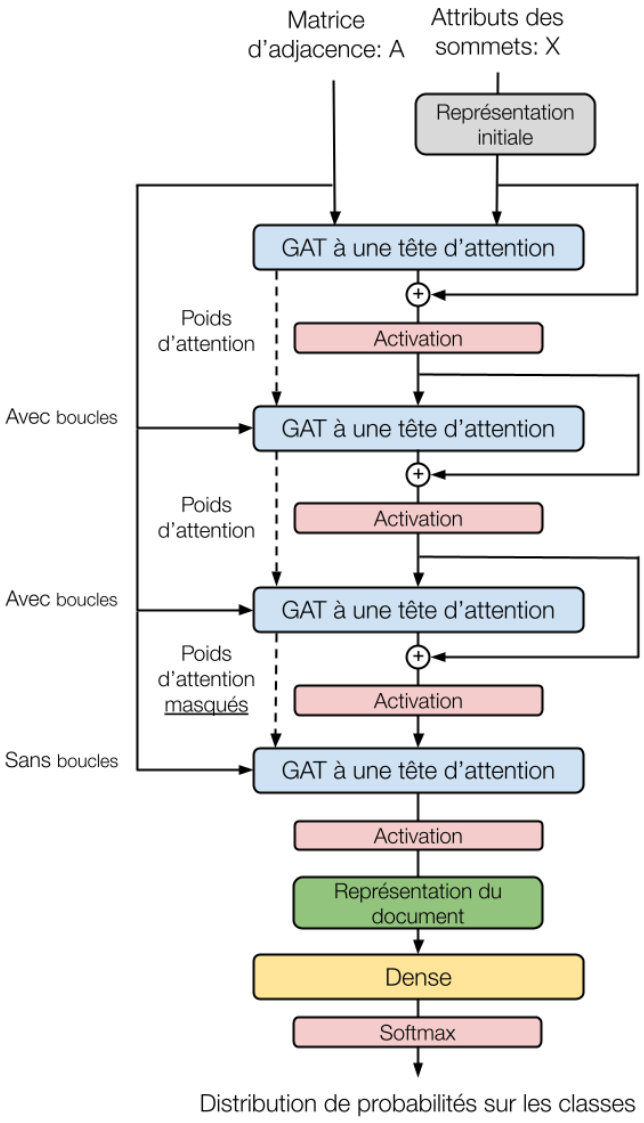


FIG. 2 – *DocGAT : Architecture du modèle.*

où $\mathbf{C} \in \mathbb{R}^{d \times c}$ et $b \in \mathbb{R}^c$ sont les paramètres du classifieur linéaire, avec c le nombre de classes. Nous régularisons l'apprentissage en ajoutant une couche Alpha-Dropout (Klambauer et al., 2017) avec une probabilité de 0,5 avant la couche de classification, et en incorporant un Dropout dans chaque couche GAT avec une probabilité de 0,25. Tous les paramètres de DocGAT sont estimés en minimisant l'entropie croisée catégorielle par descente de gradient stochastique sur des mini-batchs de documents.

TAB. 1 – *Description des jeux de données.*

	Reuters	**20-NG**	**AG-News**	**Yelp**	**Yahoo-Answers**
# documents (train)	5 584	10 741	120 000	560 000	1 400 000
# documents (test)	2 208	7 532	7 600	38 000	60 000
# classes	8	20	4	2	10

4 Expériences

4.1 Protocole et jeux de données

Nous effectuons des expériences sur cinq jeux de données bien connus, allant d'environ 5000 documents d'entraînement à plus d'un million (voir Tab. 1) :

— **Reuters** [1] : Une collection de dépêches publiées par l'agence Reuters. Nous conservons le découpage officiel entraînement/test et appliquons le même pré-traitement que Nikolentzos et al. (2020). L'objectif est de prédire le thème.

— **20-NG** [2] : Un ensemble de messages de forum sur différents sujets. Nous gardons le découpage officiel train/test. L'objectif est de prédire le sujet.

— **AG-News** [3] : Un ensemble d'articles de presse provenant de plus de 2000 sources différentes. Nous conservons le découpage officiel entraînement/test (Zhang et al., 2015). L'objectif est de prédire la catégorie.

— **Yelp**[3] : Un ensemble de critiques tirées du Yelp Challenge 2015. Nous conservons le découpage officiel entraînement/test (Zhang et al., 2015). L'objectif est de prédire la polarité.

— **Yahoo-Answers**[3] : Un ensemble de questions & réponses collectées sur Yahoo-Answers. Nous conservons le découpage officiel entraînement/test (Zhang et al., 2015). L'objectif est de prédire le sujet des questions.

Nous prélevons aléatoirement un échantillon de 10% des documents d'entraînement pour la validation, identique pour toutes les méthodes. Nous exécutons chaque méthode 10 fois sur chaque jeu de données et nous rapportons le taux de réussite moyen.

4.2 Évaluation

Nous comparons DocGAT avec 2 CNNs, 2 RNNs et 2 GNNs développés pour la classification de documents :

— **DocGAT** : 4 couches, avec $d_{\text{out}} = 300$ pour toutes les couches sauf la dernière où $d_{\text{out}} = 200$ (permis par l'absence de connexion résiduelle sur les représentations autour de la dernière couche).

1. Source : `https://www.kaggle.com/nltkdata/reuters`
2. Source : `https://scikit-learn.org/0.19/datasets/twenty_newsgroups`
3. Source : `http://goo.gl/JyCnZq`

— **CNN** [4] (Kim, 2014) : filtres de largeur 3, 4 et 5, avec 100 filtres pour chaque largeur comme dans l'article original.
— **XML-CNN**[4] (Liu et al., 2017) : filtres de largeur 2, 4 et 8, avec 100 filtres pour chaque largeur et un sous-échantillonnage en 8 zones comme Adhikari et al. (2019).
— **HAN**[4] (Yang et al., 2016) : GRU bidirectionnels pour les encodeurs phrase et document, en dimension 100, comme dans l'article original.
— **Reg-LSTM**[4] (Adhikari et al., 2019) : LSTM suivant toutes les bonnes pratiques modernes pour la classification de documents, états cachés en dimension 512, régularisé via un dropout spatial et interne aux représentations, avec des taux à 0,1 et 0,2 respectivement, comme dans l'article original.
— **MPAD** [5] (Nikolentzos et al., 2020) : 2 couches de passage de messages, chacune suivie d'un perceptron à 2 couches qui réduit la dimension à 64 comme dans l'article original.
— **MPAD-Sent**[5] (Nikolentzos et al., 2020) : Même configuration que MPAD, mais comme décrit dans (Nikolentzos et al., 2020), elle est appliquée indépendamment à chaque phrase plutôt qu'au document entier ; les représentations des phrases sont ensuite agrégées via un mécanisme d'attention.

Comme les auteurs de toutes ces méthodes, nous initialisons la couche d'embedding de ces réseaux avec les vecteurs Word2Vec, appris selon l'approche Skip-Gram avec échantillonnage négatif (Mikolov et al., 2013), [6] en dimension 300. Pour DocGAT, nous ajoutons deux vecteurs initialisés aléatoirement correspondant respectivement à la représentation initiale pour les phrases et le document. Tous les réseaux sont entraînés à l'aide de la variante ADAM de la descente de gradient stochastique en mini-batchs, avec un pas d'apprentissage initial à 0,001, sur un seul GPU NVIDIA Tesla V100-SXM2 avec 16 Go de VRAM. Le nombre d'epochs est plafonné à 100, avec un une patience de 10 sur Reuters, 20-NG et AG-News, 5 sur Yelp et 2 sur Yahoo-Answers.

4.3 Résultats principaux

Les taux de réussite moyens sont reportés dans la table 2. Parmi les 7 méthodes comparées, DocGAT se classe soit à la première soit à la deuxième place sur 4 des 5 corpus, ce qui en fait la méthode la plus cohérente. La seule exception concerne Yelp, où DocGAT (95,4%) est à 0,8 points de la meilleure méthode. Nous supposons que cette différence réside dans la capacité des méthodes comparées à accéder aux collocations de mots (les CNN opèrent sur les 2, 3 et 5-grammes ; HAN/Reg-LSTM traitent le texte de manière séquentielle ; MPAD encode les cooccurrences des mots dans le graphe), ce qui leur permet de mieux capturer des structures syntaxiques particulières comme la négation, alors que DocGAT ne peut accéder qu'aux cooccurrences au niveau de la phrase. Par ailleurs, les RNN n'ont pas été en mesure de terminer une seule passe d'entraînement en moins de 24 heures sur Yahoo, le temps de calcul maximal dont nous disposions, d'où l'absence de résultats. L'implémentation de MPAD-Sent ne peut opérer sur ce jeu de données car elle sature la RAM à notre disposition (50GB).

4. Code : `https://github.com/castorini/hedwig`
5. Code : `https://github.com/giannisnik/mpad`
6. Retrieved from : `https://code.google.com/archive/p/word2vec/`

TAB. 2 – *Moyenne des taux de réussite en test (double/simple soulignage : meilleur/2ème meilleur score).*

	Reuters	**20-NG**	**AG-News**	**Yelp**	**Yahoo-Answers**
CNN (Kim, 2014)	96.9	84.6	92.3	95.7	73.0
XML-CNN (Liu et al., 2017)	97.0	83.2	92.3	96.1	74.0
HAN (Yang et al., 2016)	96.8	82.1	92.6	96.2	*DNF*
Reg-LSTM (Adhikari et al., 2019)	97.1	83.5	92.9	96.0	*DNF*
MPAD (Nikolentzos et al., 2020)	97.1	79.3	92.3	95.4	70.7
MPAD-Sent (Nikolentzos et al., 2020)	96.5	75.0	91.7	94.1	*OOM*
DocGAT (Ce papier)	97.4	84.2	92.8	95.4	74.1

4.4 Analyse détaillée

TAB. 3 – *Nombre moyen d'epochs pour atteindre le meilleur modèle en validation.*

	Reuters	**20-NG**	**AG-News**	**Yelp**	**Yahoo-Answers**
CNN (Kim, 2014)	14.2	26.6	2.0	2.2	2.0
XML-CNN (Liu et al., 2017)	12.4	15.6	1.6	1.6	1.0
HAN (Yang et al., 2016)	8.2	7.8	2.0	2.0	*NA*
Reg-LSTM (Adhikari et al., 2019)	19.2	21.6	2.0	2.0	*NA*
MPAD (Nikolentzos et al., 2020)	53.3	72.0	31.4	7.6	4.2
MPAD-Sent (Nikolentzos et al., 2020)	64.8	82.0	28.3	7.0	*NA*
DocGAT (this paper)	15.3	41.8	17.0	9.6	6.6

Vitesse de convergence et temps d'entraînement. La table. 3 indique le nombre moyen d'epochs nécessaires pour atteindre le modèle le plus performant sur les données de validation. Il apparaît que DocGAT converge beaucoup plus rapidement que la méthode la plus semblable, MPAD, sur les trois plus petits jeux de données (d'environ 5k documents à 100k documents). Cependant, DocGAT semble converger plus lentement sur les plus grands corpus, en particulier lorsqu'il est comparé aux méthodes CNN et RNN. Nous supposons que cela est dû au fait que DocGAT doit apprendre des représentations pour les phrases, en plus des représentations des mots et des documents. Notons que l'entraînement de DocGAT pour une epoch prend 4 secondes, 10 secondes, 14 secondes, 395 secondes et 636 secondes sur Reuters, 20-NG, AG-News, Yelp et Yahoo-Answers, respectivement. A titre d'exemple, réaliser une epoch requiert deux fois moins de temps que pour MPAD (1365 secondes) sur Yahoo. Une passe d'entraînement prend environ 1,5 heure pour DocGAT contre 2,3 heures pour MPAD et plus de 24 heures

pour les méthodes RNN, tandis que les méthodes CNN nécessitent environ 0,5 heure.

Nombre de paramètres. Sans compter la couche de représentation initiale qui est commune à tous les modèles, DocGAT a avec 333 000 paramètres, ce qui le rend d'une taille comparable au CNN le plus simple. XML-CNN (en raison des cartes d'attributs plus grandes et de la couche de réduction de dimension) et Reg-LSTM (en raison de la complexité de sa cellule récurrente) ont chacun environ 1 660 000 paramètres.

Interprétabilité. Les tables 4 et 5 présentent deux documents tirés de de l'échantillon de test du corpus AG-News. La colonne de gauche liste les poids d'attention entre le sommet document et les sommets phrases. La colonne de droite montre les phrases, avec les mots surlignés d'après les poids d'attention entre les sommets phrases et les sommets mots. Le premier document est principalement une citation qui, prise hors contexte, serait difficile à classer dans la catégorie "Sports". Pourtant, DocGAT concentre son attention sur la phrase d'introduction, en sélectionnant l'expression "Athens Olympis" (jeux olympiques d'Athènes), en sautant la citation pour prédire la bonne catégorie. Le deuxième document est un article à propos de navigateurs Web. Encore une fois, on peut voir que DocGAT est capable de sauter la troisième phrase, qui ne fournit aucun indice pour prédire la catégorie "Science/Technologie", et qu'il se concentre sur les mots pertinents dans les autres phrases.

TAB. 4 – *Poids d'attention dans la dernière couche de DocGAT. Prédiction : "Sports".*

Poids	Phrase
0.651	Notable quotes Tuesday at the Athens Olympics .
0.068	“It hurt like hell.
0.078	I could see (Thorpe) coming up.
0.115	But when I was breathing, I saw my team going crazy – and that really kept me going.
0.091	” ...

5 Conclusion

Nous avons présenté DocGAT, réseau de neurones pour classifier des documents encodés comme des graphes. Les expériences ont montré la pertinence de cette approche puisque d'une part, DocGAT obtient des résultats comparables ou meilleurs que les méthodes de l'état de l'art, tout en étant efficace à entraîner et à exécuter. D'autre part, les prédictions faites par DocGAT sont naturellement interprétables, ce qui permet de mettre en évidence les parties importantes des documents (phrases/mots). Nous pensons que cela fait de DocGAT une solution intéressante dans de nombreux scénarios applicatifs et nous mettons notre code à disposition pour favoriser son adoption par les praticiens et les chercheurs.

TAB. 5 – *Poids d'attention dans la dernière couche de DocGAT. Prediction : "Sci/Tech".*

Poids	Phrase
0.191	10 features for a perfect browser .
0.263	There are some great browsers out there.
0.061	But they all seem to have some slight niggles, different for each, that make it hard for me to kick back and enjoy them.
0.264	While there are some projects out there to make browsers more useful for some specialised purposes or by bolting on handy extensions, wouldn't it be great if these people could come up with a standardised set of nice features like these ?
0.221	A lot of browsers may support one or two, but I'll bet none have them all.

Références

Adhikari, A., A. Ram, R. Tang, et J. Lin (2019). Rethinking complex neural network architectures for document classification. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL, pp. 4046–4051.

Chen, D., Y. Lin, W. Li, P. Li, J. Zhou, et X. Sun (2020). Measuring and relieving the over-smoothing problem for graph neural networks from the topological view. In *Proceedings of the AAAI Conference on Artificial Intelligence*, AAAI, pp. 3438–3445.

Cho, K., B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, et Y. Bengio (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP.

Devlin, J., M.-W. Chang, K. Lee, et K. Toutanova (2019). BERT : pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL, pp. 4171–4186.

He, R., A. Ravula, B. Kanagal, et J. Ainslie (2021). Realformer : Transformer likes residual attention. In *Proceedings of the Joint Conference of the Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing*, ACL-IJCNLP.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP, pp. 1746–1751.

Kipf, T. N. et M. Welling (2017). Semi-Supervised Classification with Graph Convolutional Networks. In *Proceedings of the International Conference on Learning Representations*, ICLR.

Klambauer, G., T. Unterthiner, A. Mayr, et S. Hochreiter (2017). Self-normalizing neural networks. In *Advances in neural information processing systems*, NeurIPS, pp. 971–980.

Liu, J., W.-C. Chang, Y. Wu, et Y. Yang (2017). Deep learning for extreme multi-label text classification. In *Proceedings of the International ACM Conference on Research and Development in Information Retrieval*, SIGIR, pp. 115–124.

Lv, Q., M. Ding, Q. Liu, Y. Chen, W. Feng, S. He, C. Zhou, J. Jiang, Y. Dong, et J. Tang (2021). Are we really making much progress ? revisiting, benchmarking and refining heterogeneous graph neural networks. In *Proceedings of the ACM Conference on Knowledge Discovery amp ; Data Mining*, KDD, pp. 1150–1160.

Mikolov, T., I. Sutskever, K. Chen, G. S. Corrado, et J. Dean (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, NeurIPS, pp. 3111–3119.

Nikolentzos, G., A. Tixier, et M. Vazirgiannis (2020). Message passing attention networks for document understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, AAAI, pp. 8544–8551.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, et I. Polosukhin (2017). Attention is all you need. In *Advances in Neural Information Processing Systems*, NeurIPS, pp. 5998–6008.

Veličković, P., G. Cucurull, A. Casanova, A. Romero, P. Liò, et Y. Bengio (2018). Graph Attention Networks. ICLR.

Yang, Z., D. Yang, C. Dyer, X. He, A. Smola, et E. Hovy (2016). Hierarchical attention networks for document classification. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL, pp. 1480–1489.

Yao, L., C. Mao, et Y. Luo (2019). Graph convolutional networks for text classification.

Zhang, X., J. Zhao, et Y. LeCun (2015). Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, NeurIPS, pp. 649–657.

Summary

In this paper, we propose to encode documents as directed acyclic graphs that preserve the sentence-level structure. We introduce DocGAT, a novel, attention-based, graph neural network for document classification that learns word, sentence and document representations in a hierarchical manner. Experiments conducted on various datasets show that DocGAT achieves state-of-the-art results. We also show that, having comparatively few parameters and being purely a feed-forward network, it is practically advantageous since it can easily be trained on large datasets or applied to long documents. Lastly, we shed light on the interpretability of its predictions, thanks to the attention mechanism that selects important sentences and important words in them, a desirable feature in many use-cases.

Analyse de Données Aberrantes pour le Texte : Taxonomie et Étude Expérimentale

Jeremie Pantin*, Marie-Jeanne Lesot*, Christophe Marsala*

*Sorbonne Université, CNRS, LIP6, Paris F-75005, France
Prenom.Nom@lip6.fr

Résumé. La tâche de détection des données aberrantes consiste à étudier un ensemble de données afin d'estimer les individus singuliers. Cette tâche est souvent effectuée pour supprimer le bruit ou les données anormales lors de l'étape de nettoyage. Si la littérature est riche pour de nombreux types de données, les contributions pour le texte sont plus rares. Dans cette étude nous nous intéressons aux données aberrantes textuelles. L'apparition de ces aberrations peut survenir de manière indépendante ou contextuelle pour un sujet donné. Nous introduisons ainsi une taxonomie pour les identifier, et sur la base de celle-ci, nous proposons GenTO qui est une approche de génération d'observations aberrantes. Son utilisation permet d'identifier les forces et faiblesses d'une approche face à différentes aberrations. Une étude comparative des approches de l'état de l'art est finalement effectuée en utilisant GenTO. Les résultats montrent que les travaux récents ne sont pas significativement plus performants.

1 Introduction

La tâche de Détection des Données Aberrantes (DDA) concerne de nombreux domaines et applications. Cette tâche est formellement présentée dans de nombreux travaux (Hawkins, 1980; Hodge et Austin, 2004; Zhang, 2013; Aggarwal, 2017; Ruff et al., 2021). Auparavant, elle était réalisée dans le domaine des statistiques (Beckman et Cook, 1983) avec des données statiques et des méthodes paramétriques. Les progrès récents en fouille de données ont fait évolué la tâche, ainsi que son formalisme, pour la rendre applicable à un plus grand nombre de scénarios. Pour Aggarwal (2017), *une donnée aberrante est une donnée qui est significativement différente des autres données*. Les méthodes effectuant la DDA visent souvent à calculer un score de "normalité" d'une instance. Bien que ces approches produisent des résultats significatifs avec de nombreux types de données, leur application au texte manque de clarté.

Dans les travaux mentionnés, il existe un nombre suffisant de contributions qui offrent un aperçu de la tâche de DDA. Toutefois, elles étudient rarement le scénario dans lequel les données sont purement textuelles. L'intérêt pour la Détection des Données Aberrantes Textuelles (DDAT) a été récemment suscité par les réseaux neuronaux (Gorokhov et al., 2017; Ruff et al., 2019; Lai et al., 2020) et les méthodes de décomposition de matrices (Allan et al., 2008; Kannan et al., 2017). Bien que ces contributions

Sujet : Sport/Tennis. Documents normaux :
x_1 : Naomi Osaka came to Flushing Meadows to entertain and did not disappoint on Monday, overcoming a slow start to beat Czech Marie Bouzkova 6-4 6-1 and get her U.S. Open title defence under way in front of a roaring capacity crowd.
x_2 : World number one Novak Djokovic had to work hard for a three-set victory over Hungarian Marton Fucsovics at the Paris Masters on Tuesday in his first match since losing the U.S. Open final in September.

Sujet : Politique. Document aberrant :
x_3 : The Olympic Games have started and yet, all countries are [...] Japanese tennis star Naomi Osaka on Friday lit the Olympic cauldron to mark the formal start of Tokyo 2020, in an opening ceremony shorn of glitz and overshadowed by a pandemic but celebrated as a moment of global hope.

FIG. 1 – *Exemple de trois articles de Reuters et Eurosport qui porte sur le tennis et la politique. Les documents normaux traitent des résultats des joueurs de tennis tandis que le sujet aberrant porte sur la cérémonie d'ouverture des Jeux olympiques.*

se concentrent sur l'exécution de la tâche de DDA, ici appelée détection des anomalies, elles n'étudient pas la nature des aberrations textuelles. Une observation récurrente concernant ces contributions est qu'elles ne se comparent pas aux travaux récents. De plus, elles n'étudient pas les aberrations construites avec leur protocole expérimental.

La Figure 1 présente un exemple de DDAT avec trois articles de presse. Alors que le sujet principal est Sport/Tennis, nous observons qu'un article Politique apparaît à tort dans le corpus. Un tel scénario est courant, et la tâche de détection de spam présente des similitudes avec ce problème. La recherche bibliographique de DDAT dans la littérature est déroutante car plusieurs tâches telles que la détection d'anomalies, la détection d'aberrations et la détection de nouveautés sont souvent utilisées de façon interchangeable. Au vu de l'absence d'ensembles de données dédiés à la DDAT, la proposition d'un algorithme de génération de données aberrantes apparait nécessaire.

1.1 Contributions

Approcher la tâche de DDAT est difficile pour plusieurs raisons : le faible nombre d'études approfondies avec des approches de l'état de l'art, la définition de nouveaux problèmes pour une tâche similaire (détection de spam, détection de plagiat, ...), le manque de corpus dédiés et les protocoles expérimentaux divergents dans la littérature.

Étude approfondie des travaux correspondants. Notre première contribution est l'étude de la DDA pour les données textuelles avec une analyse approfondie des méthodes connexes. Alors que les approches de DDAT sont rares dans la littérature, nous étudions les méthodes populaires de détection de nouveautés, de détection d'anomalies, de détection de plagiats, de détection de spams et de classification à classe unique (CCU).

Taxonomie et génération d'aberrations. L'absence de jeux de données dédiés a conduit les approches de la littérature à préparer spécifiquement les corpus de classification de textes. Nous proposons une taxonomie qui identifie les différents types d'aberrations pour le texte. Nous formalisons celle-ci à partir des travaux de référence de la tâche de DDA. L'avantage de cette taxonomie est sa connexion avec des types d'aberrations bien connus de la littérature. À partir de celle-ci, nous proposons une approche qui génère des aberrations ponctuelles/indépendantes et des aberrations contextuelles/conditionnelles : GenTO. Notre algorithme peut être facilement appliqué à n'importe quel corpus possédant une hiérarchie de sujets.

Comparatif des configurations expérimentales. Dans la littérature, le type des aberrations construites n'est pas clairement identifié lors de l'étape d'évaluation, conduisant à une lecture incomplète des résultats. La proposition de GenTO est fortement motivée pour cette raison. En complément, nous définissons le taux de contamination d'un corpus (poids du nombre d'aberrations dans un corpus). Dans divers travaux, ce paramètre peut varier afin de donner une idée de la robustesse d'une approche par rapport aux données du monde réel. Nous proposons une étude comparative, basée sur GenTO, qui met en évidence ces caractéristiques.

1.2 Structure de l'article

La Section 2 passe en revue les travaux connexes et souligne les principales caractéristiques de la tâche. La Section 3 présente notre taxonomie et son application au texte, ainsi que l'algorithme de génération d'aberrations : GenTO. Une étude comparative est présentée dans la Section 4 sur tous les types d'aberrations identifiés. Enfin, nous concluons avec la Section 5 et abordons les perspectives de ce travail.

2 Préliminaires

Dans cette section, nous présentons les connaissances principales associées à l'analyse des données aberrantes, ainsi que les travaux connexes de la tâche de DDAT. Nous soulignons également les différences entre les termes anomalie et nouveauté qui, dans la pratique, sont souvent employés de manière interchangeable avec aberration.

2.1 Qu'est-ce qu'une aberration textuelle ?

Bien que la définition d'une observation aberrante soit claire dans le cadre des statistiques, les progrès récents avec les données plus complexes (image, texte, ...) impliquent de fournir de nouvelles définitions. Des termes tels que *anomalie* et *nouveauté* renvoient à des observations similaires à celles des données aberrantes. Concrètement, une anomalie est une observation d'intérêt, une nouveauté est un type d'observation qui nécessite la mise à jour des modèles et une aberration est une instance fréquemment considérée comme une donnée qui devrait être supprimée (Ruff et al., 2021).

Le texte possède des propriétés telles que la rareté et la grande dimensionnalité, la rendant difficile à traiter. En ce qui concerne la définition de Aggarwal (2017),

nous proposons qu'une aberration textuelle est *un document qui est significativement différent des autres documents.* Cette définition soulève un problème lié au texte qui est le niveau d'application : syntaxique ou sémantique. Nous nous concentrons sur cette dernière, et proposons la définition suivante : *"une aberration textuelle est un document associé à un sujet qui est significativement différent des sujets des autres documents."*

2.2 Détection de données aberrantes dans le texte

Dans cette partie, nous présentons quatre types d'approches qui effectuent la DDA. La plupart de ces contributions sont réalisées avec des données textuelles, mais pas nécessairement pour la tâche de DDAT. Premièrement, nous étudions les approches basées sur la distance qui calculent un score à partir de l'éloignement entre les observations. Le deuxième type d'approche est basé sur la densité, où les observations proches se concentrent autour d'une même partie de l'espace. Les approches SVM basées sur des noyaux séparent quant à eux les données via un hyperplan. Enfin, nous terminons par les approches par réseaux de neurones artificiels.

Approches basées sur la distance. Plusieurs travaux sont fondés sur l'utilisation d'approches par distance, tel que KNN : (Ramaswamy et al., 2000; Koppel et Seidman, 2013; Kannan et al., 2017). Koppel et Seidman (2013) propose l'utilisation de KNN avec une métrique de similarité de second ordre afin d'effectuer la détection du plagiat dans des textes. Wang et al. (2018), quant à eux, proposent trois modèles qui calculent la distance d'un document à un autre, d'un document à un cluster, et d'un document au reste du corpus. Les approches basées sur la distance sont très sensibles à la métrique de distance et peinent avec les matrices éparses (Aggarwal, 2017; Kannan et al., 2017).

Approches basées sur la densité. Nous pouvons observé que les approches basées sur la densité peuvent traiter le problème de DDA. L'hypothèse naturelle qui suit notre définition d'aberration textuelle est : *une aberration est situé en dehors de la zone de densité formée par les données normales.* Srivastava et Zane-Ulman (2005) proposent une approche basée sur le modèle de mélange gaussien (GMM) et l'Analyse en Composantes Principales (ACP) dans le but de réduire la matrice sac de mots (BOW). Des travaux tels que Tran Manh Thang et Juntae Kim (2011) utilisent DBSCAN pour détecter des anomalies via une optimisation des paramètres et de mécanismes dédiés. Local Outlier Factor (LOF) (Breunig et al., 2000) est un autre algorithme populaire qui estime le taux d'aberration d'une observation en fonction de sa proximité avec une densité locale. Walkowiak et al. (2020) l'utilise pour distinguer les documents incorrectement associés à des clusters. Récemment, Fouché et al. (2020) proposent également une approche pour l'extraction de données aberrantes dans des répertoires de documents. Leur méthode, kj-NN, utilise la similarité sémantique et estime au niveau des phrases la pertinence d'un document. Il est intéressant de noter qu'ils proposent également une taxonomie dédiée possédant deux types de d'aberrations : Type-O (documents hors distribution) et Type-M (documents mal classés).

Approches à noyaux. OCSVM (One-Class Support Vector Machine) est l'une des approches les plus populaires pour la détection d'aberrations et d'anomalies. Manevitz et Yousef (2001) utilisent OCSVM pour des données textuelles où ils étudient son efficacité sur quatre représentations. Ils effectuent leurs expérimentations avec des noyaux linéaires, polynomiaux, radiaux et sigmoïdes. Également, Shravan Kumar et Ravi (2017) ont récemment proposé d'utiliser OCSVM couplé à une représentation sémantique du texte avec LSA (Latent Semantic Analysis). Comparativement, ils ajoutent la réduction de dimension au travail de Manevitz et Yousef (2001).

Approches basées sur des réseaux de neurones artificiels. Les progrès récents des réseaux neuronaux artificiels ont eu un impact positif sur la tâche de DDAT. Deux types d'approches peuvent être observés : la reconstruction et la classification à classe unique. Mei et al. (2018) proposent une approche de détection de nouveautés qui utilise des auto-encodeurs. Bien que les résultats montrent que la méthode manque de stabilité au fil des expériences, elle parvient à éviter correctement les faux positifs. L'approche proposée par Lai et al. (2020) utilise une couche de récupération de sous-espace robuste (RSR). Cette couche cherche à extraire des sous-espaces significatifs où les aberrations seraient difficiles à localiser. Cette technique facilite l'étape de reconstruction du décodeur. Gorokhov et al. (2017) proposent une méthode de détection d'anomalies avec un réseau neuronal convolutif (CNN). On observe toutefois que l'efficacité des modèles de langage récents n'est pas pleinement exploitée dans la tâche. Ruff et al. (2019) proposent ainsi de les utiliser avec un Context Vector Data Description (CVDD).

3 Taxonomie des aberrations textuelles

Dans cette section, nous présentons deux des contributions de nos travaux : une taxonomie des aberrations textuelles et un algorithme de génération d'aberrations : GenTO. Une introduction des notations est effectuée au préalable.

3.1 Notations

Il existe plusieurs types de documents, et il est courant d'observer différentes sources de textes dans la littérature. Étant donné un corpus, *une aberration textuelle est un document qui est significativement différent des autres documents.* En suivant cette définition, nous pouvons l'appliquer aux articles de presse, par exemple, où une *aberration textuelle est un document associé à un sujet qui est significativement différent des sujets des autres documents.* Soit une séquence de M mots $x_1, ..., x_M$ avec un vocabulaire fixe $\mathcal{V}$ de taille $|\mathcal{V}| = V$. Chaque mot est représenté par $x_i \in [0, N]$ pour $i \in \{1, ..., M\}$ lorsque le modèle de représentation est un sac de mots (BOW). Ainsi, $\mathrm{x} \in \mathrm{X}$ où X est un corpus de N documents.

La DDA peut être associée à la classification à classe unique (CCU) où la plupart du temps, la sortie est un score tel que $s : \mathrm{X} \mapsto \mathcal{R}$. Les sorties de la classification binaire et de la CCU sont similaires, ainsi -1 désigne une aberration et $+1$ une donnée normale. Un classifieur de DDA vise à trouver le nombre optimal de données divergentes de X tout en minimisant autant que possible les faux positifs.

3.2 Taxonomie

Plusieurs définitions d'une aberration ont été proposées dans la littérature : aberration indépendante/ponctuelle, aberration conditionnelle/contextuelle et aberration collective/groupée. La nature du texte implique que lorsque plusieurs données aberrantes apparaissent, elles sont à la fois contextuelles et collectives.

Considérant que les documents contiennent des informations d'un ou plusieurs sujets, nous supposons qu'il existe une hiérarchie des sujets du corpus. Ce type de hiérarchie est nécessaire afin d'étudier les aberrations contextuelles sans ensemble de données dédié. Dans la tâche de classification, les approches hiérarchiques ne sont pas rares (Toutanova et al., 2001) et les structures à deux ou trois niveaux sont courantes.

Les sujets (catégories) sont organisés sous forme d'une structure arborescente où ils peuvent être associés, tout au plus, un parent. Nous ajoutons également la contrainte qu'une catégorie est unique, et qu'elle ne peut pas apparaître deux fois comme enfant ou parent. Dans cette configuration, les documents ne peuvent apparaître qu'au niveau des feuilles, et ne peuvent donc pas être parent d'un sujet. Soit H une hiérarchie qui a $l \in L$ niveaux et $y \in Y$ sujets. Chaque élément $h \in H$ admet un ensemble d'enfants qui mène au niveau $l+1$ qui est soit un sujet, une feuille ou $\{\emptyset\}$.

Les aberrations indépendantes sont des observations dont le sujet ne partage aucune relation avec un autre sujet. Précisément, le sujet aberrant et le sujet définie comme normal ont un parent différent dans la structure hiérarchique des catégories. On note ζ la catégorie normale et son sous-ensemble correspondant $X_\zeta \subseteq X$. Nous définissons O le sous-ensemble de toutes les observations aberrantes, tel que $O \subset X$. On a :

$$O = X \backslash X_\zeta \tag{1}$$

En ce qui concerne O, nous pouvons distinguer deux contraintes différentes. Premièrement, une observation x_i est considérée aberrante si son sujet parent est différent du sujet parent normal, tel que :

$$O_{indépendante} = \{\text{parent}(y) \neq \text{parent}(\zeta) | \forall (o, y) \in O \times Y\} \tag{2}$$

La deuxième contrainte correspond aux documents qui ne se trouvent pas dans X_ζ mais qui partagent le même sujet parent que ζ. Ces observations sont identifiées comme un autre type d'aberration : aberrations contextuelles. Ainsi :

$$O_{contextuelle} = \{\text{parent}(y) = \text{parent}(\zeta) | \forall (o, y) \in O \times Y, O \backslash O_{point}\} \tag{3}$$

3.3 GenTO : Generation d'aberrations textuelles

À partir de l'Équation 2 et de l'Équation 3, nous proposons GenTO pour générer des aberrations textuelles. Dans la section précédente, nous avons défini deux types de d'aberrations : indépendantes et contextuelles. On note également que GenTO peut être appliqué pour chacun d'entre eux. L'Algorithme 1 décrit GenTO et suit l'Équation 2 pour la génération d'aberrations indépendantes et l'Équation 3 pour la génération d'aberrations contextuelles.

Algorithm 1 GenTO : Generation of Textual Outliers

Require: Sujet normal ζ, corpus X, taille l, taux de contamination ν, booléen *est_contextuelle*
Ensure: $0 < l \leq N$
  $c \leftarrow l\nu$
  $i \leftarrow 0$
  Initialisation de la matrice vide Z ▷ Matrice d'échantillonnage
  $\mathrm{O} \leftarrow \{\mathrm{x}_j \times \mathrm{y}_j \in \mathrm{X} \times \mathrm{Y} | \forall j \in [0, N], \mathrm{y}_j \neq \zeta\}$ ▷ Matrice de données aberrantes
  $\mathrm{X}_\zeta \leftarrow \{\mathrm{X} \backslash \mathrm{O}\}$ ▷ Matrice de données normales
  while $|\mathrm{Z}| < c$ **do**
    if *est_contextuelle* **then**
      Ajouter$(\mathrm{x}_i, \mathrm{y}_i)$ dans Z sachant que Parent(y_i) = Parent(ζ)
    else
      Append$(\mathrm{x}_i, \mathrm{y}_i)$ dans Z sachant que Parent$(\mathrm{y}_i) \neq$ Parent(ζ)
    end if
    $i \leftarrow i + 1$
  end while
  Remplir Z avec X_ζ jusqu'à ce que $|\mathrm{Z}| = l$
  return Mélanger(Z)

4 Expérimentations

Dans cette section, nous présentons les expériences menées sur la DDAT avec des aberrations générées via GenTO. Nous décrivons également les jeux de données et la manière dont ils sont utilisés, de l'étape de pré-traitement à l'étape de préparation. Les mesures d'évaluation utilisant Area Under the Receiver Operating Characteristics Curve (AUROC) et Area Under the Precision-Recall Curve (AUPRC) sont détaillées dans la deuxième partie. Les approches de référence sont introduites dans le troisième point, en plus de leur configuration. Enfin, nous présentons les résultats de notre expérimentation.

Données. Comme indiqué précédemment, les ensembles de données dédiés à la DDAT sont inexistants et la préparation de corpus compatibles est une étape importante. Même si il existe des groupes de jeux de données dédiés à la DDA (ODDS et UCI), ils fournissent principalement des données multidimensionnelles, des séries temporelles et des images. Dans ce contexte, les corpus textuels sont peu représentés. Toutefois, des tâches telles que la détection de spam et le détection de plagiat disposent d'un riche ensemble de corpus. Des travaux récents (Lai et al., 2020; Ruff et al., 2019; Kannan et al., 2017; Mahapatra et al., 2012) utilisent des ensembles de données de classification tels que Reuters-21578 [1] et 20 Newsgroups [2] dans le but d'évaluer leurs approches.

Pour 20 Newsgroups, nous transformons les documents en une matrice BOW après avoir appliqué la suppression des mots vides et la mise en minuscule du texte brut.

1. `http://www.daviddlewis.com/resources/testcollections/reuters21578/`
2. `http://qwone.com/~jason/20Newsgroups/`

Les termes qui apparaissent moins de trois fois ne sont pas pris en compte et le modèle BOW est entraîné sur le sous-ensemble d'entraînement. Nous préparons ensuite chaque classe avec GenTO pour générer des aberrations ponctuelles et contextuelles avec $\nu \in \{0.05, 0.10\}$. La motivation derrière ces valeurs est d'évaluer les approches contre plusieurs niveaux de contamination. Nous séparons les sous-thèmes entre sept thèmes principaux : informatique, vente, moteurs, politique, religion, science et sports.

Reuters-21578 est un corpus contenant des documents associés à plusieurs sujets. Nous supprimons tous ces documents afin de ne garder que ceux qui possèdent un unique sujet. Ensuite, le corpus est préparé de la même manière que 20 Newsgroups avec un modèle BOW entraîné sur le sous-ensemble d'entraînement. À partir de Toutanova et al. (2001), nous réorganisons les sujets pour obtenir une hiérarchie. Ainsi, quatre thèmes parents sont créés : matières premières, finance, métaux et énergie. Nous appliquons GenTO aux huit sujets qui ont le plus grand nombre de documents dans Reuters-21578.

Évaluation. La DDA est une tâche qui implique des données fortement déséquilibrées où les données normales (vrais négatifs) sont prédominantes. Par conséquent, la précision moyenne est souvent utilisée. Les différentes représentations sont comparées à l'aide de AUROC et de AUPRC. Ces mesures sont dérivées de la matrice de confusion et sont toutes deux souvent utilisées pour la tâche de détection des aberrations. La courbe ROC affiche le taux de vrais positifs (TPR) en fonction du taux de faux positifs (FPR) pour plusieurs seuils. L'augmentation ou la diminution de ce seuil influence les vrais positifs par rapport aux faux positifs. Elle aide à choisir le meilleur seuil pour un classifieur.

L'AUROC évalue les performances des méthodes de référence en fonction du score de l'aberration. Toutes les expériences ont été menées sur dix essais où l'AUROC et l'AUPRC sont calculées. L'étape d'évaluation est ensuite effectuée sur le sous-ensemble d'entraînement. Effectuer cette étape sur le sous-ensemble de test est similaire à évaluer la robustesse d'un modèle à la nouveauté. Dans les travaux futurs, l'étude de cette évaluation sera effectuée en raison de l'importance de l'information liée : est-ce qu'une approche est robuste contre les nouveaux types/générations de documents.

Méthodes de référence. Notre étape de préparation des données est similaire à Kannan et al. (2017); Lai et al. (2020); Ruff et al. (2019); Fouché et al. (2020), nous permettant de facilement conduire les expérimentations avec leur implémentation. Nous suivons de près leurs recommandations afin d'être le plus équitable possible. Pour TONMF (Kannan et al., 2017), nous avons fixé $\alpha = 15$ et $k = 30$ pour lesquels les résultats obtenus étaient les meilleurs. RSRAE (Lai et al., 2020) est configuré avec les paramètres décrits dans leurs travaux : la dimension latente est fixée à 10, le taux d'apprentissage à 0.00025 et le nombre d'époques à 200. Avec CVDD (Ruff et al., 2019), la configuration proposée par les auteurs est également rigoureusement respectée.

Pour la mise en œuvre de LOF, Isolation Forest (IForest), OCSVM (Manevitz et Yousef, 2001), KNN (Ramaswamy et al., 2000) et PCA (Shyu et al., 2003) nous utilisons l'outil PyOD (Zhao et al., 2019). La métrique de distance pour LOF est *cosine* et le nombre de voisins est fixé à 20, il en va de même pour KNN. Nous avons obtenu

		$\nu = 0.05$				$\nu = 0.10$			
		Indépendante		Contextuelle		Indépendante		Contextuelle	
Modèle		AUPRC	AUROC	AUPRC	AUROC	AUPRC	AUROC	AUPRC	AUROC
LOF	20 Newsgroups	0.139	0.762	0.091	0.652	**0.217**	0.745	**0.153**	**0.637**
IForest		0.068	0.518	0.062	0.516	0.125	0.527	0.116	0.513
OCSVM		0.104	0.609	0.073	0.591	0.130	0.641	0.129	0.617
KNN		0.064	0.489	0.063	0.505	0.118	0.505	0.114	0.500
PCC		0.121	0.663	0.086	0.607	0.203	0.665	0.148	0.594
LSA_{l2}		0.105	0.614	0.079	0.592	0.172	0.700	0.148	0.687
NMF_{l2}		0.119	0.651	0.091	0.619	0.184	0.716	**0.150**	0.621
TONMF		0.072	0.548	0.073	0.514	0.147	0.619	0.127	0.547
CVDD		**0.176**	**0.783**	**0.107**	**0.702**	0.208	**0.822**	**0.150**	0.629
RSRAE		0.110	0.729	0.078	0.622	0.196	0.725	0.145	0.621
LOF	Reuters-21578	0.348	0.870	**0.302**	0.795	0.380	0.812	0.316	**0.751**
IForest		0.136	0.648	0.097	0.564	0.197	0.642	0.136	0.534
OCSVM		0.235	0.750	0.130	0.641	0.300	0.816	0.223	0.747
KNN		0.117	0.618	0.094	0.549	0.177	0.621	0.134	0.526
PCC		0.222	0.756	0.143	0.674	0.297	0.746	0.188	0.635
LSA_{l2}		0.229	0.756	0.155	0.719	0.268	0.709	0.245	0.610
NMF_{l2}		0.308	0.792	0.168	0.765	0.294	0.787	0.260	0.688
TONMF		0.109	0.632	0.102	0.587	0.140	0.647	0.148	0.693
CVDD		0.377	0.889	0.291	0.812	0.416	0.799	0.312	0.740
RSRAE		**0.390**	**0.898**	0.295	**0.825**	**0.449**	**0.866**	**0.346**	**0.752**

TAB. 1 – *Résultats des modèles de l'état de l'art pour les aberrations indépendantes et contextuelles, avec deux taux de contamination. Les mesures d'évaluation utilisées sont AUPRC et AUROC.*

de meilleurs résultats avec le nombre de voisins $n \in [20, 30]$, mais 20 semble être la valeur possédant le plus de stabilité. Isolation Forest (IForest) (Liu et al., 2008) est une approche basée sur le parcours d'arbres. Nous prenons la configuration par défaut proposée par les auteurs afin de calculer l'AUROC et l'AUPRC. Basé sur l'ACP, PCC est un classifieur de composantes principales robuste qui a été évalué à l'origine sur le jeu de données KDD'99. La méthode calcule un hyperplan de faible dimension construit par k vecteurs propres. Nous conservons toutes les composantes dans notre configuration. Également, nous utilisons le noyau polynomial pour OCSVM et définissons : degré = 2.

Nous proposons d'utiliser deux méthodes d'approximation de rang faible : analyse sémantique latente (LSA) et factorisation matricielle non négative (NMF). Nous prenons le rang le plus performant pour r en variant $r \in [20, 100]$. Afin d'aligner l'utilisation de NMF sur celle de TONMF, nous définissons tolerance $= 1e-6$. Également, β-loss est calculé avec la norme de Frobenius. Pour ces deux méthodes, nous calculons la norme l_2 sur la matrice réduite.

Résultats. Les résultats du tableau 1 montrent les performances des méthodes de référence avec AUROC et AUPRC, pour les aberrations indépendantes et contextuelles. Pour 20 Newsgroups, l'approche KNN est l'une des méthodes les moins performantes, en particulier avec $\nu = 0.05$. LOF avec la distance *cosine* et CVDD sont les méthodes les plus performantes tandis que RSRAE réussit à obtenir de bons résultats. Pour Reuters-21578, RSRAE est la méthode se démarquant le plus. LOF et CVDD ont également

de bons résultats tandis que TONMF, KNN et IForest rencontrent des difficultés, en particulier avec les aberrations contextuelles.

D'après les résultats, nous observons que les aberrations indépendantes enregistrent de meilleures performances que les aberrations contextuelles. De même, les approches sont plus performantes avec un taux de contamination plus élevé. Pour les réseaux de neurones artificiels, nous pouvons voir qu'ils sont plus robustes en fonction du taux de contamination. LOF semble également être beaucoup plus robuste que les autres approches basées sur la densité. Les approches à noyau et à distance peinent à obtenir des résultats similaires. Également, les méthodes d'approximation de rang faible obtiennent des résultats compétitifs dans la plupart des cas.

5 Conclusion

Nous avons appliqué la détection des aberrations aux données textuelles et nous avons également proposé une taxonomie dédiée. De plus, celle-ci peut être étendue à plusieurs corpus et applications. Une telle taxonomie peut aider à identifier de nouvelles caractéristiques dans un corpus. Sur la base de cette taxonomie, GenTO génère des aberrations indépendantes ou contextuelles à partir d'un corpus. Les expérimentations que nous avons menées montrent que les aberrations contextuelles sont plus difficiles à traiter que les aberrations indépendantes. Contrairement à la littérature, notre protocole expérimentale est transparent quant à la préparation des aberrations et des corpus.

Concernant les futurs travaux, nous souhaitons étudier le processus génératif pour les aberrations textuelles en absence d'étiquettes. De même, la robustesse par rapport au taux de contamination et aux types d'aberrations peut conduire à une métrique unique pour évaluer certaines approches. D'autres expériences sont prévues afin de proposer une étude comparative plus approfondie des approches pouvant réaliser la DDAT. En effet, nous souhaitons ajouter à l'étude des mécanismes propres au texte comme Kassab et Alexandre (2009).

Références

Aggarwal, C. C. (2017). *Outlier analysis* (Second edition ed.). Cham: Springer.

Allan, E. G., M. R. Horvath, C. V. Kopek, B. T. Lamb, T. S. Whaples, et M. W. Berry (2008). Anomaly Detection Using Nonnegative Matrix Factorization. In M. W. Berry et M. Castellanos (Eds.), *Survey of Text Mining II*, pp. 203–217. Springer.

Beckman, R. J. et R. D. Cook (1983). Outlier.....s. *Technometrics 25*(2), 119–149.

Breunig, M. M., H.-P. Kriegel, R. T. Ng, et J. Sander (2000). Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pp. 93–104.

Fouché, E., Y. Meng, F. Guo, H. Zhuang, K. Böhm, et J. Han (2020). Mining text outliers in document directories. In *2020 IEEE International Conference on Data Mining (ICDM)*, pp. 152–161. IEEE.

Gorokhov, O., M. Petrovskiy, et I. Mashechkin (2017). Convolutional Neural Networks for Unsupervised Anomaly Detection in Text Data. In H. Yin, Y. Gao, S. Chen, Y. Wen, G. Cai, T. Gu, J. Du, A. J. Tallón-Ballesteros, et M. Zhang (Eds.), *Intelligent Data Engineering and Automated Learning IDEAL 2017*, Volume 10585, pp. 500–507. Springer.

Hawkins, D. M. (1980). *Identification of Outliers.* Dordrecht: Springer.

Hodge, V. et J. Austin (2004). A Survey of Outlier Detection Methodologies. *Artificial Intelligence Review 22*(2), 85–126.

Kannan, R., H. Woo, C. C. Aggarwal, et H. Park (2017). Outlier detection for text data. *SDM International Conference on Data Mining 17*, 489–497.

Kassab, R. et F. Alexandre (2009). Incremental data-driven learning of a novelty detection model for one-class classification with application to high-dimensional noisy data. *74*(2), 191–234.

Koppel, M. et S. Seidman (2013). Automatically Identifying Pseudepigraphic Texts. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, Seattle, Washington, USA, pp. 1449–1454. Association for Computational Linguistics.

Lai, C.-H., D. Zou, et G. Lerman (2020). Robust subspace recovery layer for unsupervised anomaly detection. *ICLR International Conference on Learning Representations 8*, 1–28.

Liu, F. T., K. M. Ting, et Z.-H. Zhou (2008). Isolation Forest. In *2008 Eighth IEEE International Conference on Data Mining*, pp. 413–422.

Mahapatra, A., N. Srivastava, et J. Srivastava (2012). Contextual anomaly detection in text data. *Algorithms 5*(4), 469–489.

Manevitz, L. M. et M. Yousef (2001). One-class SVMs for document classification. *Journal of Machine Learning Research 2*, 139–154.

Mei, M., X. Guo, B. C. Williams, S. Doboli, J. B. Kenworthy, P. B. Paulus, et A. A. Minai (2018). Using Semantic Clustering And Autoencoders For Detecting Novelty In Corpora Of Short Texts. In *2018 International Joint Conference on Neural Networks (IJCNN)*, Rio de Janeiro, pp. 1–8. IEEE.

Ramaswamy, S., R. Rastogi, et K. Shim (2000). Efficient algorithms for mining outliers from large data sets. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data - SIGMOD '00*, Dallas, Texas, United States, pp. 427–438. ACM Press.

Ruff, L., J. R. Kauffmann, R. A. Vandermeulen, G. Montavon, W. Samek, M. Kloft, T. G. Dietterich, et K.-R. Müller (2021). A Unifying Review of Deep and Shallow Anomaly Detection. *Proceedings of the IEEE 109*(5), 756–795.

Ruff, L., Y. Zemlyanskiy, R. Vandermeulen, T. Schnake, et M. Kloft (2019). Self-attentive, multi-context one-class classification for unsupervised anomaly detection on text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4061–4071. Association for Computational Linguistics.

Shravan Kumar, B. et V. Ravi (2017). One-class text document classification with

OCSVM and LSI. In S. S. Dash, K. Vijayakumar, B. K. Panigrahi, et S. Das (Eds.), *Artificial Intelligence and Evolutionary Computations in Engineering Systems*, Volume 517, pp. 597–606. Springer.

Shyu, M.-L., S.-C. Chen, K. Sarinnapakorn, et L. Chang (2003). A novel anomaly detection scheme based on principal component classifier. Technical report, Miami Univ Coral Gables Fl Dept Of Electrical And Computer Engineering.

Srivastava, A. et B. Zane-Ulman (2005). Discovering recurring anomalies in text reports regarding complex space systems. In *2005 IEEE Aerospace Conference*, pp. 3853–3862. IEEE.

Toutanova, K., F. Chen, K. Popat, et T. Hofmann (2001). Text classification in a hierarchical mixture model for small training sets. In *Proceedings of the tenth international conference on Information and knowledge management - CIKM'01*, pp. 105. ACM Press.

Tran Manh Thang et Juntae Kim (2011). The anomaly detection by using DBSCAN clustering with multiple parameters. In *2011 International Conference on Information Science and Applications*, pp. 1–5. IEEE.

Walkowiak, T., S. Datko, et H. Maciejewski (2020). Utilizing local outlier factor for open-set classification in high-dimensional data - case study applied for text documents. In *Intelligent Systems and Applications*, Volume 1037, pp. 408–418. Springer.

Wang, F., R. J. Ross, et J. D. Kelleher (2018). Exploring Online Novelty Detection Using First Story Detection Models. In *Intelligent Data Engineering and Automated Learning IDEAL 2018*, Volume 11314, pp. 107–116. Springer.

Zhang, J. (2013). Advancements of Outlier Detection: A Survey. *ICST Transactions on Scalable Information Systems 13*(1), e2.

Zhao, Y., Z. Nasrullah, et Z. Li (2019). Pyod: A python toolbox for scalable outlier detection. *Journal of Machine Learning Research 20*(96), 1–7.

Summary

Outlier detection is the task that studies a set of data and estimates its outlying observations. This task is often performed to remove noise or abnormal data during the cleaning stage. While the literature is rich for many types of data, contributions for text are poorly represented. In this study we are interested in textual outliers. The appearance of these outliers can occur independently or contextually for a given topic. We introduce a taxonomy to identify them, and based on it, we propose GenTO which is an approach to generate outliers. Its use allows us to identify the strengths and weaknesses of an approach against different type of outliers. A comparative study of the state of the art approaches is finally performed using GenTO. Results show that, although recent work tends to perform better, it does not perform significantly better than previous one.

Automatic Reference Mining: Review and perspectives

Rodrigo Cuéllar-Hidalgo*, Gerardo Reyes-Salgado**,
Juan-Manuel Torres-Moreno***

*El Colegio de México
rcuellar@colmex.mx
http://colmex.mx
**CENIDET
gerardo.rs@cenidet.tecnm.mx
http://cenidet.tecnm.mx
***Laboratoire Informatique Université d'Avignon
juan-manuel.torres@univ-avignon.fr
https://lia.univ-avignon.fr

Abstract. During the last 15 years there has been a growing interest in automating the extraction of bibliographic references, mainly due to the large amount of scientific literature that is born digitally year after year and, especially, with the pandemic caused by COVID19. The objective of this work is to review the literature on the various techniques that have emerged for automatic bibliographic references, highlighting the most relevant approaches at present. The application of these techniques has provided various advances and even implementations that allow various institutions to design and adapt services that use these technologies, such as GROBID, CERMINE or ParsCite. It should be noted that despite the progress, there are still several unexplored areas of opportunity that in turn represent challenges in themselves.

1 Introduction

Currently, many repositories of scientific publications integrate automated processes for the extraction of citations, these systems, although useful, are usually far from perfect, and require constant maintenance, updating and correction of the information extracted, this has become a task of great importance and that presents constant growth (Patro, 2012; Tkaczyk et al., 2015; Agrawal et al., 2019), which exceeds the human capacities of the institutions that operate these repositories.

This is the case of digital libraries, which are having become an important resource for the scientific and academic communities, not only as places that store and facilitate the consultation of these publications, but also for their inherent classification and analysis activities, since they allow to significantly improve the capacities of grouping and retrieval of relevant information for its users.

It is precisely in this last point where the registration and analysis of citations and bibliographic references takes special relevance, which also have the characteristic of not presenting problems of ambiguity.

The registration and analysis of citations and bibliographic references allows not only to measure the impact of a publication on the scientific community, but also to extract new relevant information, such as the disciplines that cite a certain work, the geographical places where it is most consulted, the language in which it is most cited. All of this allows libraries to identify needs and areas of opportunity in their acquisition of material and development of special collections (Becker and Chiware, 2015).

The objective of this work is to review the various approaches to solve the Reference Mining task. To achieve this objective, searches were carried out to identify relevant literature using the terms "reference extraction, citation extraction and reference parsing". The databases consulted were Google Scholar and Microsoft Academic, identifying 112 articles that talk about the subject, of which only 28 match the inclusion criteria, which is, the article must deal with one or more of the sub-tasks of Reference Mining, and the rest were discarded.

The paper is organized as follows: in Section 2 we present our State of the Art of Reference Mining. In Section 3 we discuss some particularities obtained by the presented approaches. Finally, we conclude the paper in Section 4.

2 State of the Art

Rodrigues Alves et al. (2018) define Reference Mining (RM) as the task of parsing bibliographic references using NLP techniques, in order to solve the sub-tasks of detection, extraction and segmentation of its components.

These several sub-tasks are described as follows:

— **Detection:** Consists of locating one or more areas within the text that contains one or more bibliographic references.
— **Extraction:** Separates the previously detected area or areas into individual reference strings.
— **Segmentation:** Analyzes each reference chain and segment it into its components to transform them into metadata records.

Reference Mining (RM) is a fundamental process for building relational citation data that is used to build citation indexes. RM is considered a sequence labeling problem, which includes speech analysis (POS) and named entity recognition (NER).

Previous work in RM can be classified into two different groups that, in turn, can have different categories and levels depending on their approaches. These groups can be seen in Figure 1, below, each group will be described with respect to its approaches, techniques and the sub-tasks they solve.

2.1 Images as contextual information

In this group, we find the approaches that use images as a contextual information base (Rizvi et al., 2019; Bhardwaj et al., 2017). These approaches are inspired by human vision to detect the chains of references in the image of a text document, that is, they focus on the problem using computer vision, using CNN standing out in the following:

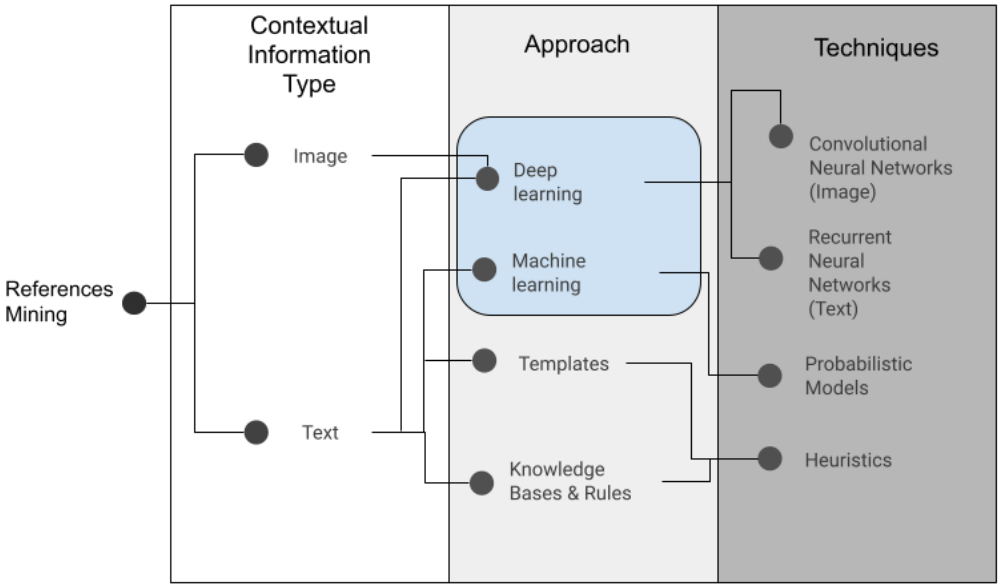


FIG. 1 – *Groupings of the different RM approaches and their levels.*

They are perfect for working with documents that were not born digitally. since they avoid the errors of the Optical Character Recognition (OCR) process. They are independent of language.These approaches are only useful for covering the reference string detection and extraction subtasks, not for segmentation. The approach by Rizvi et al. (2019) compensates for this disadvantage by applying OCR to the extracted reference strings and using Conditional Random Fields (CRF) for the segmentation task.

It is important to mention that Bhardwaj et al. (2017) used an implementation of a Fully Convolutional Neural Network (CNN), also known as FCN-8, this implementation bears the name of DEEPBibx, while Rizvi et al. (2019) uses a CNN with region masks (Mask R-CNN) and its implementation bears the name of DeepBird.

The main advantage of this approach is that they are capable of detecting strings of references in any part of the document, not only in a specific section, in addition to being totally agnostic of the language or the citation style.

2.2 Text as contextual information

This group of papers is the one that contains the largest number of approaches, and all of them present different approaches that can be categorized as follows. These approaches consider RM as a problem of tagging sequences at the word level (token) and in turn these approaches can be divided into two categories, those that use Machine Learning (ML) models and those that use models based on Deep Learning (DL).

2.2.1 Approaches based on Machine Learning

ML approaches are based on statistical models that learn from labeled sets of bibliographic references. and they usually cover all the subtasks of RM in this way. Some RM works in this category address the subtask detection by identifying the reference section in a document, for this they use Support Vector Machines (SVM) (Zou et al., 2010; Tkaczyk et al., 2015). Meanwhile, others (Lopez, 2009; Körner, 2017) opt for the use of CRF, due to their ability to model limits when deciding between different classes to assign a label to a word.

Regarding the subtasks of extraction and segmentation, SVM, CRF and Hidden Markov Models (HMM) have mainly been used. Since these subtasks correspond to the sequence tagging problem, HMM seems to be the appropriate tool to estimate the tag of a token based on previously detected states (Boukhers et al., 2019).

Hetzner (2008) applies a model based on simple first-order HMM, managing to approximate 90% in a set of homogeneous reference data from the health sciences domain. Yin et al. (2004) manage to further refine the results using a variant of the HMM model, known as Bigram, which simply modifies the estimation probability without modifying the proper structure of an HMM. For his part, Ojokoh et al. (2011), used another variant of an HHM model, called Trigram, which is considered second-order and whose particularity is a larger context window size when estimating the probability that a token corresponds to a certain tag.

In Zhang et al. (2011), the authors propose a structural SVM to segment the references of biomedical literature, where the precision reached was approximately 98%, this was mainly due to the fact that this domain has a strong structural regularity in its style of references. It is important to mention that (Zou et al., 2010) made a comparison between SVM and CRF in the same domain, finding that both techniques have almost identical precision (approximately 97%).

Peng and McCallum (2006) were the first to exploit the power of CRF models by applying Gaussian variations. These models are a specialization of HHMs, which basically condition the calculation of the probability of a token corresponding to a specific label, taking into account the labels of previous iterations. Romanello et al. (2009) uses CRF combined with an N-Grams analyzer (subsequence of n elements of a given sequence) for a special type of reference chain known as "Canonical" which correspond to classical texts. It is important to note that the CRF models are a reference in the field of RM, to the point that the first implementations (integrated tools for public use) use them, as is the case of GROBID (available on: https://github.com/kermitt2/grobid) (Lopez, 2009) and ParsCit (Available on: https://github.com/knmnyn/ParsCit) (Councill et al., 2008) both use CRF.

Tkaczyk et al. (2015) publish their CERMINE (available on:http://cermine.ceon.pl/index.html) implementation, which use various tools to achieve the complete flow that solves all the RM subtasks. This method uses heuristic and ML techniques, such as: Support Vector Machines (SVM), K-Means and CRF; achieving good results and allowing users who are not experts in these techniques to do RM.

Unlike the other approaches, Körner (2017) proposes a model based on CRF that does not label words, but takes into account all the lines of a document, instead of only identifying the reference section, in this way it is possible to create a model that classifies lines instead of individual words, which reduces the complexity of the model to be used and allows extracting strings of references contained in the entire document and not just in a section. It should be noted that this approach is only useful for RM detection and extraction tasks.

It is relevant that, given that it implies a smaller number of random variables, this is efficient when considering a training with a relatively low amount of manually labeled data. In a later publication, Körner et al. (2017) compare their approach with others such as GROBID using the German language. In addition to that, Boukhers et al. (2019), present a holistic approach to address the problem of RM, that is, a whole flow was designed to contemplate subtasks detection, extraction and segmentation of the RM (unlike other approaches that only focus on one of these tasks), in order to have a coherent scheme that reduces errors and follows a probabilistic approach.

This approach works in two correlated phases. The first is the line classification, which by means of a trained model based on Random Forest (RF), labels each line of a text that belongs to a reference, classifying them into three types: first line, intermediate line and last line In this way, each reference chain is extracted to later enter the second phase, which consists of segmenting the extracted reference chains through CRF.

It is concluded that, given the results, this approach surpasses other approaches given its holistic approach, in addition to that, the fact that this approach is able to search for references that are not contained in a specific section if not distributed throughout the document is highly common in social sciences and humanities.

It is important to mention a comment made by Yang et al. (2020) on approaches based on ML, who point out that their precision depends on manually designed characteristics, and therefore they are dependent on the domain that is being studied, this fact implies that models trained in a domain are not usually suitable to be transferred just as they are to another domain and/or bibliographic reference style, which implies the need to retrain them, generate tagged data for this and most likely, require adjustments in their parameters.

2.2.2 Approaches based on Deep Learning

Boukhers et al. (2019), consider that the deficiency of the little generalization of data in ML-based approaches, can be overcome by using Deep Neural Networks, which are considered to have a great capacity to represent data which they train in a more precise and general way. An example of this is the one developed by (Huang et al., 2015), who developed an RNN model based on a bidirectional LSTM (Bi-LSTM), which extracts representations of words and their relationships with both previous and subsequent sequences of words, to finally feed an output towards a CRF model.

Continuing on this idea, Prasad et al. (2018) designed an approach where they use a Neural Network of the LSTM type to obtain representations of the tokens that correspond to a reference and, with the characteristics extracted from these, train a model based on CRF. In this way, it was possible to create a solid model with characteristics extracted by the RNN, thus avoiding the need to manually create the training characteristics and giving rise to the Neural-Parscit (available on: https://github.com/WING-NUS/Neural-ParsCit) implementation.

The last of the works in this line is that of Rodrigues Alves et al. (2018), who used a Bi-LSTM-CRF (available on: https://github.com/dhlab-epfl/LinkedBooksDeepReferenceParsing/) architecture, where they use SoftMax-CRF prediction layers and embeddings of words and characters, in order to be able to identify references in an entire text document (Domains like Humanities and Social Sciences often contain references in different parts of the document) and not only in the references section.

It is important to take into account what was said by Grennan and Beel (2020), who point out an important bias to the approaches based on Deep Learning, which consists in the insufficient amount of labeled data for effective training, for which they were given the task of comparing the efficiency of trained models with real data and manually labeled, with what they call "synthetic reference strings", which are basically reference strings that do not correspond to a work, meaning that they are false, concluding that the synthetic data is very suitable for the training of models for dating analysis.

At the time of writing this article, no approach could be found that implements Transformers for mining bibliographic references, instead only one was found for the extraction of patents (Voskuil and Verberne, 2021), which leaves it outside the scope of this article.

2.2.3 Templates-based approaches

These approaches are efficient for the segmentation subtask of RM, due to their ability to represent patterns in a general way. Templates are designed based on prior knowledge of the domain, which is why they extract relevant information from texts when the conditions defined in them coincide (Boukhers et al., 2019). Despite their efficiency, their design requires a great deal of human effort, and given the non-uniform nature (in practice) of reference writing, they are often not considered a feasible approach.

Chowdhury (1999) makes the first attempt to solve this, proposing a format template for the writing of the reference by the authors, in order to reduce irregular citations and therefore the complexity in the design of the templates. For their part, Ding et al. (1999) used a greater number of templates (three instead of one) to solve the problem of irregular patterns in the writing of the references, achieving good results to extract components with little variability, such as authors or journal names. This did not work for other components.

Chen et al. (2007) propose BibPro, an implementation that improved the performance of previous works by replacing the Knowledge Base (KB) with punctuation symbols to better identify the reference format, reporting better results than (Councill et al., 2008) using a test data set that contained six different reference styles.

Day et al. (2007) propose another implementation that uses hierarchical templates, INFOMAP, which allow them to overcome the lack of customization of the rules through a hierarchical tree structure. In this way, it is able to represent various patterns for the components of different styles of references.

2.2.4 Approaches based on Rules and Knowledge Bases

These approaches are mainly based on the design of rules through heuristics and KB to be able to face the tasks of extraction and segmentation of RM. The main exponents of these approaches are: Cortez et al. (2007) with their approach called Flux-CiM and Hsieh et al. (2014) with Framework Based Approach (FBA).

Cortez et al. (2007) present Flux-CiM, which uses a KB for the extraction of references, this approach emphasizes that it does not require a training process on the case of models. based on machine and/or DL, achieving great flexibility and sustainability. All its power comes from a KB that basically consists of a structured reference collection.

The process consists of identifying each reference chain, segmenting the words into blocks (analogous to the concept of tokenization) and identifying the separating markers (punctuation

marks such as commas, dots, double dots, etc.) and estimating the probability that a value will appear in a certain segment taking into account what KB exists in it and the values found in a certain field and also takes into account neighboring blocks to increase precision. The results ranged around 93% effectiveness in their tests where it was applied to two different domains (computer science and health sciences).

For their part, Hsieh et al. (2014) present a Framework Based Approach (FBA), which proposes to build a flexible representation of the information, which is constructed by experts who manually compile abbreviations, position limit patterns, and prefixes; to finally use a matching approximation algorithm and thus recognize the desired components of the reference strings. Finally, it was compared with a CRF-based solution in different domains, achieving an error reduction of about 70%.

2.2.5 Other approaches

It is worth mentioning that approaches that have recently been presented do not fit into any of the previous categories. The first case is that of Tkaczyk et al. (2018b), with the approach called ParsRec, which has the peculiarity of being an approach based on recommendation and meta-learning. Its main premise is "Although there are many reference analyzers, which approach the problem from different approaches, none of them offers optimal results in all scenarios".

ParsRec recommends the best analyzer, out of ten contemplated, based on the chain reference in turn. The results are considered promising by the authors, since they show the potential of the system. The combination of both proposed approaches outperforms the best analyzer by up to 18.9% in reducing the error rate.

The last case is that of Yang et al. (2020) who propose the “Principle Based Approach” (PBA) to address the RM extraction and segmentation subtasks. Starting from an automatic template generator that captures patterns using a dominant set algorithm, these templates are capable of extracting references using an alignment-based template matching technique that uses a logistic regression model, which makes it more general and flexible than rule-based approaches.

This approach has been compared with other existing approaches, showing better performance in various data sets. In the words of the authors, this approach takes the best of template-based approaches and statistical modeling.

It is concluded that the main contribution of the PBA is to maintain the explainability of the template-based methods, while taking advantage of the optimization capacity of the statistical models of ML.

2.3 Comparison

To finish the focus section, it is important to mention that Tkaczyk et al. (2018a) made a comparison between different approaches for the extraction of citations, among these are: CRF, Regular Expressions, Rules, Template Comparison and LSTM Neural Networks.

Although a brief description of the state of the art is made, it is clarified that it was not possible to evaluate all the approaches since there are no public and/or available versions of said tools (or there are errors in their installation), as is the case of knowledge base-based approaches or LSTM-based neural networks.

During their comparison, they determined that three of the CRF-based tools have the best performance with a set of data generated to solve a case with requirements defined by them. Subsequently, they re-trained the models of these three tools, improving their precision and concluding that this CRF is the best of the approaches that they could evaluate, mainly due to its ability to adapt it to different extraction requirements and different citation styles.

2.4 Datasets

Datasets are usually collected by each group of researchers in most of the articles reviewed, without giving access to them; however, there are several exceptions that used open access data sets and/or developed their own and made them public, Table 1 compiles these cases.

TAB. 1 – *Datasets used in most of the reviewed articles.*

Data set	**Observations**	**Domain**
Grobid (constant growth)	6835	Multidomain
Cora	1295	Computer science
Citeseer	1563	Artificial intelligence
FLUX-CiM CS	300	Computer science
FLUX-CiM HS	2000	Health Sciences
GROTOAP2	6858	Biomedicine, Sciences of the computing
Venice	40000	Humanities
GIANT	One billion	Multi-domain and multi-dating styles

Grennan et al. (2019) point out that a great bias in the literature is that less than the previously mentioned data sets are used, and that is why they were in charge of developing a massive data set, known as Giant (available on: `https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/LXQXAO`), which contains properly labeled reference strings in XML, covering multiple domains and citation styles, and of course it is freely accessible.

2.5 List of cited sources

Table 2 compiles a comparison of all the publications cited in the present article, where key phrases, categories, subtasks and the evaluation obtained in the experimental cases are listed.

It should be noted that specifically speaking of the **Results** column, the value corresponds to the average of the results offered by the authors in their own training data set.

TAB. 2 – *Summary of the main characteristics of the sources discussed in this article*

Authors	Key pharses	Categorie	Sutasks	Results
Grennan and Beel (2020)	Synthetic references	Datasets	N/A	N/A
Yang et al. (2020)	PBA - Template Generation - Meta-Learning	Others	Segmentation	Accuracy 0.9833
Rizvi et al. (2019)	DeepBird - Computer vision	Deep Learning	Detection y Extraction	AP50 98.56%
Boukhers et al. (2019)	Random Forest - CRF - Holistic Approach	Machine Learning	All	Extraction F1-Score 0.78 Segmentation de F1-Score 0.92
Grennan et al. (2019)	Giant - Massive Datasets	Datasets		N/A
Prasad et al. (2018)	Neural Parscit - LTSM - CRF	Deep Learning	Segmentation	F1-Score 0.9137
Rodrigues Alves et al. (2018)	Bi-LTSM-CRF Embedings for data preProccessing	Deep Learning	All	Extraction F1-Score 0.9509 Segmentation F1-Score 0.8966
Tkaczyk et al. (2018b)	ParsRec - Parser recommendation - Meta-Learing	Others	All	F1-Score 0.891
Tkaczyk et al. (2018a)	Comparison	Comparison	Segmentation	N/A
Körner (2017)	CRF - line detection intestate word	Machine Learning	Detection y Extraction	N/A
Körner et al. (2017)	CRF - line classification intestate word	Machine Learning	Detection y Extraction	F1-Score 0.885
Bhardwaj et al. (2017)	DeepBibx - Vision computacional	Deep Learning	Detection y Extraction	Precición 0.839 Recall 0.846
Tkaczyk et al. (2015)	Cermine - SVM CRF K-Means - Heuristic	Machine Learning	All	Extraction F1-Score 0.39 Segmentation F1- Score 0.89
Hsieh et al. (2014)	FBA - principle-based approach - Knowledge bases FBA	Rules y KB	Segmentation	Accuracy 0.9795
Ojokoh et al. (2011)	HMM trigram	Machine Learning	Segmentation	F1-Score 0.8966
Zhang et al. (2011)	structural SVM	Machine Learning	Segmentation	Accuracy 0.9899
Zou et al. (2010)	SVM and CRF compared in HTML	Machine Learning	Segmentation	Accuracy 0.974
Romanello et al. (2009)	CRF N-grams	Machine Learning	Detection y Extraction	F1-Score 0.8707
Lopez (2009)	Grobid - CRF	Machine Learning	Segmentation	F1-Score 0.89
Hetzner (2008)	HMM - Viterbi	Machine Learning	Segmentation	F1-Score 0.847
Councill et al. (2008)	ParsCit - CRF	Machine Learning	Segmentation	F1-Score 0.95
Day et al. (2007)	Infomap - Hierarchical knowledge	Templates	Segmentation	Accuracy 0.9239
Chen et al. (2007)	BibPro – Order of punctuation marks as feature	Templates	Segmentation	F1-Score 0.9043
Cortez et al. (2007)	Flux-CiM – KB, Unsupervised	Rules y KB	Segmentation	F1-Score 0.9639
?	CRF - Feature engineering	Machine Learning	Segmentation	F1-Score 0.915
Yin et al. (2004)	HMM bigram	Machine Learning	Segmentation	Precision 0.9015 Recall 0.915
Ding et al. (1999)	Multiple Templates to overcome irregular patterns	Templates	Segmentation	N/A
Chowdhury (1999)	Heuristic - reduces error when proposing fill format for writing references	Templates	Segmentation	N/A

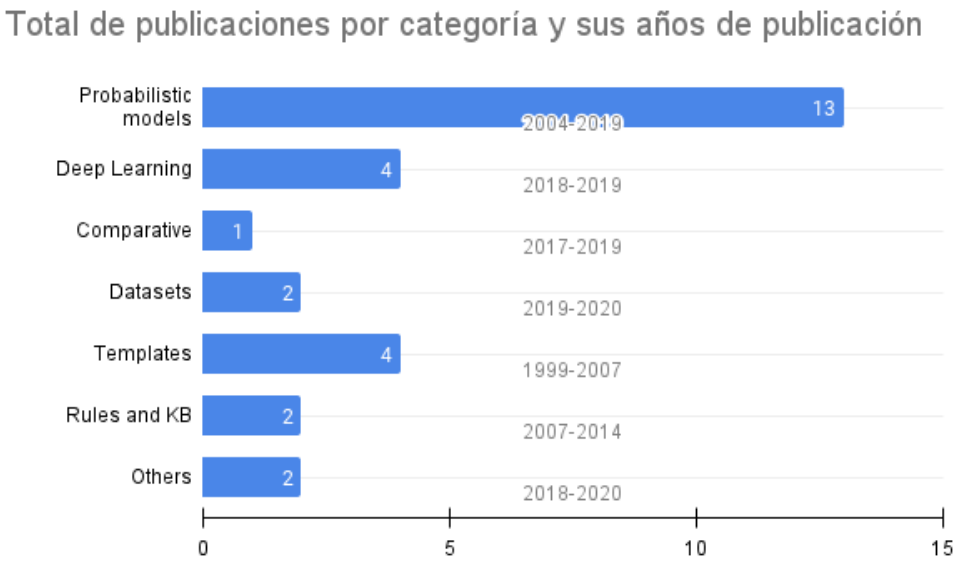


FIG. 2 – *Categories detected in the state of the art.*

3 Discussion

Of the 28 articles that met the inclusion criteria, a series of well-differentiated categories were distinguished (see Figure 2).

The template-based approach was one of the first (it was born in 1999) but began to be discontinued in 2011 and only two publications were found that use the Rules and KB approach, but they were discontinued in 2007.

It highlights the fact that the most popular approach is ML (17 publications), of which 13 correspond to probabilistic models. In addition, it is also distinguished by the fact that it is the approach that has been the longest (fifteen years) on the subject of RM. The approach based on DL has a short time (it started in 2015) and present a few number of publications (4).

Recently, approaches that address RM problems from a "higher" perspective have emerged, such as in the case of Yang et al. (2020) calls it "Meta-learning", which is basically about making use of ML to find design templates (reviving this approach) that can be adjusted to each reference that you want to identify, extract and segment, while the other approach (Tkaczyk et al., 2018a) consists of an algorithm that recommends an ideal reference parser for each case. Amongst comparisons between approaches, only one has emerged, and it did not have the means to assess DL.

It is worth to mention that the RM topic has led to the appearance of publications that specifically deal with the subject of data sets, where in the first one a set with more than a billion observations is proposed (Grennan et al., 2019). And in the second, the possibility of creating "synthetic citations" is explored, concluding that they are effective for training models based on DL (Grennan and Beel, 2020).

Factor	Value	Quantity
Language	German	1
	English	27
Domain	Humanities	1
	Computer Science/Health Sciences	27
Context	Image	2
	Text	26
Subtask	Identification, Extraction and Segmentation	3
	Identification and Extraction	4
	Segmentation	21

TAB. 3 – *List of works reviewed and Context Factors, Subtasks, Language, Metrics and Domain.*

We can observe, distinctive aspects of the reviewed publications within the state of the art in 3. Reaching the following statements:

— Virtually all publications work with references in the English language (96.42%).
— Practically all the publications work with references that belong to the domain of Computer Sciences and/or Health Sciences, which have a very uniform citation style (96.42%).
— Segmentation is the most studied subtask (74.99%). The most used context information is the text (92.85%).
— Image-based context information (7.14%) is only used to Identify and Extract tasks.
— There are few publications that address the entire RM process (10.71%).

4 Conclusion

Given the biases detected within the literature, with respect to language, what was said by Bender (2019) should be taken into account, who highlights that the great linguistic variety, between one language and another, has a negative impact on the effectiveness of the ML techniques that work with natural language, because these are only designed, trained and tested with Corpus in the English language. An example of this is the metadata extractor for texts in Spanish, proposed by Iturbe Herrera et al. (2019), which, when compared with other approaches developed for the English language, failed to match its results in the precision and coverage metrics of the different metadata to be extracted.

Regarding the domain of the references, given that the different approaches focus on estimating the probability that a certain token corresponds to a class, taking into account the classifications of the previous tokens, it is appropriate to assume that a model trained in a certain domain does not have the expected results when trying to classify references from another domain. A representative case of this is the arts and humanities mentioned by Rodrigues Alves et al. (2018).

A final bias to highlight is the lack of transformer-based approaches, the nature of these models could well provide better results to MR subtasks.

Therefore, it is concluded that if RM is really expected to have a true impact on scientific work, it is imperative to start expanding the frontiers in the application of techniques already developed, testing them in new scenarios (such as new domains and/or languages) and if problems are detected, to develop strategies to overcome them.

References

Agrawal, K., A. Mittal, and V. Pudi (2019). Scalable, Semi-Supervised Extraction of Structured Information from Scientific Literature. pp. 11–20.

Becker, D. A. and E. R. Chiware (2015). Citation Analysis of Masters' Theses and Doctoral Dissertations: Balancing Library Collections With Students' Research Information Needs. *Journal of Academic Librarianship 41*(5), 613–620.

Bender, E. (2019). The #benderrule: On naming the languages we study and why it matters. *The Gradient*.

Bhardwaj, A., D. Mercier, A. Dengel, and S. Ahmed (2017). DeepBIBX: Deep Learning for Image Based Bibliographic Data Extraction. *LNCS 10635 LNCS*(October), 286–293.

Boukhers, Z., S. Ambhore, and S. Staab (2019). An end-to-end approach for extracting and segmenting high-variance references from PDF documents. In *ACM/IEEE Joint Conference on Digital Libraries*, Volume 2019-June, pp. 186–195. IEEE.

Chen, C. C., K. H. Yang, C. L. Chen, and J. M. Ho (2007). BibPro: A citation parser based on sequence alignment. *IEEE Transactions on Knowledge and Data Engineering 24*(2), 236–250.

Chowdhury, G. G. (1999). Template mining for information extraction from digital documents. *Library Trends 48*(1), 182–208.

Cortez, E., A. S. Da Silva, M. A. Gonçalves, F. Mesquita, and E. S. De Moura (2007). FLUX-CIM: Flexible unsupervised extraction of citation metadata. *Proceedings of the ACM International Conference on Digital Libraries* (January), 215–224.

Councill, I. G., C. L. Giles, and M.-Y. Kan (2008). ParsCit: An open-source CRF Reference String and Logical Document Structure Parsing Package. In *6th International Conference on Language Resources and Evaluation*, Number LREC 2008, pp. 661–667.

Day, M. Y., R. T. H. Tsai, C. L. Sung, C. C. Hsieh, C. W. Lee, S. H. Wu, K. P. Wu, C. S. Ong, and W. L. Hsu (2007). Reference metadata extraction using a hierarchical knowledge representation framework. *Decision Support Systems 43*(1), 152–167.

Ding, Y., G. Chowdhury, and S. Foo (1999). Template Mining for the Extraction of Citation From Digital Documents. *Proceedings of the Second Asian Digital Library Conference Taiwan 639798*(February 2013), 47–62.

Grennan, M. and J. Beel (2020). Synthetic vs. Real Reference Strings for Citation Parsing, and the Importance of Re-training and Out-Of-Sample Data for Meaningful Evaluations: Experiments with GROBID, GIANT and Cora. *arxiv.org*, 1–7.

Grennan, M., M. Schibel, A. Collins, and J. Beel (2019). Giant: The 1-billion annotated synthetic bibliographic-reference-string dataset for deep citation parsing. In *CEUR Workshop Proceedings*, Volume 2563, pp. 260–271.

Hetzner, E. (2008). A simple method for citation metadata extraction using Hidden Markov models. *Proceedings of the ACM International Conference on Digital Libraries*, 280–284.

Hsieh, Y. L., S. H. Liu, T. H. Yang, Y. H. Chen, Y. C. Chang, G. Hsieh, C. W. Shih, C. H. Lu, and W. L. Hsu (2014). A frame-based approach for reference metadata extraction. *LNCS 8916*, 154–163.

Huang, Z., W. Xu, and K. Yu (2015). Bidirectional LSTM-CRF Models for Sequence Tagging.

Iturbe Herrera, A., A. Montes Rendón, J.-M. Torres-Moreno, G. Sierra Martínez, N. A. Castro Sánchez, and J. G. González Serna (2019). Extracción semiautomática de metadatos en documentos no estructurados utilizando procesamiento de lenguaje natural y propiedades tipográficas. *Research in Computing Science 148*(7), 331–345.

Körner, M. (2017). Reference String Extraction Using Line-Based Conditional Random Fields.

Körner, M., B. Ghavimi, P. Mayr, H. Hartmann, and S. Staab (2017). Evaluating reference string extraction using line-based conditional random fields: A case study with German language publications. *Communications in Computer & Information Science 767*, 137–145.

Lopez, P. (2009). GROBID: Combining automatic bibliographic data recognition and term extraction for scholarship publications. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Volume 5714 LNCS, pp. 473–474.

Ojokoh, B., M. Zhang, and J. Tang (2011). A trigram hidden Markov model for metadata extraction from heterogeneous references. *Information Sciences 181*(9), 1538–1551.

Patro, S. (2012). *Data Clustering and Cleansing for Bibliography Analysis*. Ph. D. thesis, University of New South Wales.

Peng, F. and A. McCallum (2006). Information extraction from research papers using conditional random fields. *Information Processing and Management 42*(4), 963–979.

Prasad, A., M. Kaur, and M. Y. Kan (2018). Neural ParsCit: a deep learning-based reference string parser. *International Journal on Digital Libraries 19*(4), 323–337.

Rizvi, S. T. R., A. Dengel, and S. Ahmed (2019). DeepBiRD: An Automatic Bibliographic Reference Detection Approach.

Rodrigues Alves, D., G. Colavizza, and F. Kaplan (2018). Deep Reference Mining From Scholarly Literature in the Arts and Humanities. *Frontiers in Research Metrics and Analytics 3*.

Romanello, M., F. Boschetti, and G. Crane (2009). Citations in the digital library of classics. (August), 80.

Tkaczyk, D., A. Collins, P. Sheridan, and J. Beel (2018a). Machine Learning vs. Rules and Out-of-the-Box vs. Retrained: An Evaluation of Open-Source Bibliographic Reference and Citation Parsers. *ACM/IEEE Joint Conference on Digital Libraries*, 99–108.

Tkaczyk, D., R. Gupta, R. Cinti, and J. Beel (2018b). ParsRec: A novel meta-learning approach to recommending bibliographic reference parsers. In *CEUR Workshop Proceedings*, Volume

2259, pp. 162–173.

Tkaczyk, D., P. Szostek, M. Fedoryszak, P. J. Dendek, and Ł. Bolikowski (2015). CERMINE: Automatic extraction of structured metadata from scientific literature. *International Journal on Document Analysis and Recognition 18*(4), 317–335.

Voskuil, K. and S. Verberne (2021). Improving reference mining in patents with BERT.

Yang, T. H., Y. L. Hsieh, S. H. Liu, Y. C. Chang, and W. L. Hsu (2020). A flexible template generation and matching method with applications for publication reference metadata extraction. *Journal of the Association for Information Science and Technology*, asi.24391.

Yin, P., M. Zhang, Z. H. Deng, and D. Q. Yang (2004). Metadata extraction from bibliographies using bigram HMM. *LNCS 3334*(June 2014), 310–319.

Zhang, X., J. Zou, D. X. Le, and G. R. Thoma (2011). A structural SVM approach for reference parsing. *BMC Bioinformatics 12*(SUPPL. 3).

Zou, J., D. Le, and G. R. Thoma (2010). Locating and parsing bibliographic references in HTML medical articles. *Int. Journal on Document Analysis & Recognition 13*(2), 107–119.

Appréhender les dynamiques spatiales et thématiques à travers la fouille de textes

Mathieu Roche
CIRAD - Environments and Societies Department, UMR TETIS, Montpellier

mathieu.roche@cirad.fr

Résumé :

Les dynamiques spatiales et thématiques sont étudiées avec grande attention en particulier dans les milieux à enjeux et de tension (épidémiologie, sécurité alimentaire, etc.). L'objectif de nos travaux est de proposer un cadre méthodologique permettant l'appréhension de ces phénomènes à partir de données hétérogènes, en particulier textuelles. Les analyses que nous produisons reposent sur trois types d'informations (thématiques, spatiales et temporelles) qui sont extraites et exploitées par des méthodes de fouille de textes. Les contributions méthodologiques seront présentées sous le prisme de différents projets pluridisciplinaires (par exemple, les projets H2020 MOOD et LEAP4FNSSA) et le déploiement d'outils et de plate-formes spécifiques. Dans ce contexte, le système PADI-Web propre à la veille automatique en épidémiologie animale et des approches dédiées à la sécurité alimentaire dans les pays du Sud seront présentés.

Détection de données personnelles pour la pseudonymisation de documents numérisés

Maëlle Brassier*,**, Asceline Goudjo*, Bernard Peultier*

* Nextino, 1 Avenue du Champ de Mars, 45100 Orléans
prenom.nom@nextino.eu
** LIFAT, Université de Tours, 41000 Blois, France
maelle.brassier@etu.univ-tours.fr

Résumé. Malgré un intérêt croissant et une mise en avant de la protection de la vie privée dans la sphère publique, des masses de données continuent d'être échangées quotidiennement à travers le web avec parmi elles des informations relatives à un individu, à savoir des données personnelles capables d'identifier l'identité de la personne concernée. Dans ce papier, nous présentons une première version de IA4trust, un système de pseudonymisation de documents qui s'attache à répondre aux différentes problématiques du RGPD tout en visant à produire un document pseudonymisé le plus cohérent sémantiquement possible. Nous procédons à une étape de dé-idenfication d'un ensemble de type de données personnelles (nom, email, adresse…) défini. Nous comparons différentes techniques par type de données personnelles et montrons que l'utilisation de différents modules de dé-identification est nécessaire.

1 Introduction

L'inquiétude autour de la sécurité et de la vie privée trouve ses racines bien avant le début des années 2000, avec notamment le système de protection des données de l'OCDE en 1980 qui pose les premiers principes de confidentialité au niveau international. À l'échelle nationale, la Loi n° 78-17 du 6 janvier 1978, plus connue sous la dénomination de Loi Informatique et libertés fait office de premières lignes directives régissant la protection de la vie privée. Cette loi a connu plusieurs évolutions et une réécriture selon l'ordonnance du 20 juin 2018. C'est dans cette même période que l'Union européenne adopte le Règlement Général sur la Protection des Données (RGPD), successeur de la Directive 95/46/CE sur la protection des données personnelles. Cette réglementation résulte de nombreuses anciennes directives et vise à la fois à moderniser et harmoniser la protection des données des personnes physiques au sein de l'Union européenne.

Le RGPD incite donc chaque acteur à mettre en place un ensemble de moyens pour protéger la vie privée, ce qui peut se traduire en deux processus bien distincts :

— l'**anonymisation** qui constitue un faisceau de techniques mises en place afin de rendre une donnée personnelle anonyme de façon définitive et irréversible, c'est-à-dire en ren-

dant toute identification à un individu et ré-association avec l'information originale impossibles

— la **pseudonymisation** qui, à l'inverse est réversible et englobe le traitement des données à caractère personnel de manière à ce que ces dernières perdent leur caractère nominatif. En somme la donnée personnelle d'origine est conservée puis remplacée par un substitut qui, sans information supplémentaire, ne permet pas d'identifier une personne physique

C'est dans ce cadre et cette problématique que nous avons développé IA4trust : un système de pseudonymisation capable de produire des documents pseudonymisés qui reste le plus proche et le plus cohérent possible vis à vis de sa version originale. Dans ce travail, nous nous intéressons principalement à la détection des mentions dénommantes, à savoir des entités tenant lieu de données personnelles ou permettant l'identification des personnes. La contribution de cet article s'inscrit alors dans une démarche de Traitement Automatique des Langues et ne traite pas ici de la problématique de ré-identification de l'identité d'un individu à travers différentes techniques telles que le recroisement de données. En outre, cette tâche peut se rapprocher par certains aspects de celle de la Reconnaissance d'Entités Nommées (NER).

2 État de l'art

L'un des premiers travaux sur la pseudonymisation a été réalisé par Sweeney (1996) qui introduit à cette époque la notion de substituts en remplaçant les dates dans des dossiers de patient par d'autres dates proches temporellement et les noms par des noms fictifs. Deux types de données dénommantes [1] sont distinguables : les "identifieurs" qui permettent d'identifier directement une personne et les "quasi-identifieurs" qui, à l'inverse, ne renvoient pas directement vers une personne mais peuvent être utilisés pour l'identification d'une personne, notamment via des combinaisons de différents quasi-identifieurs.

Comme mentionné précédemment, la dé-identification passe par des étapes de détection d'entités dénommantes et leur remplacement dans le texte par substitution. Pour réaliser la pseudonymisation d'un document, il faut donc déjà pouvoir identifier les entités dénommantes présentes dans le texte. Le domaine médical possède un nombre important de travaux sur la détection de données à caractère personnel dû à leur présence massive au sein des dossiers électroniques de patients [2]. Les techniques précurseuses ont été des approches à base de règles telles que Gupta et al. (2004), suivies par des techniques d'apprentissage automatique comme les arbres de décision (Szarvas et al., 2006) et les Machines à Vecteurs de Support (SVM) (Hara, 2006).

Les entités dénommantes ont une structure très variée suivant leur type : un numéro de téléphone répond ainsi par exemple à des contraintes structurelles bien plus forte qu'une adresse postale. D'autres approches, que l'on qualifiera d'hybrides, consistent précisément à utiliser des techniques de détection spécifiques à chaque type d'entité et/ou à combiner plusieurs techniques. Ces approches offrent les meilleures performances comme l'ont montré plusieurs campagnes d'évaluation portant sur la dé-identification de dossiers cliniques. Sur 8 systèmes soumis à la tâche 1 de dé-identification i2b2/UTHealth de 2014, 6 d'entre eux proposent des

1. on utilise le terme "entités dénommantes" tel que l'a introduit Eshkol (2010) comme synonyme de données personnelles

2. Electronic health records (EHR)

systèmes hybrides d'approches Machine Learning et d'approches symboliques avec 3 d'entre eux se trouvant dans les 4 meilleurs systèmes (Yang et Garibaldi, 2015). Le meilleur système du défi propose un modèle hybride combinant une méthode de Champ aléatoire conditionnel (CRF), une approche à base de règles et le repérage de mots-clés (Yang et Garibaldi, 2015). Les auteurs justifient leur choix par le fait que les entités dénommantes sont hétérogènes, affichant des caractéristiques distinctes au niveau du lexique, syntaxe et sémantique et de ce fait, un modèle hybride couplant plusieurs techniques est plus adapté qu'un unique modèle de langage. Leur stratégie de combinaison d'approches repose également sur la quantité de données d'entraînement par catégorie d'entité dénommantes.

Pour la tâche 1.B du défi de dé-identification 2016 CEGS N-GRID (Stubbs et al., 2017), seulement 5 systèmes hybrides ont été soumis pour un total de 15 systèmes, néanmoins les quatre premiers systèmes du défi se sont avérés être des systèmes hybrides. Le meilleur système par (Liu et al., 2017) compare 4 méthodes distinctes : une à base de CRF, une à base de Bidirectional LSTM (BI-LSTM) et un BI-LSTM avec features. Bien que le Bi-LSTM avec features surpasse le BI-LSTM qui lui même surpasse le CRF, les auteurs observent que chaque approche identifie certaines occurrences d'entités dénommantes que les autres approches ne parviennent pas à détecter. De plus, aucune méthode ne dépasse les autres sur l'ensemble complet des catégories. De ce fait, en combinant les trois approches via un classifieur ensembliste dont le résultat est fusionné avec celui d'un sous-système à base de règles, le système final obtient un f-score de 0.9143.

L'état de l'art sur la détection et la dé-identification des entités dénommantes montre que leur nature variée et leur spécificité justifient une approche hybride. C'est cette observation de la pertinence des approches hybrides qui nous a conduit à adopter une telle démarche pour le développement de notre système à visée industrielle. Ce système est entraîné et évalué sur un corpus de données originales répondant à nos besoins applicatifs. Ce jeu de données, ainsi que notre système, sont décrits dans la section suivante. Si une grande majorité des travaux concernent le domaine médical, il est important de préciser que notre travail s'inscrit dans un contexte industriel dont l'objectif est de pouvoir brasser un éventail de domaines applicatifs varié tels que les domaines juridique, administratif etc. Un second objectif est de pouvoir réaliser une première chaîne de traitement à partir de méthodes éprouvées et existantes afin de déterminer si ces dernières peuvent s'appliquer à une dé-identification traitant plusieurs domaines et observer parallèlement quels verrous scientifiques attendent la tâche de pseudonymisation.

3 Implémentation

3.1 Données

Catégorie de données personnelles L'objectif de notre système de pseudonymisation est donc de pouvoir détecter un ensemble de catégorie de données à caractère personnel pour ensuite les substituer, tout en gardant une cohérence avec la donnée d'origine. Nous découpons nos données personnelles tels que les identifieurs : **name** pour le nom de personne physique (cette catégorie est divisée en deux sous-catégories : **first name** et **last name**), **email**, **phone** (chaque numéro de téléphone est associé à un pays ou bien à l'étiquette UNKNOWN) et **credit card**. Et les quasi-identifieurs : **date of birth** et **address** pour les adresses postales.

Jeu de données	Corpus	# documents	# données pers.
train NAME	enron + sf + conll-03	2000 + 500 + 946	8003
test NAME enron	enron	500	3562
test NAME salesforce	salesforce	500	1935
test EMAIL	enron + sf	500 + 500	3562
test PHONE	enron + SMS Spam	1000 + 500	1935
test DATE OF BIRTH	Wikidetox + CNN	297 + 10	3562
test ADDRESS	Data gouv txt	117	281

TAB. 1 – *Constitution des jeux d'entraînement et de test*

Jeux de données Afin de se constituer un ensemble de test pour chacune de nos catégories de données personnelles ainsi qu'un ensemble d'entraînement pour celles basées sur une approche par apprentissage, nous nous sommes constitués différents ensembles de données à partir de plusieurs corpus. Il est important de noter qu'IA4trust est destiné à traiter les langues française et anglaise et par conséquent les corpus choisis contiennent des documents des deux langues. L'un des principaux corpus utilisés est le corpus d'Enron, un ensemble de données de plus de 600 000 mails anglais, contenant l'entièreté des catégories de données personnelles (Klimt et Yang, 2020). 500 documents ont été utilisés pour le corpus de test. Le corpus Salesforce est un corpus de 1840 cases [3], anglais et français, qui contient également toutes les catégories excepté les cartes de crédit. 500 documents ont été utilisés pour le corpus de test. Enfin, le corpus Conll2003 (Tjong Kim Sang et De Meulder, 2003), qui est un extrait du corpus Reuter contenant des articles journalistiques, a été utilisé lors de l'entraînement et la validation du modèle NAME. Le corpus d'entraînement NAME est ainsi constitué de l'ensemble de ces trois corpus, à savoir Enron, Salesforce et Conll03. Le corpus Spam SMS Collection a permis d'enrichir le corpus de test de PHONE avec des numéros de téléphone britanniques (Almeida et al., 2013) tandis que le corpus Wikidetox Wulczyn et al. (2016) a permis de créer le corpus DATE OF BIRTH. Le corpus "Texte extrait des pdfs trouvés sur data.gouv.fr" [4] qui contient le texte extrait de 6602 PDFs comporte de nombreuses catégories comme des noms, des emails et numéros de téléphone. Il a néanmoins été utilisé uniquement pour se constituer un corpus d'entraînement et de test pour les adresses postales. Le Tableau 1 détaille l'ensemble des corpus et le nombre de données personnelles qui ont été annotées.

3.2 Architecture générale

Notre processus de détection d'entités dénommantes se décompose en plusieurs étapes : 1) un pré-traitement des données, 2) une détection d'identifieurs et quasi-identifieurs et enfin 3) leur identification qui passe par un rattachement à un individu. Si le pré-traitement s'apparente à un simple nettoyage de données, l'étape de détection se trouve au cœur de notre travail. Pour chaque catégorie de données, un module de détection est créé avec des méthodes propres à

3. selon la définition de Salesforce, une case engloble l'ensemble des retours, questions ou problèmes de clients
4. https://www.data.gouv.fr/fr/datasets/texte-provenant-des-pdfs-trouves-sur-data-gouv-fr/

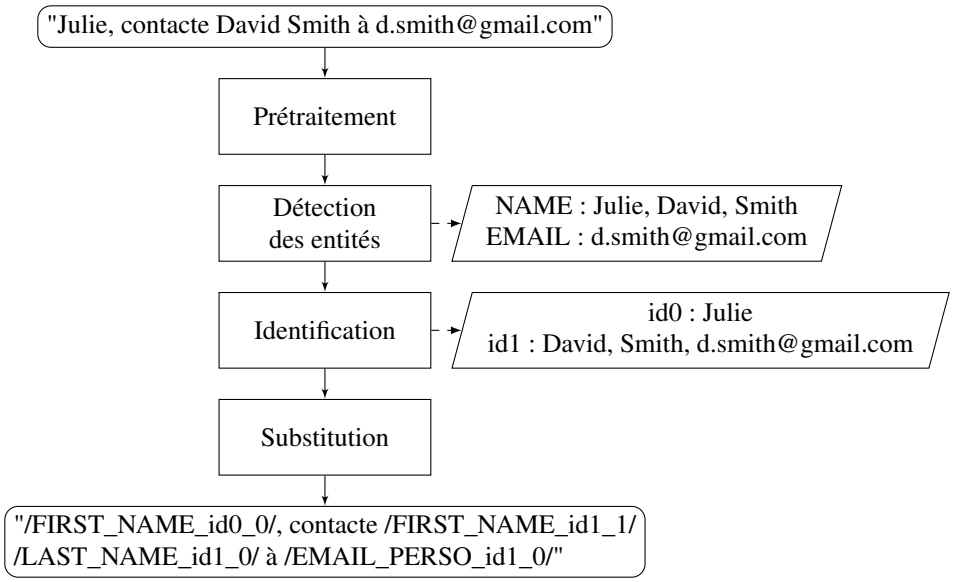


FIG. 1 – *Architecture de pseudonymisation pour un document*

chacune d'entre elles, dû à l'hétérogénéité de nos catégories de données. Un des points fondamentaux de notre approche réside dans le fait que les modules doivent être exécutés en cascade, c'est-à-dire de façon consécutive et dans un ordre précis. Cet ordonnancement résulte de deux besoins. Tout d'abord, certains modules sont dépendants d'autre module et utilisent les résultats du précédent pour détecter leur propre catégorie de données personnelles. De plus, lors de notre étape d'identification des données personnes, nous cherchons à rattacher chaque donnée à une personne physique présente dans le texte. La figure 1 résume l'architecture de notre système de pseudonymisation.

3.2.1 Détection des données personnelles

Comme évoqué précédemment, chaque catégorie de données personnelles possède un module de détection associé. Ce choix est guidé par l'hétérogénéité de nos données et leurs natures. En effet, certaines catégories demandent une approche sémantique se reposant sur le contexte tandis que d'autres suivent un format normé qui peut être alors traité via un ensemble de règles. Nous présentons ici les approches utilisées et les caractéristiques qui leur sont propres.

Transformers Le module de **détection de noms** adopte une approche d'apprentissage profond avec les Transformers BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2019). L'avantage des Transformers, comparé aux Réseaux de Neurones Récurrents, se trouve dans leur caractère parallélisable, c'est-à-dire leur capacité à traiter les données séquentielles sans qu'elles soient traitées dans l'ordre. C'est donc à la fois pour leur vitesse d'entraînement et leur capacité à apprendre sur des textes larges que nous avons favorisé les

Transformers. Plusieurs modèles de transformeurs ont été testés et sont détaillés dans la Section 5.

Le modèle retenu est le modèle Distilroberta (Sanh et al., 2019) de par ses résultats et son temps d'exécution. Il s'agit d'un modèle distillé de RoBERTa-base. Un modèle distillé est une technique de compression dans lequel un modèle apprenant est entraîné pour reproduire le comportement d'un modèle plus large ou bien un ensemble de modèles (Hinton et al., 2015). L'entraînement s'est effectué sur l'ensemble d'apprentissage Train NAME qui a été annoté suivant le standard IOB et qui utilise son tokenizer associé *distilroberta-base*. L'entraînement a été lancé avec une optimisation des hyperparamètres qui sont une longueur maximale de token de *512*, une *taille de batch de 7* et dans un maximum de *3 epochs*. L'entraînement fait appel à AdamW, une méthode d'optimisation stochastique qui optimise Adam et remplace sa L2 réguliarisation en découplant le weight decay de l'étape d'optimisation. Ainsi, le weight decay apparaît dans la mise à jour du gradient et devient indépendant du learning rate. Le learning rate est fixé à *0.00003*.

Une fois que le modèle a identifié les noms dans le document, un post-traitement est effectué afin de les catégoriser en FIRST NAME ou LAST NAME. Ce post-traitement s'appuie notamment sur des listes de prénoms et noms, à la fois anglais [5] et français [6].

Approche à base de règles Plusieurs modules de détection font appel à une approche à base de règles, à savoir les modules d'emails, de numéro de téléphone, de carte de crédit et de date de naissance. Ces identifieurs et quasi-identifieurs possèdent une norme plus ou moins fixe que des méthodes tels que les expressions régulières peuvent détecter.

Ainsi, le **module de détection d'emails** effectue une première détection via le système de Reconnaissance d'Entités Nommées de Spacy (Honnibal et al., 2020). Bien que ce dernier offre une première détection acceptable, l'utilisation d'expressions régulières permet de récupérer les adresses mails au format plus complexe qui ont été ignorées par le NER Spacy et également pour pouvoir distinguer les adresses professionnelles et personnelles. De plus, le module de détection d'emails utilise la liste des noms détectés par le module de noms pour pouvoir rattacher une adresse email à un individu dans des cas ambigus. Un algorithme va créer toutes les combinaisons possibles entre les prénoms et noms de chaque personne, créant une liste d'adresses candidates. Cela permet de lever l'ambiguïté pour des cas où une adresse mail affiche le nom d'une personne de manière partielle comme "julie_d@gmail.com" et qu'il existerait "Julie Dupont" et "Julie Martin" dans le texte.

Le **module de détection de numéros de téléphone** repose également sur des expressions régulières mais passe d'abord par une étape de détection de langue via le module de détection *spacy-langdetect* [7] afin de différencier les pays partageant les mêmes formats de numéro de téléphone.

Enfin, le **module de détection de cartes de crédit** détecte dans un premier temps les numéros suivant leur format puis confirme leur validité via l'algorithme de Luhn (Luhn, 1954).

Le **module de date de naissance**, quant à lui, va également détecter l'ensemble des dates de naissance via NER Spacy mais va par la suite étudier le voisinage (c'est-à-dire une fenêtre de 15 tokens) de chaque date identifiée pour confirmer s'il s'agit d'une date générique ou une

5. https://www.kaggle.com/kaggle/us-baby-names
6. https://www.insee.fr/fr/statistiques/3536630
7. https://spacy.io/universe/project/spacy-langdetect

date de naissance grâce à la détection de mots clés liés à la naissance (e.g "born", "birthdate", "naissance"...).

CRF Les Condition Random Field (CRFs) sont des méthodes statistiques proposées par Lafferty et al. (2001), très utilisées dans la tâche de reconnaissance d'entités nommées et adaptées aux séquences d'unités linguistiques. Leur principe revient à déterminer la probabilité jointe d'une séquence de labels, en se basant sur une séquence observée. Puisque les **adresses postales** suivent un format de séquence de sous-catégories d'adresse (numéro de rue, nom de rue, code postal...), nous avons utilisé CRFSuite pour les détecter. Chaque token d'adresse a été annoté en sous-catégorie et un ensemble de traits a été créé : mot en minuscule, type de mot, la casse, si le mot est un nombre, un code postal, une ville ou un pays, une distribution (Boite Postale, Code Spécial) un mot commun et si les voyelles contenues dans le mot. La détection des villes et pays se fait à l'aide de dictionnaires dédiés. Les mots voisins ($n-1$, $n+1$ et $n+2$) sont également passés en tant que traits. Le CRF utilise l'algorithme d'apprentissage L-BFGS, avec respectivement un coefficient de régularisation L1 et L2 de 0.11215 et 0.00482, trouvés après une recherche optimale des hyperparamètres.

3.2.2 Identification des données personnelles

Une fois notre étape de détection d'entités dénommantes, nous procédons à un traitement qui consiste à associer à chaque donnée personnelle un identifiant. Cet identifiant comporte deux-sous parties : a) un identifiant de personne idPerson, à savoir l'identifiant d'un individu présent dans le texte et dont l'entité dénommante lui est associée et b) un identifiant unique de la donnée au sein de sa catégorie idData. Ainsi, si une donnée personnelle est associée à un individu, elle sera identifiée sous la forme **/{catégorie}_{idPerson}/**, sinon on indiquera que la donnée n'est reliée à personne avec **/{catégorie}_{idx}/**. Cette association d'identifiants dépend également de l'ordonnancement des modules. En effet, l'anonymisation des NAME est exécutée en premier de sorte à ce que le reste des identifieurs et quasi-identifieurs puissent être reliées à un individu et son id. Ainsi, si l'on reprend l'exemple de la Figure 1, "David" et "Smith" sont substitués par respectivement **/FIRST_NAME_id1/** et **/LAST_NAME_id1/**, ce qui permet de relier l'adresse mail "d.smith@gmail" à David Smith et donc à l'identifiant 1, donnant ainsi **/EMAIL_PERSO_id1/**.

Les autres identifieurs et quasi-identifieurs, qui ne peuvent être reliés en se basant seulement sur les tokens, sont reliés à un individu de deux façons. La première consiste à détecter la présence d'une signature au sein du document à l'aide de la librairie Talon [8]. Si un identifieur ou un quasi-identifieur apparaît dans une signature en même temps qu'un nom, alors ces derniers sont reliés. Sinon, nous créons alors des regroupements de données personnelles, à savoir des ensembles de données personnelles se trouvant à proximité, c'est-à-dire chacun séparé par moins de 15 tokens [9]. Nous partons ensuite du principe que ces derniers ne peuvent être reliés à un individu que lorsqu'ils se trouvent dans un groupement de données personnelles ne contenant qu'un seul individu (i.e qu'un seul identifiant de personne) et au moins un identifieur NAME pour les identifieurs et au moins un couple de NAME et EMAIL pour les quasi-identifieurs. Cette différence s'explique par le fait que les quasi-identifieurs, qui ne

8. https://github.com/mailgun/talon
9. cette distance a été définie de manière empirique

peuvent théoriquement pas identifier une personne de façon directe, ont besoin de plus d'informations pour être reliés à une personne physique.
Enfin, la dé-identification se conclue par une étape de substitution. À l'heure actuelle, cette étape consiste simplement à attribuer l'identifiant idData, c'est-à-dire un identifiant unique de la donnée selon sa catégorie de donnée personnelle. Cet identifiant permet notamment de rattacher plusieurs données distinctes mais de même catégorie à un individu (e.g différents numéros de téléphone pour un individu). Le substitut final pour chaque entité dénommante sera donc de la forme : **/{catégorie}_{idPerson}_{idData}/**. Néanmoins, cette méthode de substitution est susceptible d'évoluer par la suite pour d'autres approches telles que la généralisation ou la substitution aléatoire.

Ainsi, un document donné sera pseudonymisé de la façon suivante :

Bonjour Julie et Gabriel, J'ai appelé l'administrateur, vous pouvez le contacter au 04 01 02 80 35. Je dois peut-être faire une session de prise en main à distance dans l'après-midi. Je suis preneur de toute idée/procédure qui pourrait aider à solutionner ou identifier le problème. On en saura plus à ce moment-là. Christophe LECORDIER christophe.lecordier@atempo.com 71, rue de la Victoire Téléphone : +33 3 72 74 24 47 Mobile : +33 6 10 44 45 30	Bonjour **/FIRST_NAME_id0_0/** et **/FIRST_NAME_id1_1/**, J'ai appelé l'administrateur, vous pouvez le contacter au **/PHONE_FRANCE_idx_0/**. Je dois peut-être faire une session de prise en main à distance dans l'après-midi. Je suis preneur de toute idée/procédure qui pourrait aider à solutionner ou identifier le problème. On en saura plus à ce moment-là. **/FIRST_NAME_id2_2/** **/LAST_NAME_id2_0/** **/EMAIL_PRO_id2_0/** **/ADDRESS_id2_0/** Téléphone : **/PHONE_FRANCE_id2_1/** Mobile : **/PHONE_FRANCE_id2_2/**

où l'on observe que l'adresse postale est bien reliée à Christophe LECORDIER puisqu'elle se trouve dans un regroupement de données personnelles comportant au moins un nom et une adresse mail. À l'inverse, le numéro de téléphone situé en début de document n'est relié à personne dû à son appartenance à un regroupement contenant deux individus identifiés, ce qui rend la tâche d'association impossible à l'heure actuelle.

4 Résultats

Cette section Résultats présente les résultats de l'étape détection des données personnelles, selon chaque catégorie. Il est à noter que nous évaluons ici l'étape de détection individuellement et que les étapes d'identification et de substitution ne sont pas prises en compte. L'évaluation des données personnelles se présente sous la forme d'un comparaison de chaque token détecté comme entité dénommante en récupérant leur position dans le texte anonymisé avec les positions des entités dénommantes présentes dans le texte gold. Le Tableau 2 contient les résultats finaux de chacune des catégories, évalués sur leur ensemble de test correspondant. De manière générale, les modules affichent un meilleur rappel comparé à la précision, ce que nous privilégions puisque dans notre cadre de pseudonymisation, un faux positif est moins

handicapant qu'un faux négatif (il vaut mieux anonymiser une donnée non-identifiante qu'en laisser passer). Seul le module NAME, testé sur le corpus NAME Salesforce possède un rappel plus bas. Cette différence peut s'expliquer par le fait que le modèle NAME est entraîné sur des données majoritairement anglaises (bien que le corpus d'entraînement contienne des données françaises, elles sont minoritaires) et est testé ici sur un corpus bilingue français/anglais. Les modules PHONE et DATE OF BIRTH proposent tous les deux des résultats extrêmement élevés avec un f-score de respectivement 0.99 et 0.98, qu'il faut néanmoins nuancer : le module PHONE est évalué avec les numéros de téléphone indifférenciés, c'est-à-dire sans l'attribution de pays. Le module DATE OF BIRTH, lui, jouit de performances élevées dû à son nombre bas de données évaluées. Enfin, le module ADDRESS produit les performances les plus basses avec un f-score de 0.83, bien qu'il soit important de noter que contrairement au reste des catégories, il est bien plus difficile de délimiter le début et la fin d'une adresse postale. C'est pourquoi nous rencontrons beaucoup de cas où une partie d'adresse est correctement anonymisée mais pas la suite, augmentant ainsi à la fois les faux positifs et les faux négatifs.

Le Tableau 3 compare les différentes approches qui ont été testées. Si le module NER de Spacy est bien en deça des résultats attendus pour le module NAME, le CRF lui parvient à atteindre un rappel acceptable de 0.93. Néanmoins, sa précision chute à 0.80, de par son manque de contexte. C'est pourquoi les Transformers proposent des meilleurs résultats, grâce à leur nature non-séquentielle, traitant les phrases en tant que tout plutôt que mot par mot. À l'inverse, les modules EMAIL Spacy et PHONE CRF ont une précision parfaite de 1.00 contre des rappels bien plus bas. Le rappel du module EMAIL Spacy peut s'expliquer par le fait que le corpus de test possède des formats d'emails atypiques que Spacy n'a pas pris en compte mais que nous avons pu récupérer via les expressions régulières. Néanmoins, la détection de Spacy reste nécessaire car elle permet de gagner 0.1 points en rappel et précision, comparé à l'utilisation des regex seule. Il est difficile de comparer nos résultats à ceux de la littérature étant donné que nous ne travaillons par sur les mêmes données. On remarquera toutefois que le niveau global de performances que nous observons est semblable à celui de Grouin et Zweigenbaum (2014) sur le français médical, ce qui suggère une certaine généricité de notre chaîne de traitement par rapport au domaine de spécialité étudié. Plus intéressant, (Grouin et Zweigenbaum, 2014) rendent compte de variations de performances par type d'entité proche de nos observations. En effet, ces derniers retrouvent de très bons résultats pour les catégories NAME, PHONE et DATE (resp. un f-score de 0.905, 1.00 et 0.966) et un score significativement plus bas pour la catégorie ADDRESS (0.222). En anglais, les résultats de (Liu et al., 2017) convergent dans la même direction avec des résultats pour les catégories NAME, CONTACT (qui regroupe les numéros de téléphone et email), DATE et LOCATION de respectivement 0.947, 0.965, 0.980 et 0.886.

5 Discussion et conclusion

Dans ce papier, nous avons présenté notre première version de notre système de détection d'entité dénommantes dont l'objectif final vise à offrir une pseudonymisation de documents numérisés. Cette tâche est particulièrement présente dans le domaine médical, cependant ces travaux se concentrent exclusivement sur des données et dossiers cliniques tandis que notre système vise à traiter un plus large éventail de type de documents comme des emails, factures, contrats, compte rendus juridiques... Un point clé de notre travail est la prise en compte de

Module	**Précision**	**# Rappel**	**F-score**.
NAME ENRON	0.97	0.95	0.96
NAME SALESFORCE	0.91	0.94	0.92
EMAIL	0.99	0.96	0.98
PHONE	0.99	0.99	0.99
DATE OF BIRTH	0.99	0.98	0.98
ADDRESS	0.84	0.81	0.83

TAB. 2 – *Résultats des modules de détection*

Module	**Précision**	**# Rappel**	**F-score**.
NAME SPACY	0.78	0.76	0.77
NAME CRF	0.93	0.80	0.86
NAME TRANSFORMERS	0.97	0.95	**0.96**
EMAIL SPACY	0.40	1.00	0.60
EMAIL REGEX	0.98	0.95	0.96
EMAIL SPACY + REGEX	0.99	0.96	**0.98**
PHONE CRF	0.67	1.00	0.81
PHONE REGEX	0.99	0.99	**0.99**

TAB. 3 – *Comparaison de différentes approches par module*

chaque spécificité de nos identifieurs et quasi-identifieurs afin de choisir l'approche la plus adéquate. Cette démarche s'aligne sur la stratégie d'hybridation des travaux de dé-identification du domaine médicale mais se démarque de la majorité des travaux de pseudonymisation qui utilisent un seul modèle pour l'entièreté de leurs données. Nous soutenons que cette approche ne peut permettre de capter toutes les subtilités des catégories et que l'hybridation de nos approches permet d'atteindre de hautes performances. Il faudra néanmoins confirmer ces résultats sur de nouveaux corpus de test pour évaluer la capacité de généralisation du système. Une étude sur la généricité des modèles permettra également d'évaluer l'impact des domaines et de leur nature structurée ou non-structurée. De plus, contrairement au milieu médical qui reste un milieu fermé/interne, nous devons répondre à des besoins industriels d'interprétabilité. Si de nombreuses techniques d'explicabilité émergent au niveau des réseaux de neurones et que nous devrons en faire usage pour notre module NAME, nous considérons pour le moment que notre approche par hybridation offre une certaine transparence tout en permettant de la performance. Cette tâche d'identification d'entités dénommantes correspond donc à la première étape de notre chaîne de pseudonymisation dont l'étape ultérieure sera d'empêcher toute ré-identification. Cette ré-identification peut potentiellement passer par un recroisement de quasi-identifiants tel que l'a démontré Sweeney (2002) et des techniques comme le k-anonymat (Sweeney, 2002).

Concernant les perspectives futures, nous souhaitons ajouter une brique de résolution de coréférence afin de pouvoir lier chaque mention d'un individu entre elles. Si le rattachement de chaque donnée à un identifiant d'individu peut être vu comme un début de résolution de coréférence, elle concerne uniquement les reprises directes (c'est-à-dire reprises avec la même tête lexicale) et exclut les reprises indirectes ou pronominales. Cela permettra ainsi d'identifier chaque personne présente dans le texte et de lever l'ambiguïté de certains cas. À notre connaissance, seuls Adams et al. (2019) ont envisagé cette approche, néanmoins ces derniers n'ont pas présenté de résultats associés.

Deux autres perspectives sont envisagées. La première consiste à élargir notre ensemble d'entités dénommantes qui pour le moment comprend des entités classiques. Nous comptons ajouter des types tels que les organisations ou les professions (c'est-à-dire "l'administrateur" présent dans le premier exemple). Les faibles résultats de Liu et al. (2017) pour la catégorie profession prouvent que cette catégorie pose un challenge non-négligeable. Enfin, l'entièreté de notre processus d'évaluation s'est effectuée sur des ensembles de tests où chaque module est testé de façon individuelle. Nous avons conscience que cela ne permet pas d'évaluer le système dans sa globalité et de ce fait, nous avons pour projet de constituer un corpus de test avec chaque catégorie de données personnelles annotée dans un document, mettant ainsi en lumière des erreurs potentielles.

Références

Adams, A., E. Aili, D. Aioanei, R. Jonsson, L. Mickelsson, D. Mikmekova, F. Roberts, J. F. Valencia, et R. Wechsler (2019). AnonyMate: A toolkit for anonymizing unstructured chat data. *Proceedings of the Workshop on NLP and Pseudonymisation*, 1–7.

Almeida, T., J. Gómez Hidalgo, et T. Silva (2013). Towards sms spam filtering: Results under a new dataset. *International Journal of Information Security Science (IJISS) 2(1)*, 1–18.

Devlin, J., M.-W. Chang, K. Lee, et K. Toutanova (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. *International Journal of Information Security Science (IJISS))*.

Eshkol, I. (2010). Entrer dans l'anonymat. Etude des "entités dénommantes" dans un corpus oral. In *Eigennamen in der gesprochenen Sprache*, pp. 245–266. Francke Verlag.

Grouin, C. et P. Zweigenbaum (2014). De-identification of clinical notes in french: towards a protocol for reference corpus development. *Journal of biomedical informatics 50*, 151–161.

Gupta, D., M. Saul, et J. Gilbertson (2004). Evaluation of a deidentification (de-id) software engine to share pathology reports and clinical documents for research. *Am J Clin Pathol 121(2)*, 176–86.

Hara, K. (2006). Applying a svm based chunker and a text classifier to the deid challenge. *i2b2 Workshop on Challenges in Natural Language Processing for Clinical Data*, 10–1.

Hinton, G. E., O. Vinyals, et J. Dean (2015). Distilling the knowledge in a neural network.

Honnibal, M., I. Montani, S. Van Landeghem, et A. Boyd (2020). spaCy: Industrial-strength Natural Language Processing in Python.

Klimt, B. et Y. Yang (2020). The enron corpus: a new dataset for email classification research. *ECML'04: Proceedings of the 15th European Conference on Machine Learning*, 217–226.

Lafferty, J. D., A. McCallum, et F. C. N. Pereira (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, San Francisco, CA, USA, pp. 282–289. Morgan Kaufmann Publishers Inc.

Liu, Z., B. Tang, X. Wang, et Q. Chen (2017). De-identification of clinical notes via recurrent neural network and conditional random field. *Journal Biomed Inform.*.

Luhn, H. P. (1954). Computer for verifying numbers. *United States Patent and Trademark Office 2(1)*.

Sanh, V., L. Debut, J. Chaumond, et T. Wolf (2019). Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv abs/1910.01108*.

Stubbs, A., M. Filannino, et Uzuner (2017). De-identification of psychiatric intake records: Overview of 2016 cegs n-grid shared tasks track 1. *Journal Biomed Inform.*.

Sweeney, L. (1996). Replacing personally-identifying information in medical records, the scrub system. *Cimino, JJ, ed. Proceedings, Journal of the American Medical Informatics Assoc. Washington, DC: Hanley Belfus*, 333–337.

Sweeney, L. (2002). k-anonymity: A model for protecting privacy. *IEEE Security and Privacy Magazine 10(5)*, 1–14.

Szarvas, G., R. Farkas, et A. Kocsor (2006). A multilingual named entity recognition system using boosting and c4.5 decision tree learning algorithms. *International Conference on Discovery Science 2006*, 267–278.

Tjong Kim Sang, E. F. et F. De Meulder (2003). Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, 142–147.

Wulczyn, E., N. Thain, et L. Dixon (2016). Wikipedia detox.

Yang, H. et J. Garibaldi (2015). Automatic detection of protected health information from clinic narratives. *Journal Biomed Inform.*.

Summary

Despite growing concerns about data privacy, large amounts of data are still shared in our daily lives including personal data which can be used to identify an individual's identity. In this paper, we present a first version of AI4trust: a pseudonymisation system for documents that aims to meet the requirements of the GDPR whilst producing a pseudonymised document as semantically coherent as possible. First, we de-identify a set of personal data categories (name, email, address…). We compare various approaches depending on the entity type and argue that the use of different de-identification modules is mandatory. We emphasize the importance of the order in which the modules are executed, especially for linking a personal data to a person.

Reconnaissance d'entités d'intérêt dans les signatures d'e-mails à partir d'un jeu de données authentique

Nihed Bendahman*, Kevin Cousot*, Cédric Lopez*

*Cap Oméga, Rond-point Benjamin Franklin CS 39521, 34960 Montpellier
prénom.nom@emvista.com,
http://www.emvista.com

Résumé. Dans un contexte industriel, les courriers électroniques constituent une source importante de renseignements. Quantité d'informations utiles sont présentes dans les méta-données et le corps du message mais également dans les signatures. On pourra notamment y trouver des entités d'intérêt telles que des noms de personnes et d'organisations, des numéros de téléphone ou encore des URLs. Une fois structurées, on peut alors tirer profit de ces signatures, par exemple en enrichissant une application de gestion de relations clients ou à des fins d'anonymisation. Dans cet article, nous proposons le premier jeu de données de signatures annoté manuellement en entités d'intérêt. Nous fournissons également une baseline pour la reconnaissance d'entités d'intérêt à l'aide de différents classifieurs.

1 Introduction

Les courriers électroniques sont une source d'informations et de connaissances de premier ordre pour les professionnels. Leur contenu textuel peut être exploité automatiquement pour générer un historique de la connaissance qu'ils contiennent (Matta et al., 2014), mettre en évidence la résolution de problèmes récurrents (Francois et al., 2016) ou identifier des tâches pour aider les utilisateurs à gérer leur temps (Khosravi et Wilks, 1999).

Un e-mail consiste en un en-tête et un corps de message. L'en-tête, structuré, contient l'émetteur et le récepteur, le sujet, la date, l'adresse de réponse *etc*. Le corps, non structuré, est un texte libre comprenant le message ainsi que la signature de l'auteur. Il n'existe pas de jeux de données publics et accessibles à la communauté, ce qui peut expliquer le peu de travaux publiés à ce sujet, et ces derniers utilisent des jeux de données privés rendant ainsi impossible la reproductibilité des résultats.

La tâche d'analyse des signatures est similaire à la reconnaissance d'entités nommées (REN) définie lors de la 6ème conférence MUC [1]. D'un point de vue technique, la principale différence consiste en ce que la REN est généralement appliquée à des textes formés d'une séquence de termes en contexte et entretenant des relations syntaxiques tandis que les signatures sont constituées d'une séquence d'informations portant sur l'émetteur et non corrélées entre

1. https://www-nlpir.nist.gov/related_projects/muc/

elles avec très peu de dépendances syntaxiques (certaines dépendances peuvent exister dans le cas d'une fonction, par exemple dans le syntagme nominal "directeur général des ventes").

Nos contributions dans cet article sont les suivantes :

1. mise en évidence de l'absence de travaux sur l'analyse des signatures d'e-mails ;
2. création du premier jeu de données annoté manuellement pour cette tâche (600 signatures) ;
3. production d'une *baseline*.

2 Travaux précédents

Au niveau international, la structuration des courriers électroniques est un sujet important de recherche depuis le début des années 2000. L'intérêt pour l'analyse et l'exploitation de ce type de texte se renforce avec la publication du corpus Enron (Klimt et Yang, 2004). La recherche porte alors principalement sur la classification d'e-mails (Kiritchenko et Matwin, 2001; Youn et McLeod, 2007; Sonbhadra et al., 2020), la reconnaissance du spam (Kumar, 2020) ou de tentative de *fishing* (Verma et al., 2012; Egozi et Verma, 2018; Kumar et al., 2020), l'identification de demandes/tâches (Lampert et al., 2008) ainsi que la réponse aux courriels (Al-Alwani, 2015).

La majorité des travaux aborde la classification du contenu des e-mails en segmentant le texte en lignes, parfois en blocs (Lampert et al., 2009; Bacchelli et al., 2012; Yin et al., 2011). Ainsi, Chen et al. (1999); Carvalho et Cohen (2004) et Estival (2008) cherchent à détecter la présence d'un bloc de signatures dans un courriel en appliquant des algorithmes d'apprentissage séquentiels et non-séquentiels. Carvalho et Cohen (2004) atteignent une *accuracy* de 99%. Chen et al. (1999) procède à une analyse linguistique au sein des blocs de signatures à l'aide de transducteurs à états finis pondérés et sur la base de contraintes lexicales et grammaticales.

Il est intéressant de noter que les travaux visant à analyser de manière fine le contenu des e-mails, c'est-à-dire au niveau des *tokens*, sont rares. C'est particulièrement le cas pour l'analyse des signatures d'e-mails qui semble être ignorée dans les travaux précédents. Par exemple, dans Li et al. (2015), l'objectif est d'extraire les entités des e-mails à partir des signatures qui sont considérées comme du bruit pour la tâche définie. À notre connaissance, aucune expérience n'a été décrite pour l'analyse fine des signatures d'e-mails, faute d'un jeu de données manuellement annoté.

3 Jeu de données et typologie

Face à l'absence de données de référence, nous avons mis en place un formulaire Web permettant le "don" de signatures. L'URL de ce formulaire a été partagée via des listes de diffusion, la plupart d'entre elles s'adressant à des Français. Une étude future devrait porter sur la corrélation entre cet ensemble de données et la zone géographique des donateurs. Par exemple, les codes postaux anglais sont alphanumériques alors que les codes postaux français sont numériques. Néanmoins, les entités présentes dans les signatures sont rarement traduisibles (nom des personnes, nom des villes, codes postaux, numéro de téléphone, etc.) et une partie des signatures est souvent en anglais (notamment le rôle de la personne). Ce jeu de données ne

contient que des signatures en texte brut (pas de HTML, pas d'images). Nous avons collecté 600 signatures sur une durée de 6 mois.

Pour préserver l'identité des contributeurs, chaque entité d'intérêt a été remplacée par une entité équivalente afin d'assurer la cohérence du jeu de données. Par exemple, un prénom féminin a été remplacé par un autre prénom féminin. Notons également que les fonctions des personnes n'ont pas été remplacées, sauf dans le cas où la description permettait d'identifier la personne. Il est important de noter également que nous préservons la taille de chaque entité remplacée, la présence de majuscules et de minuscules ainsi que la ponctuation. Un exemple est donné ci-dessous :

Signature :

Mary Margho
Directeur Général
Téléphone : +33.(0)1.52.62.32.65
6 Rue Jean-Paul Montagne
75001 Paris
www.saveprogramma.com

Signature pseudonymisée :

Anna Dupont
Directeur Général
Téléphone : +33.(0)1.55.52.12.96
72 Rue Paul-Marie L'abbé
75001 Paris
www.anywaythewall.com

Au total, nous avons identifié 13 types d'entités : personne, lieu, organisation (y compris leurs divisions), fonction (*i.e.* le poste de la personne), nom du projet (*e.* "Direction stratégique du marketing"), CEDEX (*i.e.* "Courrier d'Entreprise à Distribution EXceptionnelle" ou "business mail with special delivery"), CS (*i.e.* "Course Spéciale" un service de distribution de courrier spécifique), code postal, nom des réseaux sociaux, référence de l'utilisateur (par exemple un *login* ou un compte Twitter), l'URL, le numéro de téléphone et l'adresse e-mail.

Le processus d'annotation s'est déroulé en trois étapes successives :

1. La première étape visait à rédiger le guide d'annotation. Celui-ci a été défini par deux annotateurs qui sont parmi les auteurs de cet article. En particulier, chaque type d'entité a été défini et des signatures pseudonymisées ont été sélectionnées dans le jeu de données de développement pour illustrer les définitions ;
2. La deuxième étape visait à calculer l'accord inter-annotateur : un sous-ensemble de 50 signatures sélectionnées au hasard dans l'ensemble des données (600 signatures) a été donné aux annotateurs. La guide d'annotation a été appliquée sans aucune forme de coopération entre les annotateurs pendant le processus d'annotation. Le coefficient kappa de Cohen était de 0,96, ce qui indique un accord presque parfait. Par conséquent, aucune modification n'a été apportée au guide d'annotation ;
3. Enfin, la dernière étape a consisté à annoter les 550 signatures restantes (*cf.* exemple en Fig. 2). Comme pour la deuxième étape, nous avons utilisé la plateforme collaborative INCEpTION (Klie et al., 2018) mais nous n'utilisons pas le mode "apprentissage actif" afin d'éviter tout biais lié à l'influence d'une pré-annotation automatique.

Finalement, le jeu de données contient $4\,518$ annotations. La signature la plus courte contient 2 tokens (un prénom et un nom) et la plus longue en contient 132. En moyenne, une signature contient 46 tokens. Les entités les plus fréquentes sont les lieux (21,4%), les numéros de téléphone (15,1%) et les organisations (13,8%). La distribution des types d'entités est illustrée dans la figure 1 qui met en évidence le déséquilibre des classes.

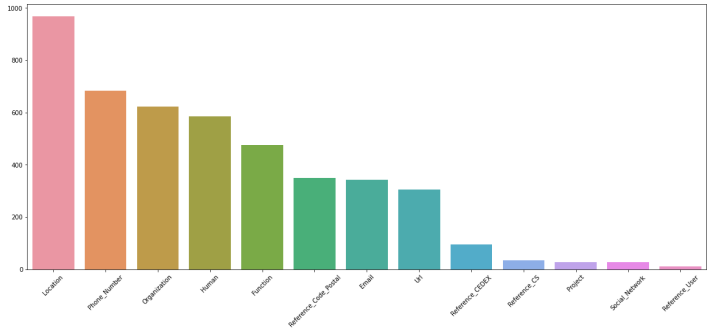


FIG. 1 – *Nombre d'occurrences par type d'entité*

FIG. 2 – *Capture d'écran d'une signature annotée dans la plateforme Inception.*

L'ensemble des données est mis à la disposition de la communauté scientifique. [2]

4 Expériences

Nos expériences visent à développer et à comparer des classifieurs de *tokens* selon les 13 types d'entités prédéfinis (*cf.* section 3). Sur la base du jeu de données de 600 signatures, nous avons utilisé et comparé quatre systèmes d'apprentissage automatique courants pour la REN : Yadav et Bethard (2018) :

— SVM avec la fonction de base radiale (RBF) comme noyau (Cortes et Vapnik, 1995), une des méthodes de prédiction les plus robustes ;
— CRF (Lafferty et al., 2001) afin de prendre en compte le contexte alors que le classifieur prédit une étiquette pour un seul échantillon ;
— Bi-LSTM (Cui et al., 2018) qui améliore le contexte dont dispose l'algorithme et prend en considération les mots qui suivent et précèdent immédiatement un mot dans un texte ;
— BERT (Devlin et al., 2018) qui bénéficie des *transformers* (Vaswani et al.).

Notre objectif est de proposer des résultats de référence, une *baseline*, pour de futurs travaux. Nous avons divisé le jeu de données en un jeu d'entraînement et un jeu de test avec 80%

2. https://www.emvista.com/recherche/#publication

et 20% respectivement. Dans les expériences suivantes, la tokenisation a été effectuée par la fonction NLTK *word_tokenize* avec le paramètre *preserve_line=true*. Une étape cruciale est la définition des caractéristiques utilisées par l'algorithme. Les modèles que nous avons développés ne réagissent pas tous de la même manière au même ensemble de caractéristiques. Par conséquent, nous avons adapté empiriquement cet ensemble à chaque algorithme comme présenté dans les sous-sections suivantes.

4.1 SVM

Les SVM sont des modèles d'apprentissage supervisé qui sont généralement adaptés aux petits ensembles de données. Les descripteurs utilisés sont présentés en Table 1.

Caractéristique	**Description**
is_digit	Indique si le token est complètement numérique.
is_alpha	Indique si le token est complètement alphabétique.
is_upper	Indique si le token est complètement en minuscule.
is_lower	Indique si le token est complètement en majuscule.
length_token	La longueur du tokens.
contain_@	Indique si le token contient le caratère "@".
contain_http ://	Indique si le token contient les caratères "http ://".
contain_digit	Indique si le token contient des chiffres.
position_token	La position du token dans la ligne.
is_in_first_line	Indique si le tokens est dans la première ligne de la signature.
is_in_intermediaire_line	Indique si le tokens est dans une ligne intermédiaire de la signature.
is_in_last_line	Indique si le tokens est dans la dernière ligne de la signature.
begin_of_signature	Indique si le tokens est le premier token de la signature.
end_of_signature	Indique si le tokens est le dernier token de la signature.

TAB. 1 – *Caractéristiques sélectionnées pour l'entraînement du SVM*

4.2 CRF

Notre modèle CRF utilise les descripteurs présentés en Table 2.

L'algorithme d'apprentissage utilisé est L-BFGS, et, en ce qui concerne les hyper-paramètres, nous avons effectué une recherche aléatoire et obtenu la configuration suivante : paramètre de régularisation $c = 0.1$ et $max\_iter \in [128, 200]$. L'évaluation est effectuée à l'aide d'une validation croisée à 3 blocs.

4.3 Bi-LSTM

Le Bi-LSTM est un modèle que nous considérons dans ces expériences car il a la capacité d'apprendre les dépendances à distance et de calculer les vecteurs de représentation du contexte pour chaque mot. Nous utilisons les hyper-paramètres suivants pour entraîner le modèle :

— batch_size : 32

Caractéristique	Description
is_digit	Indique si le token est complètement numérique.
is_alpha	Indique si le token est complètement alphabétique.
is_upper	Indique si le token est complètement en minuscule.
is_lower	Indique si le token est complètement en majuscule.
word_upper	La version minuscule du token
word[:3]	les trois premières lettres du token.
word[-3 :]	les trois dernières lettres du token.
contain_@	Indique si le token contient le caratère "@".
contain_http ://	Indique si le token contient les caratères "http ://".
contain_digit	Indique si le token contient des chiffres.
is_in_first_line	Indique si le tokens est dans la première ligne de la signature.
is_in_intermediaire_line	Indique si le tokens est dans une ligne intermédiaire de la signature.
is_in_last_line	Indique si le tokens est dans la dernière ligne de la signature.
begin_of_signature	Indique si le tokens est le premier token de la signature.
end_of_signature	Indique si le tokens est le dernier token de la signature.

TAB. 2 – *Features sélectionnées pour l'entraînement du CRF.*

— epochs : 60
— learning_rate : 0.0005
— optimizer : adam (nous avons également expérimenté *adamW* mais les résultats étaient moins bons)

4.4 BERT

Les signatures de courriel sont difficilement comparables à des textes en langage naturel, en particulier sur les aspects morphosyntaxiques et syntaxiques. Par exemple, il n'y a pas de verbe ni d'adverbe et il n'y a ni sujet ni objet dans notre jeu de données. Récemment, des modèles de langage pré-entraînés ont obtenu de très bons résultats en reconnaissance d'entités nommées appliquée à des textes en langue naturelle. Même si les signatures sont structurellement différentes des textes habituels en langue naturelle, nous avons ajusté un modèle de langage pré-entraîné, à savoir la base multilingue BERT, afin d'observer son comportement sur ce type de données.

Pour affiner le modèle, nous avons fixé les hyper-paramètres suivants :

— batch_size : 16
— epochs : 30
— learning_rate : 2e-05
— optimizer : adamW

5 Résultats

Le jeu de test contient 100 signatures (20% de l'ensemble du jeu de données). Il a été utilisé pour évaluer les 4 modèles présentés dans la section précédente. Pour chaque modèle (CRF, Bi-LSTM, SVM et BERT) nous avons commencé l'entraînement avec 100 signatures et nous avons ajouté des signatures par bloc de 100 afin d'observer l'évolution des performances du modèle. Les figures 3, 4 et 5 présentent respectivement la précision, le rappel et le F-score.

Un comportement similaire est observé quelle que soit la mesure : comme prévu, l'ajout de données améliore les résultats. Le CRF obtient les meilleures performances (*cf.* Fig. 3). Il atteint $0,80$ pour la précision, $0,78$ pour le rappel et $0,79$ pour le F-Score. Il est intéressant de noter que le Bi-LSTM a une croissance plus forte que les autres modèles lorsqu'on ajoute les 100 dernières signatures. De nouveaux exemples dans l'ensemble d'entraînement sont nécessaires pour déterminer si le Bi-LSTM peut surpasser le CRF avec les paramètres donnés.

Modèles	Précision	Rappel	F-Score
Bert	0.49	0.21	0.29
Bi-LSTM	0.66	0.61	0.63
CRF	**0.8**	**0.78**	**0.79**
SVM	0.72	0.67	0.69

TAB. 3 – *Précision, Rappel and F-Score pour chaque modèle*

Le tableau 4 indique les scores obtenus par le meilleur modèle (CRF) pour chaque type d'entité. Il apparaît que des classes bien équilibrées peuvent obtenir des scores très éloignés, ce qui suggère que certains types sont plus difficiles à apprendre que d'autres. Par exemple, *Organization* (521 occurrences) et *Human* (488 occurrences) obtiennent respectivement $0,53$ et $0,84$ sur l'ensemble de test.

Le modèle CRF a par la suite été amélioré en intégrant des descripteurs booléens à base d'expressions régulières, précisément pour les CS, Cedex, Postal Code, Email, Phone Number, URL. Par ailleurs, des lexiques de termes ont été mis en place pour rechercher leur présence parmi les tokens dans le but d'identifier les types Social Network, Project et Function. Certains de ces descripteurs sont dépendantes du jeu de données (Cedex, CS, codes postaux et numéros de téléphone français). Les types cités ci-avant sont par conséquent mieux reconnus et permettent de lever l'ambiguïté sur les autres types sémantiques. Par exemple, le type Organization augmente son F-score de 0,02%, Location de 0,05%, Human de 0,07%. Les résultats obtenus après réajustement sont présentés en Table 5.

6 Conclusion

Même si les e-mails sont un important vecteur de communication dans le monde entier, il semble qu'aucune étude n'ait porté sur l'extraction d'entités à partir de signatures. Nous avons donc développé le premier jeu de données de signatures de référence annotées par des humains et nous l'avons mis à la disposition de la communauté scientifique. Sur la base de ce jeu de données, nous fournissons une baseline avec des classifieurs SVM, CRF, Bi-LSTM et BERT.

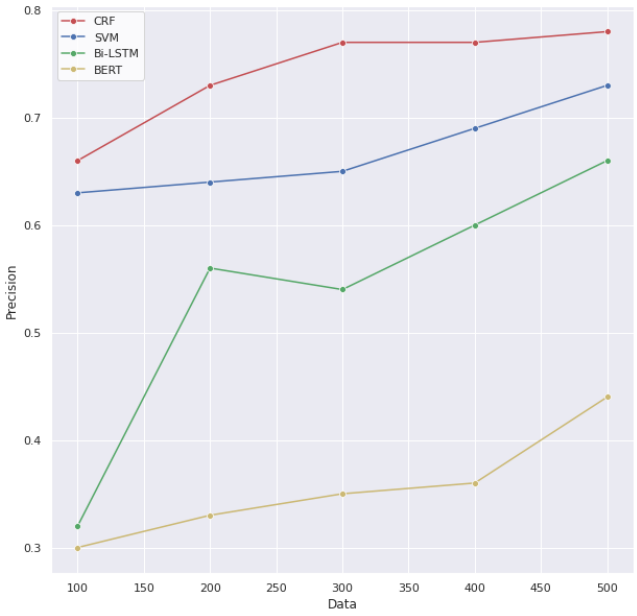


FIG. 3 – *Précision de 100 à 500 signatures dans le je de données de dev.*

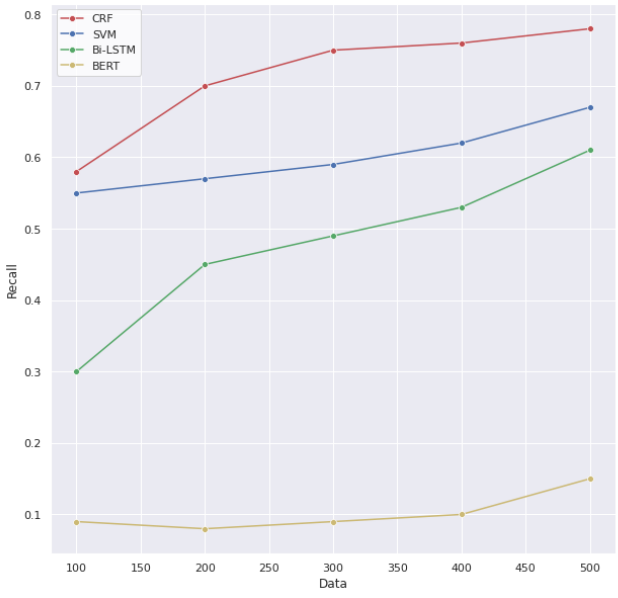


FIG. 4 – *Rappel de 100 à 500 signatures dans le jeu de données de dev.*

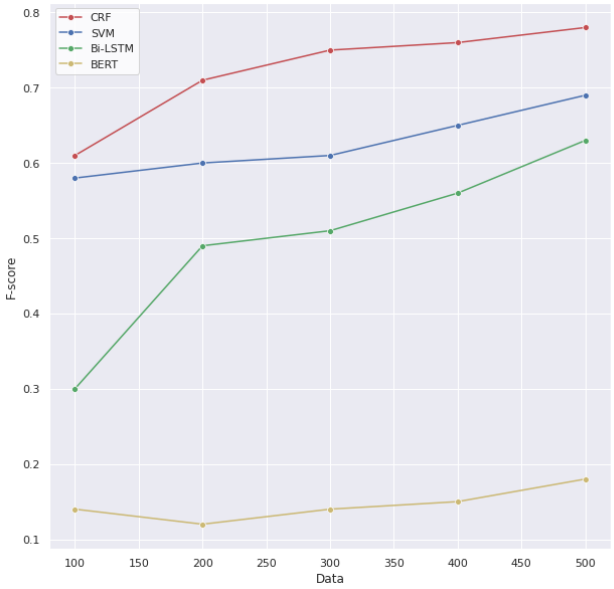


FIG. 5 – *F-score de 100 à 500 signatures dans le jeu de données de dev.*

Il apparaît que la tâche n'est pas triviale : les meilleurs résultats sont de $0,77\%$ F-score avec le CRF.

Les résultats obtenus indiquent que l'ajout de nouvelles signatures annotées dans l'ensemble d'entraînement pourrait améliorer significativement les résultats. Nous prévoyons d'expérimenter d'autres classifieurs et d'étendre l'ensemble d'entraînement (en particulier avec des techniques d'augmentation des données) afin d'améliorer les résultats et d'étudier l'impact sur les performances des classifieurs.

Nous prévoyons enfin de construire un nouveau jeu de données en suivant le même processus mais avec des signatures d'autres nationalités afin d'étendre notre typologie et d'améliorer la généricité des modèles.

Références

Al-Alwani, A. (2015). Improving email response in an email management system using natural language processing based probabilistic methods. *Journal of Computer Science 11*(1), 109.

Bacchelli, A., T. Dal Sasso, M. D'Ambros, et M. Lanza (2012). Content classification of development emails. In *2012 34th International Conference on Software Engineering (ICSE)*, pp. 375–385. IEEE.

Carvalho, V. R. et W. W. Cohen (2004). Learning to extract signature and reply lines from email. In *Proceedings of the Conference on Email and Anti-Spam*, Volume 2004.

	dev			test		
Entity types	P	R	F	P	R	F
Location	0.87	0.87	0.87	0.77	0.83	0.79
Phone number	0.95	0.98	0.96	0.94	0.95	0.94
Organization	0.64	0.71	0.67	0.55	0.52	0.53
Human	0.85	0.9	0.88	0.83	0.88	0.84
Function	0.71	0.69	0.7	0.64	0.62	0.62
Postal Code	0.92	0.94	0.93	0.88	0.85	0.86
Email	0.96	0.98	0.97	0.89	0.9	0.89
URL	0.91	0.7	0.79	0.83	0.64	0.72
CEDEX	0.95	0.97	0.96	0.94	0.74	0.82
CS	0.88	0.78	0.83	0.8	0.5	0.61
Project	1	0.13	0.24	0.45	0.4	0.42
Social network	0.86	0.33	0.47	0.94	0.33	0.48
Reference user	1	0.5	0.67	0.7	0.5	0.58

TAB. 4 – *Précision, Rappel and F-score pour chaque type d'entité avec le CRF*

Chen, H., J. Hu, et R. W. Sproat (1999). Integrating geometrical and linguistic analysis for email signature block parsing. *ACM Transactions on Information Systems (TOIS) 17*(4), 343–366.

Cortes, C. et V. Vapnik (1995). Support-vector networks. *Machine learning 20*(3), 273–297.

Cui, Z., R. Ke, Z. Pu, et Y. Wang (2018). Deep bidirectional and unidirectional lstm recurrent neural network for network-wide traffic speed prediction. *arXiv preprint arXiv :1801.02143*.

Devlin, J., M.-W. Chang, K. Lee, et K. Toutanova (2018). Bert : Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv :1810.04805*.

Egozi, G. et R. Verma (2018). Phishing email detection using robust nlp techniques. In *2018 IEEE International Conference on Data Mining Workshops (ICDMW)*, pp. 7–12. IEEE.

Estival, D. (2008). Author attribution with email messages. *Journal of Science, Vietnam National University 1*, 1–9.

Francois, R., M. Nada, et A. Hassan (2016). Ktr : an approach that supports knowledge extraction from design interactions. *IFAC-PapersOnLine 49*(12), 473–478.

Khosravi, H. et Y. Wilks (1999). Routing email automatically by purpose not topic. *Natural Language Engineering 5*(3), 237–250.

Kiritchenko, S. et S. Matwin (2001). Email classification with co-training. In *Proceedings of the 2001 conference of the Centre for Advanced Studies on Collaborative research*, pp. 8. Citeseer.

Klie, J.-C., M. Bugert, B. Boullosa, R. E. de Castilho, et I. Gurevych (2018). The inception platform : Machine-assisted and knowledge-oriented interactive annotation. In *Proceedings of the 27th International Conference on Computational Linguistics : System Demonstrations*, pp. 5–9.

Entity types	Précision	Rappel	F-Score
Location	0.82	0.85	0.84
Phone number	0.96	0.98	0.97
Organization	0.53	0.57	0.55
Human	0.87	0.96	0.91
Function	0.66	0.61	0.63
Postal Code	0.88	0.97	0.92
Email	0.98	0.99	0.98
URL	0.91	0.88	0.9
CEDEX	0.94	0.75	0.83
CS	0.8	0.5	0.61
Project	0.45	0.4	0.42
Social network	1	0.56	0.71
Reference user	0.75	0.5	0.58

TAB. 5 – *Précision, Rappel and F-score pour chaque type d'entité avec la prise en compte de descripteurs à base d'expression régulières*

Klimt, B. et Y. Yang (2004). The enron corpus : A new dataset for email classification research. In *European Conference on Machine Learning*, pp. 217–226. Springer.

Kumar, A., J. M. Chatterjee, et V. G. Díaz (2020). A novel hybrid approach of svm combined with nlp and probabilistic neural network for email phishing. *International Journal of Electrical and Computer Engineering 10*(1), 486.

Kumar, P. (2020). Predictive analytics for spam email classification using machine learning techniques. *International Journal of Computer Applications in Technology 64*(3), 282–296.

Lafferty, J., A. McCallum, et F. C. Pereira (2001). Conditional random fields : Probabilistic models for segmenting and labeling sequence data. *Proceedings of ICML 2001*, 282–289.

Lampert, A., R. Dale, et C. Paris (2009). Segmenting email message text into zones. In *Proceedings of the 2009 conference on empirical methods in natural language processing*, pp. 919–928.

Lampert, A., R. Dale, C. Paris, et al. (2008). The nature of requests and commitments in email messages. In *Proceedings of the AAAI Workshop on Enhanced Messaging*, pp. 42–47.

Li, J., S. Sen, et N. Zaman (2015). Entity extraction from business emails. *International Journal of Information Technology and Computer Science 7*(9), 15–22.

Matta, N., H. Atifi, et F. Rauscher (2014). Knowledge extraction from professional e-mails. In *IFIP International Workshop on Artificial Intelligence for Knowledge Management*, pp. 43–57. Springer.

Sonbhadra, S. K., S. Agarwal, M. Syafrullah, et K. Adiyarta (2020). Email classification via intention-based segmentation. In *2020 7th International Conference on Electrical Engineering, Computer Sciences and Informatics (EECSI)*, pp. 38–44. IEEE.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, et I. Polosukhin. Attention is all you need.

Verma, R., N. Shashidhar, et N. Hossain (2012). Detecting phishing emails the natural language way. In *European Symposium on Research in Computer Security*, pp. 824–841. Springer.

Yadav, V. et S. Bethard (2018). A survey on recent advances in named entity recognition from deep learning models. In *Proceedings of the 27th International Conference on Computational Linguistics*, Santa Fe, New Mexico, USA, pp. 2145–2158. Association for Computational Linguistics.

Yin, M., J. Luo, D. Cao, X. Liu, et M. Li (2011). Automatically locating salutation and signature blocks in emails. In *2011 Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, Volume 3, pp. 1783–1787. IEEE.

Youn, S. et D. McLeod (2007). A comparative study for email classification. In *Advances and innovations in systems, computing sciences and software engineering*, pp. 387–391. Springer.

Summary

In the industrial context, emails are an important source of information. Valuable information can usually be found in the metadata, in the body of the message, even in the signatures. In this article, we focus on signatures that contain key information to identify the interlocutor. Our objective is to develop a model able to identify entities of interest within signatures contents, such as names of persons, organizations, phone numbers, or URL. Such a structured information is useful for many applications, for instance to enrich a Customer Relationship Management tool (CRM), or for an anonymization purpose. In this paper, we describe the typology and the annotation process we applied to develop the first email signature dataset, then we propose a first baseline from classifiers based on SVM, CRF, Bi-LSTM, and BERT.

Analyse automatique d'émotions pour l'optimisation de campagnes d'e-mails en français

Alexis Blandin*,***, Farida Said**
Jeanne Villaneau***
Pierre-François Marteau***

*UNEEK Kosmopoelead 168 route de Saint Joseph 44300 Nantes
alexis.blandin@kosmopolead.com,
https://www.kosmopolead.com/
**Université Bretagne Sud, CNRS UMR 6205, LMBA, F-56000 Vannes, France
{farida.said}@univ-ubs.fr
https://http://lmba-math.fr
***IRISA, Université Bretagne Sud, Rue André Lwoff, 56000 Vannes
{alexis.blandin,jeanne.villaneau,pierre-francois.marteau}@univ-ubs.fr
https://http://lmba-math.fr

Résumé. Les campagnes d'e-mail sont stratégiques pour optimiser la relation entre une entreprise et ses clients. Ainsi, éviter que les informations et sollicitations soient ignorées ou considérées comme une nuisance apparaît comme une nécessité. Dans cet article, nous nous intéressons au texte des *newsletters* et aux émotions et opinions qu'il exprime. Plus particulièrement, nous cherchons à évaluer comment ces dernières peuvent influer sur les indices de performance (taux d'ouverture et taux de clics) des campagnes de *newsletters*. Nous décrivons d'abord le jeu de données en français que nous avons constitué. Ensuite, nous explorons les paramètres et les plongements (*embeddings*) représentant les émotions du contenu textuel pour estimer l'importance de leur relation avec les performances de la campagne. Nous cherchons enfin si nos représentations émotionnelles du texte permettent de prédire les performance des campagnes.

1 Introduction

L'intelligence artificielle (IA) est devenue un outil stratégique d'aide à la décision dans le domaine du marketing. Dans cet article, nous exploitons plus particulièrement les techniques du traitement de la langue (TAL) pour étudier comment optimiser la rédaction de *newsletters*. Nous avons concentré nos études sur la manière dont les émotions transmises au sein d'une campagne d'*emailing* peuvent influer sur ses performances. Ainsi, nos travaux se situent à la croisée de trois domaines de recherche : les études de marketing, les analyses d'*e-mails* et la détection d'émotions (*sentiment analysis*) (cf. section 2). Pour ce faire, nous avons construit un jeu de données de plus de 900 *newsletters* en français venant de différentes structures ; il est présenté dans

la section 3.1. Nous nous sommes appuyés sur les techniques d'analyse de sentiment pour proposer et comparer deux représentations vectorielles des émotions qui émanent du texte de ces *newsletters* (section 3). Nous pouvons ainsi étudier statistiquement la relation entre ces représentations des émotions et des indicateurs de performances des *newsletters* tels que le taux de clics ou le taux d'ouverture. Nous avons enfin étudié dans quelle mesure nos représentations vectorielles permettent une prédiction de la performance des *newsletters* basée exclusivement sur cette analyse de sentiment (section 4).

2 Travaux connexes

Cette section présente les travaux relatifs à nos objectifs dans les trois domaines connexes concernés, le Marketing, le TAL et la représentation des émotions et l'apprentissage automatique.

2.1 Études Marketing

Il est probable que les émotions ne soient pas nécessairement le critère le plus influent sur l'impact de la réception d'une *newsletter* et le comportement du receveur. Ainsi un envoi trop intensif de *newsletters* dans un court laps de temps, l'utilisation de certains mots, ou encore une longueur trop importante peuvent faire que la *newsletter* soit vue comme une nuisance ou un *spam*. Cependant, si l'on constate que les communications par e-mail prennent plus d'ampleur dans notre quotidien, elles ont aussi tendance à devenir plus informelles et donc à véhiculer davantage d'émotions. Il est donc intéressant de voir dans quelle mesure ces émotions influent sur la performance des campagnes d'e-mail.

Certaines hypothèses proposées par des études de marketing peuvent s'appliquer à notre problème d'impact des émotions dans les communications par *e-mail*. Par exemple, une étude proposée par Byron (2008) suggère que le manque d'interactions face à face dû au contexte des communications par *e-mail*, peut donner lieu à une mauvaise interprétation des émotions.

Selon l'auteur, le manque d'indices sur les intentions des émotions de l'envoyeur, induit le receveur de l'*e-mail* en erreur. Cette incompréhension menant bien souvent à un *effet de neutralité* ou même à un *effet de négativité*. L'auteur précise que, faute d'indices suffisants sur les émotions venant de l'auteur du mail, les émotions négatives dans le texte seraient plus saillantes. Ainsi un sarcasme, par exemple, pourrait être moins bien perçu par *e-mail* que dans une relation face à face. Cette étude met de plus l'accent sur le contexte social au sein de l'entreprise pour expliquer cet effet (genre, âge, hiérarchie, etc.) Cependant, l'auteur pointe aussi des conséquences positives à cet effet, comme la non-nécessité de chercher à embellir les *e-mails* d'émotions positives qui ne seraient pas, ou mal perçues. Ce biais de négativité pourrait donc poser problème dans un contexte de *newsletters*, où transmettre des émotions positives comme la fierté (Kim et Johnson 2013) semble crucial pour obtenir des attitudes positives de la part des consommateurs, et encore plus dans les cultures occidentales.

De plus, le fait que, lors des campagnes d'*e-mail*, envoyeur et receveur ne se connaissent pas, accentue encore davantage la méconnaissance de certains éléments du contexte et

peut, par conséquent, mener d'autant plus facilement vers une mauvaise interprétation des émotions. Enfin, une étude récente de Virginie Rodriguez (2021) portant sur les conduites de *newsletters* pendant la pandémie du COVID-19 tend à montrer que les communications commerciales par *e-mail* ne sont plus simplement informatives mais qu'elles contiennent également aujourd'hui des éléments divertissants et avec un ton moins formel. Cela peut se traduire par des newsletters plus personnelles, et moins uniquement informatives. Or, ce type de discours aura plus tendance à faire intervenir des émotions pour susciter de l'intérêt. Ainsi l'attention portée à l'expression des émotions devient un élément essentiel dans la communication des entreprises pour éviter d'éventuelles erreurs d'interprétation.

2.2 Analyse d'*e-mails*

L'impact des émotions sur les performances de campagne d'*e-mails* a été étudié par Miller et Charles (2016), qui ont validé plusieurs hypothèses sur la manière dont les émotions issues de l'objet d'un mail peuvent influencer sa perception. Les auteurs ont travaillé sur le jeu de données de Enron (Klimt et Yang 2004), composé d'*e-mails* rédigés en anglais et issus de communications internes à une même entreprise. L'absence de telles ressources en langue française (cf. Kalitvianski (2018) et Guenoune et al. (2020)) nous a contraints à construire notre propre jeu de données. Par ailleurs, notre étude prend en compte l'objet de la *newsletter*, et aussi son contenu.

2.3 Détection d'émotions

Récemment, la détection d'émotions dans un texte est devenue de plus en plus populaire par son grand nombre d'applications. Cette approche se veut plus fine que l'analyse d'opinion, et propose d'évaluer le texte selon tout un spectre d'émotions. Si beaucoup d'études proposent leur propre définition de ce spectre (Seyeditabari et al. 2018), on peut toutefois retenir les 6 émotions de base proposées par Ekman (1999) (joie, peur, colère, tristesse, surprise et dégoût) comme une bonne base de travail. De plus si de nombreuses méthodes existent pour détecter les émotions en langue anglaise, les ressources sont beaucoup moins abondantes en langue française. Ainsi notre état de l'art se concentre sur ces ressources-là, et nous avons adapté les modèles destinés à analyser de l'anglais pour analyser du français si besoin.

2.3.1 Approche par lexique

L'une des méthodes pour détecter les émotions est d'utiliser un lexique de mots ayant ou non une connotation avec chacune d'elles. Un tel lexique existe en français, celui de Abdaoui et al. (2017), obtenu par traduction du lexique NRC-EmoLex de l'anglais vers le français. Ce lexique a ensuite été enrichi et validé par des traducteurs professionnels, pour un total de plus de 10 000 lemmes ; il permet entre autres de réaliser une analyse des émotions par sacs de mots. Cette approche est détaillée dans une précédente étude (Blandin et al. (2021)).

Analyse automatique d'émotions pour l'optimisation de campagnes d'e-mails en français.

2.3.2 Approche par modèle entraîné

L'approche précédente a cependant ses limites ; en effet, faute de prendre en compte le contexte du mot au sein d'une phrase, elle peut mener à de mauvaises interprétations, voire à des contresens. Une autre approche consiste à entraîner un modèle pour attribuer automatiquement une émotion à une phrase. Par exemple, en utilisant un modèle auto-encodeur multilingue, on peut, par ré-apprentissage, l'entraîner à détecter automatiquement si une phrase donnée renvoie à une certaine émotion. Dans cet esprit, nous avons traduit en français le jeu de données proposé par (Saravia et al. 2018) constitué de phrases en anglais et des émotions qui leur sont attribuées. Les auteurs ont généré un jeu de données labellisé en utilisant un algorithme semi-supervisé basé sur une approche par graphes enrichie à l'aide d'un plongement de mots. Leur méthode d'enrichissement de données semble très performante par rapport à l'état de l'art, et leur jeu de données comporte 20 000 phrases associées chacune à une émotion. Les auteurs suggèrent d'utiliser ce jeu de données pour ré-entraîner un auto-encodeur d'architecture T5 (Raffel et al. 2019).

Dans leurs travaux, (Saravia et al. 2018) associent à des phrases en anglais une émotion parmi les 6 émotions d'Eckman décrites précédemment, à l'exception du *dégoût*, et avec l'*amour* comme nouvelle catégorie. Nous avons traduit ces phrases en français à l'aide d'un traducteur automatique pour ensuite réaliser un ré-apprentissage. Pour ce faire, nous avons utilisé *Google translate* dont une étude récente, Aiken (2019), relève les bonnes performances pour la traduction de l'anglais vers le français. Par ailleurs, notre apprentissage se fait sur des données issues de tweets donc assez courtes pour espérer que la traduction n'engendre pas trop d'erreurs. Les performances du modèle ainsi obtenu figurent dans le tableau 1.

TAB. 1 – *Résultat du modèle T5 à l'issue d'un ré-apprentissage (fine tuning) visant à prédire des émotions sur des données traduites en français, et la différence de performance par rapport au modèle originel en anglais.*

Emotion	**Précision**	**Rappel**	**F1-score**	**Support**
Colère	0,72 *(-22%)*	0,64 *(-29%)*	0,68 *(-25%)*	275
Peur	0,73 *(-13%)*	0,66 *(-26%)*	0,69 *(-20%)*	224
Joie	0,80 *(-17%)*	0,85 *(-8%)*	0,83*(-12%)*	695
Amour	0,55 *(-24%)*	0,29 *(-60%)*	0,38*(-46%)*	159
Tristesse	0,74 *(-23%)*	0,84*(-12%)*	0,79*(-18%)*	581
Surprise	0,69*(-6%)*	0,71*(-3%)*	0,70*(-5%)*	66
Exactitude			0,75*(-18%)*	2000
Moyenne	0,70*(-18%)*	0,67*(-13%)*	0,68*(-21%)*	2000
Moyenne pondérée	0,74*(-19%)*	0,75*(-18%)*	0,74*(-19%)*	2000

On observe que la précision du modèle chute en moyenne d'environ 20% par rapport à la référence du modèle originel en anglais, avec des différences notables en fonction des émotions. Ainsi, si le modèle perd jusqu'à près de 50% de ses performances pour

détecter l'émotion *amour*, la perte n'est que de 5% pour la *surprise*. On peut voir plus en détail les résultats des tests de prédiction avec la matrice de confusion en figure 1.

FIG. 1 – *Matrice de confusion des prédictions du modèle T5 ré-entraînés pour la détection d'émotions en français.*

Il apparaît, que si effectivement la prédiction de l'émotion *amour* est mauvaise, cela est surtout dû à la confusion du modèle avec la *joie* et, dans une moindre mesure, la *tristesse*. Sans faire de conclusions trop hâtives sur l'interprétation des émotions par le modèle, on peut toutefois remarquer que *joie* et *amour* sont les deux seules émotions *positives* de l'ensemble du panel d'émotions.

3 Contexte et présentation des données

Ces recherches s'inscrivent dans un but pratique qui vise à aider à rédiger des *newsletters*. Un applicatif de ce que nous présentons ici est la conception d'un outil d'aide à la rédaction de *newsletters*. Le scénario d'utilisation d'un tel outil est le suivant : lorsque l'utilisateur édite une *newsletter*, il dispose, grâce à une analyse du contenu, d'une prédiction de la performance future de sa *newsletter*. Les informations analysées peuvent être diverses mais ici, comme précisé précédemment, nous ne nous intéressons qu'à l'impact (réel ou supposé) des émotions d'un texte, en utilisant pour cela les méthodes de détection d'émotions et d'opinions décrites et testées précédemment. Plus généralement, aujourd'hui les communications par *e-mail* deviennent de plus en

plus présentes dans notre quotidien, et moins formelles (Virginie Rodriguez (2021)), l'objectif ici est d'étudier comment grâce à des outils de traitement du langage nous pouvons aider à éviter de commettre des impairs de communication en transmettant des émotions qui peuvent pénaliser le taux d'ouverture ou de clics, qui sont dans notre contexte les deux seuls indicateurs de performances quantifiables puisque nous étudions essentiellement des *newsletters* à but non commercial.

3.1 Présentation du jeu de données

Notre jeu de données se compose de *newsletters* envoyées par diverses organisations comme des entreprises, des écoles, ou des associations. Ces organisations utilisent le système de relation client, et rédigent leurs *newsletters* dans un environnement similaire. Elles sont destinées à informer leurs abonnés et n'ont a priori aucun objectif commercial, comme celui qui consisterait à rediriger leur lecteur vers une plate-forme d'achat.

La performance d'une *newsletter* peut être mesurée en comptant le nombre d'ouvertures uniques par *e-mail* envoyé, et le nombre de clics générés par *e-mail* ouvert. Ces indicateurs sont couramment utilisés pour l'analyse d'*e-mail* comme dans Kumar (2021) ou Bonfrer et Drèze (2009). On peut estimer que le taux d'ouverture permet de mesurer l'attractivité d'une *newsletter*, et le taux de clics son taux d'engagement. Une fois notre jeu de données nettoyé des *newsletters* indésirables issues du CRM (doublons, tests, etc.), nous obtenons un ensemble de 973 *newsletters*, chacune envoyée à plusieurs inscrits, et pour chacune d'elle leurs indicateurs de performance décrits précédemment. Nous avons obtenu nos données en compilant les campagnes venant de plusieurs clients. Si la répartition des *newsletters* dans notre jeu de données selon leur origine n'est pas homogène, comme on le voit dans la figure 2, on peut toutefois faire l'hypothèse que cela n'induit pas de biais dans les données en raison de la similitude importante entre les différentes structures et les objectifs de leurs envois de *newsletters*. En effet nous nous intéressons ici à des caractéristiques qui sont *a priori* indépendantes de l'origine de l'*e-mail*.

3.2 Résultats statistiques sur les descripteurs

Nous avons représenté notre jeu de données à l'aide de plusieurs caractéristiques émotionnelles. Deux d'entre elles, la polarité et la subjectivité ont été obtenues à l'aide de l'outil consacré de la bibliothèque *Python TextBlob for Natural Language Processing* détaillée par Yaqub et al. (2017), qui utilise un modèle pré-entraîné pour associer ces deux scores à des phrases. Les autres caractéristiques sont mesurées soit en utilisant le lexique FEEL et une approche par sac de mots, soit en utilisant notre modèle ré-appris pour la détection d'émotions. Pour ces valeurs nous avons uniquement considéré le contenu du mail, l'objet du mail s'étant avéré trop court pour apporter une information significative. On peut aussi noter certaines particularités observées lors de l'application du modèle T5 au jeu de données. On constate que la joie est l'émotion la plus souvent détectée, certaines *newsletters* ayant jusqu'à 80% de leur phrases labellisées comme "joyeuses". Bien que la joie puisse figurer le ton par défaut dans une communication commerciale, cette sur-représentation de la joie est imputable à l'absence d'un label

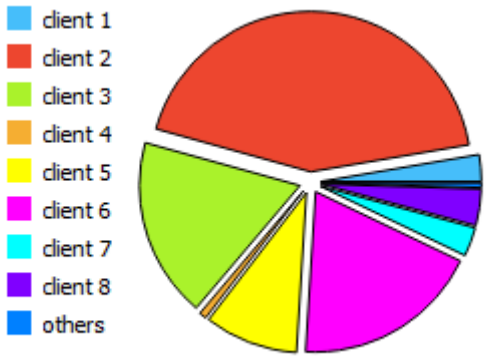


FIG. 2 – *Distribution des newsletters par client du CRM.*

neutre dans le modèle, ce qui est un défaut de ce modèle, la plupart des études sur la détection d'émotions ayant un label "neutre".

À l'inverse, dans l'ensemble de notre jeu de données, aucune phrase n'est labellisée avec l'émotion *colère* ; ce qui peut aisément se concevoir dans un contexte de communication commerciale.

On peut aussi remarquer que la détection de *amour* est corrélée négativement avec le taux de clics. Cependant, le modèle étant peu précis pour la détection de cette émotion en particulier, nous ne conclurons pas sur l'impact du ton employé par l'auteur de l'*e-mail*.

Enfin, et surtout, on peut remarquer que, hormis la joie détectée par T5, les émotions sont négativement associées aux taux de clics générés. Les corrélations entre les différentes caractéristiques extraites et les indicateurs de performance figurent dans le tableau 2.

D'une manière générale, les caractéristiques issues du modèle T5 entraîné sur le français ont une corrélation moindre avec le taux de clics que les caractéristiques issues de l'approche par sac de mots.

4 Prédiction de performances

Cette section concerne la prédiction de l'intérêt suscité auprès des lecteurs des *newsletters* à partir de leurs caractéristiques émotionnelles. Le taux de clics permet de partager *a priori* l'ensemble des *newsletters* en deux classes, les "bonnes" et les "mauvaises", de même cardinalité, pour éviter d'avoir une classe sur-représentée par rapport à l'autre. À cette fin, nous avons effectué des classifications de notre ensemble de données avec diverses méthodes d'apprentissage automatique (Mueller et Guido 2018). Le tableau 3 présente les mesures de performance estimées par la procédure de validation croisée portant sur un partitionnement en 10 sous-ensembles. Les différents

TAB. 2 – *Corrélation de Pearson entre les caractéristiques des newsletters et leurs performances, suivant les deux approches : par lexique (en noir) et avec le modèle T5 réappris (en bleu)*

Caractéristique	Taux d'ouverture	Taux de clics	
Taille du fichier	**-0.14*** **	**0.25*** **	
Longueur de l'objet	**-0.13*** **	**0.18*** **	
Polarité de l'objet	-0.07**	-0.03^{ns}	
Subjectivité de l'objet	-0.01^{ns}	-0.07*	
Polarité du contenu	-	0.09**	
Subjectivité du contenu	-	-0.07*	
Joie dans le contenu	-	-0.10**	0.06*
Peur dans le contenu	-	**-0.11*** **	-0.02^{ns}
Tristesse dans le contenu	-	**-0.23*** **	-0.06^{ns}
Colère dans le contenu	-	0.06^{ns}	-
Surprise dans le contenu	-	**-0.11*** **	-0.01^{ns}
Dégoût/Amour dans le contenu	-	-0.07*	**-0.12*** **

*p-value < .05, **p-value < .01, ***p-value < .001,^{ns} not significant

types de caractéristiques considérées dans nos différents plongements sont listés ci-dessous :

1. **FEEL** - 6 émotions : joie, peur, colère, surprise, tristesse et dégoût ; issues de notre approche par lexique calculées sur l'ensemble du contenu du mail.
2. **T5_Fr** - 6 émotions : joie, peur, colère, surprise, tristesse et amour ; issues de notre approche utilisant un modèle réappris pour la détection d'émotions en français, et appliqué sur l'ensemble du contenu du mail.
3. **Subj&Pol** : score de polarité et subjectivité calculé sur l'ensemble du mail via une méthode automatisée issue d'un modèle entraîné pour l'analyse d'opinion.
4. **Objet** : score de polarité et subjectivité calculé sur l'objet du mail uniquement via une méthode automatisée issue d'un modèle entraîné pour l'analyse d'opinion.

TAB. 3 – *F1-Score des différents classifieurs exploitant les différents plongements.*

Embedding	AdaBoost	RNN	Rand. Forest	kNN	Naive Bayes	SVM	Rég Log
FEEL	0,718	0,702	0,711	0,690	0,656	0,621	0,643
FEEL+Subj&Pol	0,721	0,711	0,711	0,679	0,666	0,621	0,628
FEEL+Subj&Pol+Objet	0,724	0,716	0,713	0,680	0,666	0,607	0,585
T5_Fr	0,697	0,651	0,689	0,676	0,641	0,620	0,616
T5_Fr+Subj&Pol	0,697	0,651	0,689	0,676	0,641	0,620	0,616
T5_Fr+Subj&Pol+Objet	0,693	0,656	0,687	0,657	0,642	0,609	0,610
FEEL+T5_Fr	0,732	0,704	0,722	0,693	0,657	0,651	0,615
FEEL+T5_Fr+Subj&Pol	**0,737**	0,712	0,720	0,702	0,666	0,630	0,629
FEEL+T5_Fr+Subj&Pol+Objet	0,733	0,713	0,721	0,686	0,664	0,637	0,623

Les différentes méthodes de classification évaluées conduisent à des F1-score assez comparables. Cela peut s'expliquer par les corrélations plutôt faibles entre certaines des caractéristiques considérées (polarité de l'objet, amour dans le contenu, etc.) et le taux de clics ; ceci semble limiter la performance globale quelque soit l'approche considérée. Néanmoins nous pouvons remarquer plusieurs choses à l'issue de ces tests.

Ainsi, il apparaît que les méthodes *AdaBoost*, *réseaux de neurones* et *forêts aléatoires* sont les plus performantes pour notre tâche. De plus, bien que plus naïve, l'approche par sac de mots et lexique d'émotions permet d'obtenir une représentation d'émotions plus pertinente pour notre tâche que celle obtenue via le modèle pré-entraîné. On peut arguer que le passage par la traduction augmente le taux d'erreur du modèle T5 en français en produisant des *embeddings* moins précis et donc moins pertinents ; ce qui vient s'ajouter au fait que le modèle réalise des prédictions discrètes (0 ou 1) et associe une émotion unique à chaque phrase. Il aurait sans doute été plus efficace de prédire pour chaque émotion un indice de confiance compris entre 0 et 1 et par suite, d'assigner plusieurs émotions à une même phrase.

Cependant, on peut aussi noter une bonne amélioration en fusionnant les deux approches pour obtenir, par simple concaténation, des plongements qui cumulent les informations apportées par chacune d'elles. En revanche, si le cumul des représentations d'émotions améliore la qualité des prédictions, la prise en compte de la subjectivité et de la polarité de l'objet du mail semble la réduire.

5 Conclusion

Dans cet article, nous avons étudié l'apport d'une détection automatique d'émotion et d'opinion dans le but d'aider à la prédiction des performances de *newsletters*. La littérature dans le domaine des études marketing suggère que la communication par *e-mail* tend à renforcer les mauvaises interprétations des émotions, à cause d'un biais de négativité ou de neutralité. Lorsque le receveur du mail est un potentiel client ou un abonné, cet effet négatif peut être observé par le biais de mesures objectives telles que le taux d'ouverture ou le taux de clics généré par la *newsletter*.

Nous avons présenté un jeu de données composé de *newsletters* en français, auquel nous associons un *embedding* d'émotions selon différentes méthodes. Le premier repose sur l'utilisation d'un lexique associant des lemmes à des émotions, en exploitant une méthode par sac de mots. L'autre est basé sur un auto-encodeur réappris afin d'assigner une émotion à une phrase en français.

Notre étude montre que la plupart des émotions sont corrélées négativement avec les performances des *newsletters*, ce qui soutient l'hypothèse selon laquelle les émotions, qu'elles soient positives ou négatives, tendent à être interprétées négativement.

Enfin nous avons utilisé des *embeddings* d'émotions et de sentiments pour prédire les performances des *newsletters*. Nous avons présenté différentes approches qui se différencient par la nature de l'*embedding* exploité. Il ressort de notre étude qu'une détection d'émotions combinant une approche par sac de mots et par un modèle ré-appris, permet de prédire les classes de performances plus efficacement. En intégrant des approches plus poussées pour la caractérisation d'émotions, nous espérons encore améliorer la prédiction de performance des *newsletters*. Ainsi une perspective future pour améliorer les

résultats sur cette tâche, serait de choisir des émotions ou des méthodes de détections d'émotions qui soient plus en rapport avec notre contexte. Des pistes possibles seraient d'améliorer ou de modifier la méthode d'extraction des émotions contenues dans le texte, et d'enrichir notre représentation des données en prenant en compte d'autres caractéristiques de la *newsletter* (mise en page, couleur, etc.).

De plus, dans la mesure où la législation et les propriétaires de ces données nous y autorisent, nous partagerons notre jeu de données afin d'enrichir les ressources mises à disposition de la communauté scientifique, en général rares pour la langue française, et permettre aux chercheurs intéressés de reproduire ou d'améliorer nos résultats.

Références

Abdaoui, A., J. Azé, S. Bringay, et P. Poncelet (2017). FEEL : a French Expanded Emotion Lexicon. Language Resources and Evaluation 51(3), 833–855. Accessed on Sep. 3, 2021.

Aiken, M. (2019). An updated evaluation of google translate accuracy. Studies in Linguistics and Literature 3, p253.

Blandin, A., F. Saïd, J. Villaneau, et P.-F. Marteau (2021). Automatic emotions analysis for french email campaigns optimization. In CENTRIC 2021.

Bonfrer, A. et X. Drèze (2009). Real-time evaluation of e-mail campaign performance. Marketing Science 28(2), 251–263. Accessed on Aug. 22, 2021.

Byron, K. (2008). Carrying too heavy a load ? the communication and miscommunication of emotion by email. The Academy of Management Review 33(2), 309–327. Accessed on Aug. 18, 2021.

Ekman, P. (1999). Basic Emotions, Chapter 3, pp. 45–60. John Wiley & Sons, Ltd. Accessed on Aug. 21, 2021.

Guenoune, H., K. Cousot, M. Lafourcade, M. Mekaoui, et C. Lopez (2020). A dataset for anaphora analysis in French emails. In Proceedings of the Third Workshop on Computational Models of Reference, Anaphora and Coreference, Barcelona, Spain (online), pp. 165–175. Association for Computational Linguistics. Accessed on Aug. 27, 2021.

Kalitvianski, R. (2018). Traitements formels et sémantiques des échanges et des documents textuels liés à des activités collaboratives [Formal and semantic processing of exchanges and textual documents related to collaborative activities.]. Theses, Université Grenoble Alpes. Accessed on Sep. 3, 2021.

Kim, J.-E. et K. Johnson (2013). The Impact of Moral Emotions on Cause-Related Marketing Campaigns : A Cross-Cultural Examination. Journal of Business Ethics 112(1), 79–90. Accessed on Aug. 18, 2021.

Klimt, B. et Y. Yang (2004). The enron corpus : A new dataset for email classification research. In J.-F. Boulicaut, F. Esposito, F. Giannotti, et D. Pedreschi (Eds.), Machine Learning : ECML 2004, Berlin, Heidelberg, pp. 217–226. Springer Berlin Heidelberg.

Kumar, A. (2021). An empirical examination of the effects of design elements of email newsletters on consumers' email responses and their purchase. Journal of Retailing and Consumer Services 58, 102349. Accessed on Aug. 29, 2021.

Miller, R. et E. Charles (2016). A psychological based analysis of marketing email subject lines. In 2016 Sixteenth International Conference on Advances in ICT for Emerging Regions (ICTer), pp. 58–65.

Mueller, A. et S. Guido (2018). Machine learning avec Python. O'Reilly Media, Inc.

Raffel, C., N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, et P. J. Liu (2019). Exploring the limits of transfer learning with a unified text-to-text transformer. arXiv preprint arXiv :1910.10683.

Saravia, E., H.-C. T. Liu, Y.-H. Huang, J. Wu, et Y.-S. Chen (2018). CARER : Contextualized affect representations for emotion recognition. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, pp. 3687–3697. Association for Computational Linguistics.

Seyeditabari, A., N. Tabari, et W. Zadrozny (2018). Emotion Detection in Text : a Review. Accessed on Sep. 1, 2021.

Virginie Rodriguez, M. S.-F. (2021). Le contenu des communications relationnelles par email des enseignes : Quelle perception par le consommateur ? [Content of retailers' relational e-mails : what is the consumer's perception ?]. In 20th International Marketing Trends Conference, Venise, Italy. Accessed on Aug. 29, 2021.

Yaqub, U., S. A. Chun, V. Atluri, et J. Vaidya (2017). Analysis of political discourse on twitter in the context of the 2016 us presidential elections. Government Information Quarterly 34(4), 613–626. Accessed on Aug. 31, 2021.

Summary

E-mail campaigns are strategic to optimize the relationship between a company and its customers. Thus, preventing news and requests from being ignored or considered a nuisance appears to be necessary. In this article, we are interested in newsletters' content and the emotions and opinions they convey. More specifically, we seek to evaluate how the latter can influence newsletter campaigns' performance indices (open rate and click rate). We first describe the French dataset we have constructed. Then, we explore the parameters and embeddings representing the emotions of the textual content to estimate the importance of their link with the performance of the campaign. Finally, we investigate whether our emotional representations of the text content can predict campaign performance.

Extraction de contenus sémantiques pour la vérification d'exigences systèmes

Aurélien Lamercerie*, David Rouquet**,
Valérie Bellynck*, Christian Boitet***, Vincent Berment****,***

* G-INP/LIG/GETALP, ** Tétras-Libre, *** UGA/LIG/GETALP, **** CS Group

Résumé. Cet article présente l'application d'une méthode d'extraction de contenu sémantique dans un contexte industriel, avec pour objectif la vérification automatique d'exigences systèmes rédigées non pas dans un langage contrôlé, mais en langue naturelle non contrainte. L'étape d'extraction utilise une analyse par transduction sémantique, implémentée en s'appuyant sur les standards du Web Sémantique du W3C. Elle part d'une représentation sémantique des textes, exprimée sous forme de graphes UNL (Universal Networking Language) à "sens garanti" (obtenus grâce à une étape de désambiguïsation interactive), qui produit d'abord une structure semi-formelle et indépendante de la langue source. Les outils développés construisent ensuite automatiquement une ontologie OWL à partir des spécifications du système, exprimées par des énoncés non contraints. Finalement, une vérification automatique des exigences est réalisée à l'aide de règles SPARQL génériques et de raisonneurs logiques. La fin de l'article décrit une mise en pratique sur des exigences extraites d'une spécification réelle.

1 Introduction

Le projet RAPID UNSEL [1], financé par la DGA [2] de 2019 à 2021, vise à fournir des outils pour accompagner la spécification de "systèmes de systèmes" (par exemple, un système de communications sol-air pour un aéroport ou un système de freinage d'urgence). Un tel système est construit à partir d'un cahier des charges, puis de documents de plus en plus détaillés : document de spécifications externes ou internes, document d'analyse générale ou détaillée, documents de programmation. Une spécification est un ensemble structuré d'exigences, qui sont majoritairement des énoncés descriptifs ou prescriptifs en langue naturelle (LN).

Les problèmes liés aux spécifications sont bien connus [3]. Ce sont *l'incohérence* [4], *l'incomplétude* et *l'inadéquation*. Ainsi, il arrive qu'il y ait des exigences contradictoires, que l'ensemble des exigences ne contienne pas tout ce qui est nécessaire ou que des exigences ne soient pas comprises par leurs lecteurs. Le coût d'une erreur s'avère parfois considérable. Ces problèmes, dont l'enjeu est important, sont pourtant peu ou mal traités au niveau opérationnel.

1. UNseL : Universal Networking system engineering Language
2. Direction Générale de l'Armement, `https://www.defense.gouv.fr/dga`
3. voir par exemple Dick et al. (2017).
4. ou plus précisément *l'inconsistance*, si on est dans un système logique comme une ontologie.

Par ailleurs, dans l'idée d'une application à des systèmes critiques, l'usage de représentations *à sens garanti* s'est ajouté aux contraintes du projet. La revue de l'état de l'art, par exemple Kamath et Das (2019), n'a pas permis de trouver des approches permettant de répondre à ce besoin. Nos travaux y apportent une réponse originale.

La vérification d'anomalies semble devoir s'appuyer sur la construction d'un système formel de type "ontologie métier" ou "ontologie de domaine". Pour cela, il faut représenter le sens de chaque exigence, ainsi que les liens structurels et séquentiels entre exigences, de façon aussi précise, exacte et complète que possible. Dans le projet UNSEL, nous implémentons les ontologies de domaine dans le langage logique OWL [5].

Il est difficile, voire impossible (Kay (2017) [6]), d'obtenir automatiquement une représentation linguistique non ambiguë, pour chaque exigence, telle que l'on puisse en dériver le sens dans une ontologie. Aucun analyseur disponible ne le fait, pour aucune langue. Une raison majeure de cette impossibilité est que les énoncés en LN sont presque toujours ambigus, et souvent imprécis, voire flous, même si on utilise un langage restreint. Pour garantir l'adéquation entre une exigence et sa représentation, le projet UNSEL implémente un système de désambiguïsation interactive intuitive permettant de voir qu'un passage est ambigu [7], et de choisir l'interprétation désirée.

Une fois que les exigences sont représentées au niveau linguistique de façon correctement désambiguïsée, dans notre cas grâce à des (hyper-)graphes UNL, il est nécessaire de les traduire dans l'ontologie considérée. C'est en effet seulement à ce niveau que les calculs de consistance et de complétude peuvent se faire. Pour répondre à cet enjeu, une technique d'extraction de sens, basée sur le concept de *transduction sémantique compositionnelle* (Lamercerie (2021)), a été adaptée aux besoins du projet.

La suite de cet article détaille la mise en œuvre de ce procédé. Partant de graphes UNL (section 2), le processus d'extraction de contenu (section 3) est appliqué à la vérification automatique d'un corpus d'exigences systèmes (section 4). Les outils présentés sont disponibles sous licence CeCILL-B dans un dépôt Gitlab dédié [8].

2 Représentation du sens des textes avec UNL

La mise en œuvre d'une approche d'extraction par transduction sémantique requiert une représentation sémantique des énoncés linguistiques, sous forme de graphes (ou mieux de réseaux) sémantiques. Dans le cadre du projet UNSEL, la représentation d'un énoncé est un (hyper-)graphe du langage UNL [9], qui est un langage formel permettant d'exprimer des structures abstraites de l'anglais, et dont le vocabulaire est formé de "lexèmes interlingues" (les UW, ou *Universal Words*).

5. W3C Working Group (2012)
6. *You can't get blood out of a stone*, Martin Kay, 2009, 40-ième anniversaire de l'ATALA.
7. projet LIDIA, Blanchon (1994), projet Eureka Eurolang, Boitet et al. (1995)
8. `https://gitlab.tetras-libre.fr/unl/tenet`
9. UNL Specification 3.3 (2004)

2.1 Le langage UNL

Le sigle UNL désigne un projet, un langage de graphes sémantiques d'énoncés en langue naturelle et un format de documents multilingues (intégré à Html via des balises de forme [xyz] et {attr=val}). Le projet international UNL a été lancé en décembre 1996 [10], avec initialement les 12 langues les plus parlées au monde, à l'initiative de l'Institute of Advanced Studies (IAS) de l'*Université des Nations Unies* [11] (UNU). Dans le *langage UNL*, le sens d'une phrase est représenté par un (hyper)-graphe, comme illustré dans la figure 1. L'idée de base est de rendre les graphes sémantiques compréhensibles et constructibles par les chercheurs et développeurs du monde entier. C'est pourquoi tous les symboles utilisés dans un graphe UNL proviennent de l'anglais. On peut néanmoins, par le jeu des restrictions sémantiques, dénoter des acceptions n'existant pas en anglais (e.g. 'tatami', 'alunir', certains niveaux de politesse, etc.).

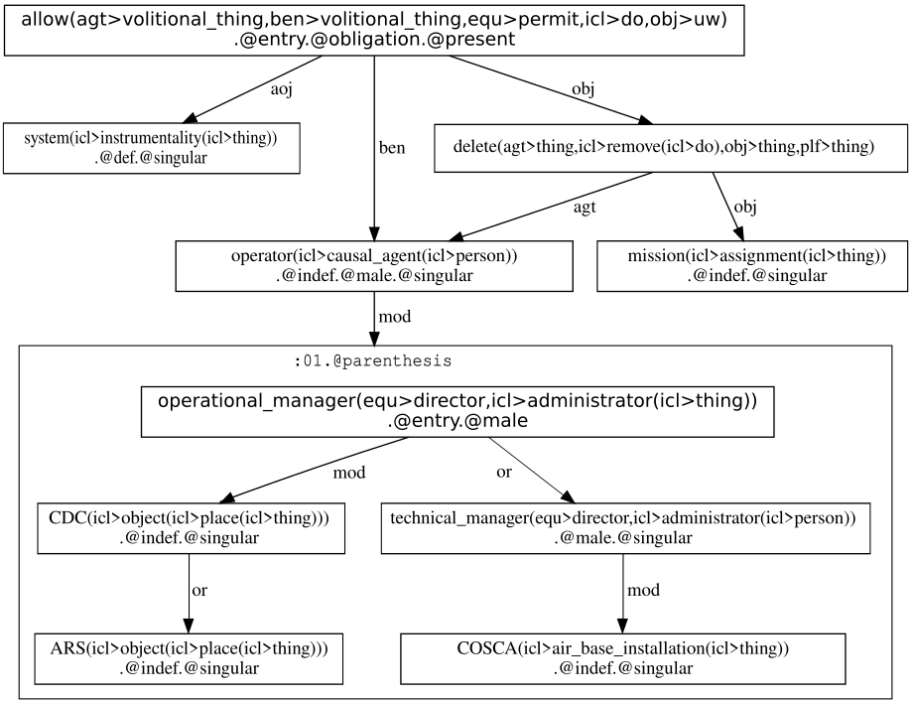


FIG. 1 – *Un graphe UNL possible pour l'exigence "Le système doit permettre à un opérateur (gestionnaire opérationnel d'un CDC ou d'un ARS ou gestionnaire technique d'un COSCA) de supprimer une mission".*

On convient qu'un graphe UNL représentant un énoncé E dans une langue quelconque L doit être la structure abstraite d'un énoncé anglais équivalent à l'énoncé d'origine E.

10. Uchida et al. (1996)
11. `https://unu.edu/`, Tokyo

C'est important parce que les relations sémantiques portées par les arguments d'un lexème de L (verbe ou déverbal comme 'permettre', 'permission', 'dépendre', ou adjectif comme 'utile', voire nom non déverbal comme 'méthode') ne sont pas toujours les mêmes que celles portées par les arguments correspondants d'un lexème anglais équivalent, bien que le répertoire des relations sémantiques soit universel. C'est pourquoi on peut dire que le langage UNL est un *langage pivot anglo-sémantique*.

Un (hyper-)graphe UNL est formé de nœuds, d'arcs orientés et d'attributs booléens, ces différents éléments portant des informations de natures différentes.

Un *nœud* peut être élémentaire ou composé.

— Un *nœud élémentaire* contient un UW (*Universal Word*, ou *lexème interlingue*) et des attributs booléens sémantiques ou pragmatiques (comme `.@past`). Un UW dénote une acception. Il est formé d'un mot-vedette (*headword*) [12], et d'une liste de restrictions. L'UW suivant peut dénoter 'amerrir' : `land(icl>do, aoj>flying thing, gol>water)`.

— Un *nœud composé* est un sous-graphe, appelé *scope*, représentant une partie de la phrase. Le scope i est constitué de tous les arcs de numéro de scope i et des nœuds qu'ils relient. Un arc ne peut appartenir qu'à un scope, dont il porte le numéro ($:00, :01, :02 \ldots$ [13]), mais un nœud peut être commun à plusieurs scopes. Tout scope doit contenir un et un seul nœud portant l'attribut `.@entry`. Enfin, un scope doit être connexe par arcs, si on néglige l'orientation des arcs.

Sur l'exemple de la figure 1, l'UW `operator(icl>causal_agent(icl>person))` dénote un *opérateur*, tandis que le verbe *supprimer* est représenté par l'UW `delete(agt>thing, icl>remove(icl>do), obj>thing, plf>thing)`.

Les *arcs* définissent des relations entre les nœuds. Ils sont orientés et étiquetés par des relations sémantiques binaires prises dans un répertoire fixé (une quarantaine).

— Les relations sont notées par des abréviations (ex : agent (`agt`), objet (`obj`), bénéficiaire (`ben`), localisation (`plc`), durée (`dur`), destination (`plt`), possesseur (`pos`), etc.).

— Une extension due à UNL-ru et UNL-fr, très utile mais pas encore introduite dans le standard UNL, consiste à ajouter un attribut de *position argumentaire* (`.@A`, `.@B`, `.@C`, `.@D`) à certains arcs, ce qui permet de distinguer deux prédicats d'arités différentes [14].

Les *attributs* (booléens) sont préfixés par `.@`, et attachés aux nœuds (élémentaires ou composés). Ils apportent des précisions sur les nœuds du graphe, telles que :

— le rôle dans le graphe (ex : `.@entry` indique le nœud principal d'un scope, d'où on doit partir pour interpréter le graphe) ;

— des informations sémantiques et pragmatiques, par exemple le temps abstrait (*time* par opposition à *tense*), l'aspect, la modalité, la négation ;

— d'autres éléments de communication comme les actes de parole, le niveau de politesse, le nombre (singulier, pluriel, collectif), la détermination (défini, indéfini), etc.

Par exemple, l'utilisation de l'imparfait en français, ou du progressif passé en anglais, peut être représentée par la combinaison d'attributs `.@past.@repetition`.

12. un lemme anglais ou un lemme emprunté par l'anglais à une autre langue, comme *tatami*.

13. le numéro du scope principal, : 00, est en général omis.

14. Par exemple, `delete(agt.@A>thing, icl>remove(icl>do), obj.@B>thing, plf>thing)` pour 'A supprimer/tuer B', et `delete(agt.@A>thing, icl>remove(icl>do), obj.@B>thing, plf.@C>thing)` pour 'A supprimer/ôter B de C'.

2.2 Enconversion et sérialisation des graphes UNL

La transformation d'un texte en graphes UNL (un par phrase) est appelée *enconversion*, et non analyse, pour souligner qu'on passe d'un espace lexical (français, anglais…) à un autre (celui des UW UNL). L'opération inverse est appelée *déconversion*.

La première étape de l'enconversion appelle un analyseur structural (syntaxique et sémantique) développé avec l'environnement Ariane-H [15] et permettant de garantir le sens analysé. En présence d'ambiguïtés, il produit toutes les analyses possibles et pose des questions à l'utilisateur pour obtenir le sens correct. La représentation obtenue est, par construction, une image complète et fidèle du texte analysé. Cette représentation est ensuite convertie en graphes UNL.

Un des problèmes des approches dites "à l'état de l'art" est qu'elles utilisent un parseur vers une représentation sémantique interlingue, comme UNL ou AMR, en supposant faussement qu'il fournit automatiquement un résultat correspondant à ce qu'a voulu dire le rédacteur. Or il y a beaucoup d'ambiguïtés lexicales, d'ambiguïtés d'attachement et d'ambiguïtés de dépendance sémantiques : même avec une phase de désambiguïsation automatique, les résultats sont en pratique très souvent incorrects, confirmant l'affirmation de Martin Kay (2017). L'étape de désambiguïsation interactive apporte une réponse à ces problèmes, réponse renforcée par le choix d'UNL. Ce formalisme est moins connu qu'AMR, mais nettement plus expressif et précis, notamment pour la partie lexico-sémantique.

Afin d'obtenir une chaîne d'extraction entièrement fondée sur les standards du Web sémantique du W3C, nous utilisons une sérialisation RDF des graphes UNL (Rouquet et al. (2020)). Un convertisseur du format standard UNL vers la sérialisation dite *UNL-RDF* a été développé dans le cadre du projet. Il est disponible sous forme d'exécutable Java [16] et de service Web [17].

3 Extraction de contenu par transduction sémantique

La contribution principale de notre démarche est l'adaptation d'une technique d'analyse par transduction sémantique à des graphes UNL, transformés pour aboutir à une nouvelle structure formelle. Cette étape permet la construction automatique d'une ontologie OWL à partir d'une ontologie cadre passée en paramètre. L'implémentation est fondée sur les standards du Web sémantique (RDF, OWL, SPARQL, SHACL) [18].

3.1 Analyse par transduction sémantique

Le principe d'analyse par transduction sémantique (Lamercerie (2021)) a été validé par une première expérimentation sur des graphes AMR (Abstract Meaning Representation, Banarescu et al. (2013)). Nous l'avons adapté aux hypergraphes UNL [19], en reprenant les notions de filet sémantique et de schéma de transduction compositionnel.

15. `https://linguarium.org`
16. `https://gitlab.tetras-libre.fr/unl/unlTools`
17. `https://unl.demo.tetras-libre.fr/`
18. W3C Standards (2021)
19. Ces graphes ne sont pas des graphes classiques en théorie des graphes, où on ne peut avoir plus d'un arc allant d'un sommet s_i à un sommet s_j (dans un graphe orienté), ou plus d'un arc reliant un sommet s_i à un sommet s_j (dans un graphe non orienté). Il s'agit plutôt de *réseaux*, et d'un abus de langage. On parle aussi de *graphes de transitions* au lieu de réseaux de transitions d'automates.

Filets sémantiques. Intuitivement, un *filet sémantique* est un objet construit sur un graphe sémantique G de façon inductive, à partir d'une base formée de filets "atomiques" (de type 'atome') correspondant aux nœuds du graphe, en utilisant des règles de construction. Nous notons F_G (ou F) l'ensemble des filets ainsi définis. Un filet est un triplet *f = (ensemble de nœuds* ou *"ancrage", type, ensemble de valeurs)*. Dans un filet atomique, les valeurs possibles sont (1) celles des parties de l'information portée par le nœud (UW complet, mot vedette, valeurs des restrictions, traits sémantiques, identificateurs des nœuds et des arcs), et (2) les résultats de l'application de fonctions (ex : score calculé par une distance Lesk liée à un thésaurus).

Une règle de "transduction sémantique" d'arité k prend en argument k filets $f_1 \dots f_k$ et produit un filet f dont l'ensemble de nœuds est l'union des nœuds des f_i, dont le type est déterminé par la règle (par exemple, *list<atom>*), et dont les valeurs sont calculées à partir des valeurs des filets arguments (par exemple, le maximum ou la moyenne des scores, l'union des traits sémantiques de certains nœuds ou filets). La définition 1 suivante précise ce concept.

Définition 1 *Soit $\mathcal{G}$ un graphe étiqueté et S l'ensemble des sommets de G. Soit $\mathcal{T}$ un ensemble de types, et $\mathcal{V}$ un ensemble de valeurs. Un filet sémantique sur $\mathcal{G}$ est une structure $f = (x, \tau, v)$ telle que $x \subseteq S$ est une partie de S, $\tau \in \mathcal{T}$ est un type et $v \subseteq \mathcal{V}$ est un ensemble de valeurs.*

Le type d'un filet définit la nature des valeurs qui lui sont associées, et les opérations qui pourront lui être appliquées. L'ensemble des types $\mathcal{T}$ est supposé muni d'une relation d'ordre, permettant d'établir plusieurs niveaux d'analyse. Notre implémentation inclut le type *atome*, caractérisant des filets *atomiques* ne couvrant qu'un seul nœud, le type *composite*, définissant des filets *composite* associant un concept à plusieurs nœuds, ou encore le type *list<τ>* qui caractérise une liste d'éléments de type τ.

La notion de filet sémantique est illustrée par la figure 2. Le filet F_a, de type *atome*, s'ancre au nœud *1* et porte plusieurs valeurs, dont la classe de l'atome (ici *operator*) et sa classe parente (*agent*). Le filet F_b, de type *list<atom>*, s'ancre aux nœuds *2* et *4* et porte plusieurs valeurs, dont les éléments de la liste et la relation associée (*mod*). Le filet F_{ab}, de type *list<composite>*, s'ancre aux nœuds *1*, *2* et *4*, et porte plusieurs valeurs, dont les éléments composites de la liste (classes *operational manager* et *technical manager*) et la classe mère (*operator*). Nous verrons plus loin que le filet F_{ab} est obtenu par composition des filets F_a et F_b.

Schémas de transduction compositionnels (STC). La définition 2 spécifie une structure associant une formule logique et un opérateur de transduction. Elle s'applique aux filets d'un graphe sémantique, et permet l'obtention de nouveaux filets par composition. Dans cette optique, nous considérons les propriétés des filets, dérivées des types et des valeurs. L'ensemble des propriétés considérées est noté $\mathcal{P}$.

Définition 2 *Soit P un ensemble de prédicats. Soit $\mathcal{R}_\mathcal{P}$ un ensemble de conjonctions d'expressions $p(f_1, \dots, f_n)$, avec $p \in P$. Un schéma de transduction compositionnel (STC) σ est une paire $\sigma = (\varphi, tr)$ telle que :*

- *φ est une formule logique sur $\mathcal{R}_\mathcal{P}$;*
- *tr (d'arité k) est une fonction de F^k dans F définie par deux fonctions ψ_τ et ψ_v et telle que, si $f_i = (x_i, \tau_i, v_i)$ avec $i \in [1..k]$, et $f = (x, \tau, v) = \sigma(f_1, \dots, f_k)$, alors $x = \bigcup x_i$, $\tau = \psi_\tau(\tau_1, .., \tau_k)$, et $v = \psi_v(v_1, .., v_k)$, où ψ_τ et ψ_v sont deux fonctions retournant respectivement un type et un ensemble de valeurs.*

La figure 2 donne un exemple d'application d'un schéma $\sigma = (\varphi, tr)$, composant un filet *atomique* et un filet de type *liste*. La partie requête du schéma est définie par $\varphi = atom(f_1) \wedge$ *list<atom>*$(f_2) \wedge mod(f_1, f_2)$. Elle permet de sélectionner les filets à composer (le filet F_a de type *atom* et le filet F_b de type *list<atom>*, ces deux filets étant liés par la relation `mod`.

Le filet résultant F_{ab} est obtenu par application de la fonction tr sur les filets sélectionnés : $F_{ab} = (F_a.x \cup F_b.x, \psi_\tau(F_a.\tau, F_b.\tau), \psi_v(F_a.v, F_b.v))$. Les fonctions ψ_τ et ψ_v permettent de définir le type (ici, *list<atom>*) et l'ensemble de valeurs du nouveau filet (objets en relation définissant une hiérarchie de concepts).

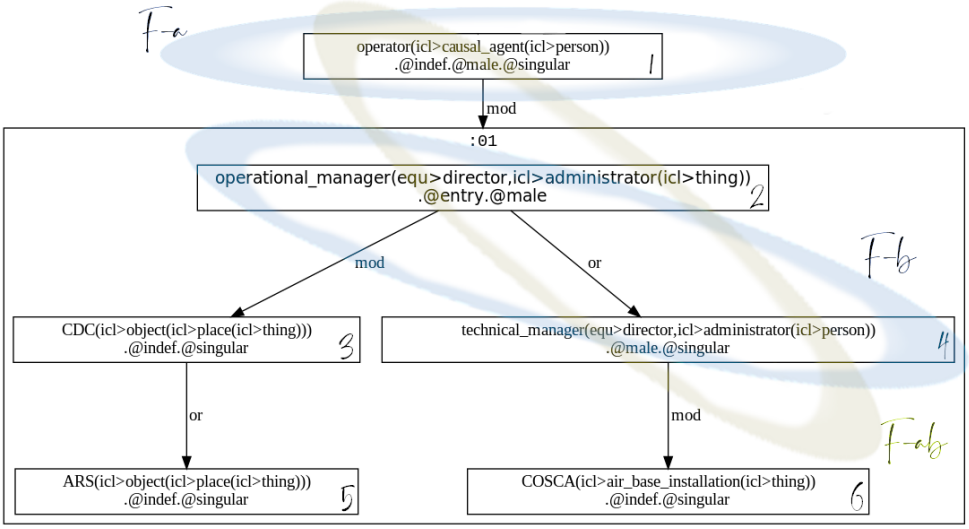


FIG. 2 – *Graphe UNL-RDF portant sur la désignation d'un opérateur, avec quelques filets.*

Application des STC. L'objectif du processus d'analyse est d'enrichir le graphe UNL-RDF en prenant en compte son contenu et sa structure. La mise en œuvre suit un mode d'exécution à quatre niveaux : (1) **extraction des éléments atomiques**, (2) **formation d'éléments composites** par un procédé récursif, (3) **extraction des propriétés et relations** pour les éléments atomiques et composites, (4) **construction de l'ontologie cible**. Le typage des filets permet d'ajuster le traitement pour en assurer l'efficacité et la terminaison. Un exemple d'implémentation est donné dans la section 3.2, avec quelques précisions complémentaires.

Les STC de niveau 1 s'appliquent sur les sommets du graphe UNL-RDF. Ils initialisent le traitement en générant des filets *atomiques*. L'extraction des éléments atomiques est paramétrée dans une *ontologie cadre* précisant le résultat attendu (voir section 3.2 ci-dessous).

Les STC de niveau 2 suivent un procédé récursif, contrôlé en s'appuyant sur le typage des filets. Cette étape permet de construire des filets *composites*, et d'obtenir ainsi une hiérarchie de concepts dans l'ontologie cible. La figure 2 est une illustration de ce procédé : le filet obtenu désigne une liste d'éléments composites (liste de classes) avec une hiérarchie entre la classe parente *operator* et les classes filles *operational manager*, *technical manager*.

Les STC de niveau 3 opèrent de manière similaire pour détecter des propriétés sur les éléments, atomiques ou composites, et des relations entre ces éléments. Tandis que les STC des niveaux précédents produisent des filets rattachés à des classes d'éléments (par exemple, des agents, des messages, des actions), les STC de niveau 3 traduisent les relations entre ces filets en propriétés sur les classes. Les filets produits à ce stade sont de type *property*. Ils sont centrés sur les verbes, traduisant l'attribution d'une propriété ou la qualification d'une action, et rattachés à plusieurs classes d'éléments. Sur l'exemple présenté, la classe *operator* est reliée à l'action *delete*, avec pour objet la classe *message*. Les filets produits par les STC de niveau 3 semblent bien correspondre aux *factoïdes* [20] utilisés dans le système Watson.

Finalement, les STC de niveau 4 génèrent l'ontologie attendue à partir des filets générés. Le typage des filets oriente directement la construction de nouvelles classes, instances et propriétés pour mettre à jour la structure cible.

3.2 Implémentation

Cette section présente une implémentation des STC, entièrement basée sur les standards du Web sémantique. Elle est disponible en Open Source sous l'acronyme TENET (*Tool for Extraction using Net Extension by (semantic) Transduction*) [21].

Entrées et paramètres. Le processus d'extraction prend en entrée un ensemble de graphes UNL, dans leur sérialisation RDF. Il prend par ailleurs en paramètre une ontologie cadre, qui définit les objets visés selon le contexte métier, ainsi que des *graines d'extraction* permettant d'initialiser les filets de niveau 1. L'ontologie cadre dépend du point de vue attendu : elle contient les grandes classes d'éléments à extraire (une dizaine de classes dans notre cas). Pour chacune de ces classes, il est nécessaire de déclarer au moins une propriété de type *has-seed* (graines d'extraction).

Processus d'extraction à base de STC. Les STC sont implémentés sous la forme de requêtes SPARQL-construct qui s'appliquent aux graphes UNL-RDF. Les requêtes sont intégrées dans un graphe de contraintes (*shapes*) respectant la spécification SHACL-SPARQL [22]. Elles sont ordonnées en niveaux d'application (voir section 3.1), grâce au mécanisme *sh :order* [23].

Les règles SPARQL dépendent fortement de la structure des graphes sémantiques en entrée, elle-même dépendant principalement du formalisme UNL et de la phraséologie du corpus. Par contre, les règles sont génériques du point de vue du contenu métier des phrases. En effet, ce dernier est paramétré par l'ontologie cadre. Ainsi, des règles développées sur la base de notre corpus, décrivant un système de communication sol-air, restent *a priori* valables pour des spécifications métier très différentes, comme celles d'un système de freinage d'urgence.

Sortie. La sortie est un ensemble de triplets RDF-OWL enrichissant et instanciant une ontologie cadre passée en paramètre. Il est tout à fait possible que le processus d'extraction vise plusieurs ontologies décrivant des facettes différentes du système.

20. Hovy et al. (2002)
21. `https://gitlab.tetras-libre.fr/unl/tenet`
22. `https://www.w3.org/TR/shacl/#dfn-shacl-sparql`
23. `https://www.w3.org/TR/shacl/#order`

4 Application à la vérification d'un corpus d'exigences

Nous avons appliqué nos méthodes sur un corpus réel d'exigences système fourni par la DGA (*SRSA-IP*). Ce corpus est composé de 367 exigences décrivant un système de communication sol-air. Il n'est pour l'instant pas accessible publiquement. Les premières applications ont été réalisées sur un corpus pilote de 40 exigences, sélectionnées dans SRSA-IP pour leur complexité linguistique et leur représentativité du corpus. Nous présentons d'abord une extraction et des vérifications réalisées sur l'exigence de la figure 1, puis les résultats obtenus sur le corpus pilote.

4.1 Ontologie cadre et règles d'extraction

La figure 3 présente un aperçu de l'ontologie cadre utilisée dans les premières expérimentations. Elle contient essentiellement une dizaine de classes accompagnées de graines d'extraction. Les requêtes SPARQL qui composent les STC de niveau 1 utilisent les graines de l'ontologie cadre, pour identifier les éléments atomiques dans les exigences. Actuellement, les graines sont basées sur les restrictions des UW. Par exemple, dans la figure 3, on voit que la classe *Agent* sera initialisée avec toutes les UW portant les restrictions `icl>administrator`, `icl>person` ou `icl>human`. Nous envisageons d'autres méthodes, permettant de définir ces graines à partir d'exemples ou de les généraliser à des sous-graphes.

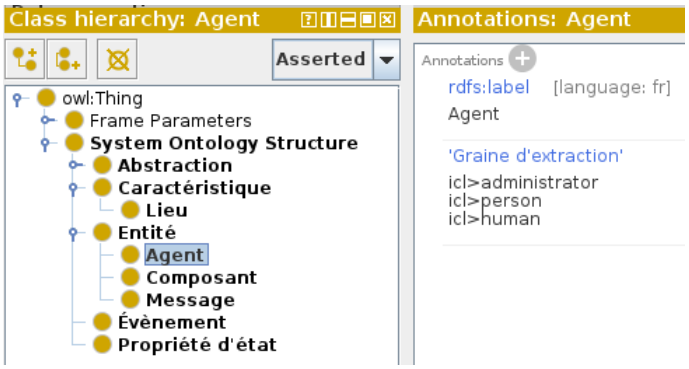


FIG. 3 – *Exemple d'ontologie cadre*

Nous avons défini 25 règles SPARQL pour l'extraction. Partant des éléments atomiques identifiés par les graines, les règles parcourent les (listes de) modificateurs (`mod` en UNL), précisant les éléments atomiques, pour aboutir à une hiérarchie d'éléments composites dans l'ontologie. D'autres règles extraient la liste ouverte des relations que les éléments composites peuvent entretenir entre eux, par exemple les actions des *Agents* sur les *Composants*. Enfin, les classes et relations de l'ontologie sont instanciées. Ce mécanisme permet en particulier d'assurer la traçabilité de ce qui est extrait, par exemple que l'exigence *STB_PHON_300* mentionne qu'un certain opérateur peut supprimer une mission particulière.

4.2 Exemple d'extraction et de vérification sur une exigence du corpus

L'application des règles d'extraction SPARQL sur le graphe UNL-RDF de la figure 1 produit 289 triplets RDF intermédiaires dans le processus de transduction, et 119 triplets ajoutés à l'ontologie cadre du système. Les informations extraites sont illustrées par la figure 4 avec :

— une hiérarchie des agents mentionnés, en haut à gauche (en jaune),

— une classe définie comme union de deux autres, en haut à droite (en jaune),

— une *propriété d'événement* dont le domaine et le but sont précisés, à droite (en bleu),

— l'assertion précisant qu'une instance d'opérateur "*delete*" une instance de mission, en bas (en violet).

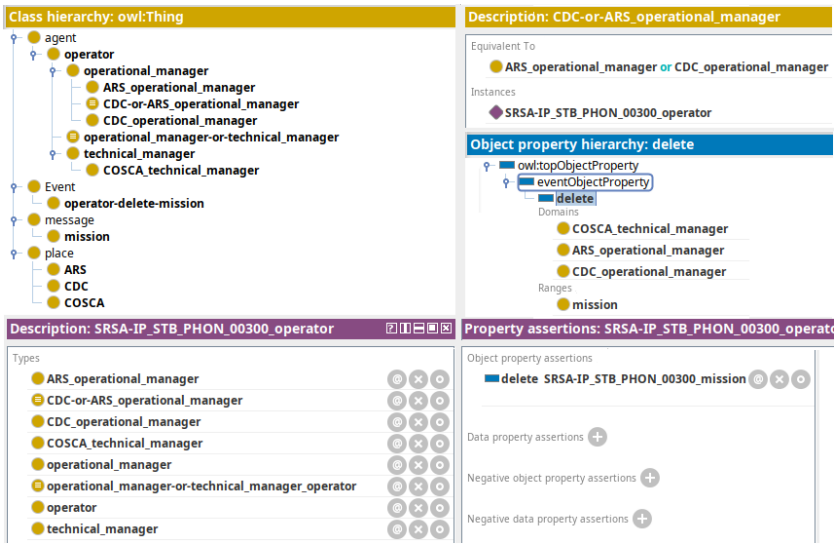


FIG. 4 – *Aperçu du résultat d'extraction sur l'exigence de la figure 1*

Le contenu RDF-OWL extrait permet de réaliser des vérifications, également implémentées sous la forme de règles SPARQL. Des messages d'alerte et des suggestions sont retournées à l'utilisateur après le contrôle des points suivants :

— **Sous-spécifications** dans les exigences, par exemple pour la mention *gestionnaire opérationnel* dans une exigence, parle-t-on bien de toutes les sous-classes comme *gestionnaires opérationnel* d'un *CDC*, d'un *ARS*, etc. ?

— **Défaut terminologique**, par exemple si la seule sous-classe de *mission*, extraite dans toutes les exigences, est *mission COSCA*, c'est soit que l'on a deux termes pour identifier le même concept, soit que l'on a sous-qualifié *mission* dans certaines exigences.

— **Qualification homogène des éléments en relation**, par exemple si une exigence mentionne qu'un *gestionnaire COSCA peut supprimer une mission COSCA* et si une autre indique qu'un *opérateur d'un CDC peut créer une mission*, on pourra proposer de compléter *mission* par *CDC* dans la seconde exigence.

D'autres vérifications de consistance entre les exigences peuvent être réalisées à l'aide de raisonneurs logiques génériques. Le rapport Rouquet et al. (2020) détaille un exemple de ce type, dont voici une version simplifiée : (1) "*Une voie radio peut prendre deux états : écoute et veille.*" et (2) "*L'opérateur place la voie radio dans l'état trafic.*". À partir de ces deux exigences, une ontologie incohérente est produite, car *trafic* ne fait pas partie de l'ensemble {*écoute*, *veille*} des états possibles pour une *voie radio*.

4.3 Résultats obtenus sur le corpus pilote

Partant des 40 exigences du corpus pilote, notre processus produit 4990 triplets RDF intermédiaires dans le processus de transduction, et 1930 triplets ajoutés à l'ontologie cadre du système. Les informations suivantes sont notamment ajoutées à l'ontologie :

— 33 classes organisées en hiérarchie et reliées par des relations ensemblistes dont 12 classes d'agents et 21 composants physiques ou "abstraits",

— 14 propriétés, liant les classes précédentes, dont 5 correspondent à des actions, des événements ou des états du système (*create*, *delete*, *release*, *set_up*, *include*).

Les mesures classiques de précision et de rappel restent à produire sur l'ensemble du corpus, mais les résultats sur le corpus pilote sont plus qu'encourageants. Les informations extraites ont permis de détecter 100% des anomalies pointées manuellement sur le corpus. On note toutefois des aspects pas ou mal pris en compte dans l'extraction et qui laissent une bonne marge de progression, par exemple les modalités déontiques et temporelles.

Malgré ces limites, l'examen du graphe sémantique OWL produit est directement instructif pour un humain et fournit des informations denses et pertinentes pour la compréhension du système spécifié. Par exemple, la notion de *mission* est correctement explicitée comme une entité contenant des *voies radio*. Les différents types de *mission* sont extraits, ainsi que les types d'*agents* qui peuvent les créer, les supprimer, ou modifier leurs *voies radio*.

5 Conclusion

Nous avons présenté dans cet article l'application concrète d'une chaîne de traitement globale, partant d'exigences système exprimées en langue naturelle (LN) pour aboutir à une ontologie OWL du système décrit par ces exigences. Cette chaîne a plusieurs étapes : (1) l'enconversion des énoncés en LN dans le format standard UNL, avec une étape de désambiguïsation interactive intuitive en langue source (de rédaction), (2) la sérialisation RDF des graphes UNL, (3) l'extraction du contenu sémantique pour construire une ontologie OWL du système et (4) la vérification automatique de l'ontologie produite.

L'expérimentation réalisée a permis de montrer qu'il est possible de construire des axiomes OWL qui supportent un raisonnement non trivial. Il devient ainsi envisageable de gérer la proximité sémantique des termes, de vérifier la cohérence structurelle des entités décrites dans les exigences, et de mettre en évidence des incohérences sur les propriétés définies. L'usage des graphes UNL permet de réduire la dépendance monolingue des logiciels envisagés, tandis que le processus de désambiguïsation interactive apporte une garantie de sens sur les représentations pivot (UNL) exploitées lors de l'extraction. L'analyse par transduction sémantique est un procédé simple, traçable et adaptable pour l'interprétation des énoncés et la construction des ontologies visées.

Références

Banarescu, L., C. Bonial, S. Cai, M. Georgescu, K. Griffitt, U. Hermjakob, K. Knight, P. Koehn, M. Palmer, et N. Schneider (2013). Abstract Meaning Representation for Sembanking. In Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse, pp. 178–186. Sofia, Bulgaria : Association for Computational Linguistics.

Blanchon, H. (1994). LIDIA-1 : une première maquette vers la TA Interactive "pour tous". Thèse, Université Joseph-Fourier - Grenoble I.

Boitet, C., E. Planas, H. Blanchon, É. Blanc, J.-P. Guilbaud, P. Guillaume, M. Lafourcade, et G. Sérasset (1995). LIDIA-1.2, une maquette de TAO personnelle multicible, utilisant la messagerie usuelle, la désambiguïsation interactive et la rétrotraduction.

Dick, J., E. Hull, et K. Jackson (2017). Requirements Engineering. Springer International Publishing.

Hovy, E., U. Hermjakob, C.-Y. Lin, et D. Ravichandran (2002). Using Knowledge to Facilitate Factoid Answer Pinpointing. In Proceedings of COLING 2002.

Kamath, A. et R. Das (2019). A Survey on Semantic Parsing. In Automated Knowledge Base Construction (AKBC).

Kay, M. (2017). Translation : Linguistic and Philosophical Perspectives, Volume 221 of CSLI Lecture Notes. CSLI Publications.

Lamercerie, A. (2021). Principe de transduction sémantique pour l'application de théories d'interfaces sur des documents de spécification. Thèse, Université de Rennes 1.

Rouquet, D., V. Bellynck, V. Berment, et C. Boitet (2020). Natural Language Representation and Content Extraction using RDF, SHACL and UNL.

Uchida, H., M. Zhu, et T. Della Senta (1996). UNL : An Electronic Language for Communication, Understanding and Collaboration. UNU/IAS/UNL Center.

UNL Specification 3.3 (2004). (http ://www.unlweb.net/wiki/images/a/ab/Spec33.pdf).

W3C Standards (2021). Semantic web (https ://www.w3.org/standards/semanticweb/).

W3C Working Group (2012). OWL-2 Overview (http ://www.w3.org/TR/owl2-overview/).

Summary

This paper presents the application of a semantic content extraction method in an industrial context, with the objective of automatic verification of system requirements written not in a controlled language, but in an unconstrained natural language. The extraction step uses a semantic transduction analysis, implemented using the W3C Semantic Web standards. It starts from a linguistic representation of the texts, in the form of UNL (Universal Networking Language) graphs with "guaranteed meaning" (obtained thanks to an intermediate step of interactive disambiguation), which first produces a semi-formal structure independent of the source language. The system then builds an OWL ontology from the system specifications, expressed by unconstrained NL statements. Finally, an automatic verification of the requirements is performed using generic SPARQL rules and logical reasoners. The end of the article describes a practical implementation on requirements extracted from a real specification.

Vers un Système de Question-Réponse Multilingue, Génératif et Unifié

Wissam Siblini, Nacir Bouazizi
Charlotte Pasqual

Worldline
prenom.nom@worldline.com

Résumé. Au cours des cinq dernières années, les systèmes de question-réponse (QA) automatiques ont beaucoup évolué, dans plusieurs directions parfois segmentées : QA extractif, QA en domaine ouvert, QA multilingue, QA unifié, etc. Dans ce papier, notre objectif est de regrouper ces différentes directions en proposant une approche qui répond convenablement à chacune d'entre elles. En particulier nous proposons un système multilingue génératif qui répond à plusieurs types de questions (unifié) en domaine ouvert. Notre proposition s'appuie sur une sélection de briques élémentaires de l'état de l'art, un travail important de préparation et d'augmentation de jeux de données, le design d'une dynamique d'entraînement adaptée, et l'assemblage d'un processus global cohérent. L'évaluation expérimentale quantitative suggère une performance compétitive sur plusieurs des sous tâches et l'évaluation qualitative permet d'observer des comportements intéressants sur des croisements de ces tâches. Notre modèle est finalement implémenté dans un agent dialoguant dont l'objectif est de répondre à des questions utilisateur à partir de Wikipedia.

1 Introduction

Depuis la parution du modèle Transformer (Vaswani et al., 2017), de nombreuses tâches du domaine du Traitement Automatique du Langage Naturel (TALN) ont fortement gagné en popularité. Nous nous intéressons ici au problème de question-réponse. Nous pouvons distinguer deux situations : en domaine fermé/limité où on cherche à répondre à partir d'un petit passage de texte ou en domaine ouvert à partir d'un ensemble varié de documents textuels (par exemple l'encyclopédie Wikipedia). Dans les deux cas, la tâche la plus populaire est appelée question-réponse extractif (Rajpurkar et al., 2016), dont l'objectif est d'extraire un petit passage de quelques mots constituants la réponse. En domaine fermé, on trouve de nombreuses directions de recherche indépendantes, par exemple : (i) certains s'intéressent à répondre à plusieurs types de questions (Khashabi et al., 2020), (ii) d'autres s'intéressent à la génération d'une réponse bien formée (Tan et al., 2017) (iii) ou encore à la capacité à traiter plusieurs langages (Siblini et al., 2019). En domaine ouvert la tâche de base, appelée Open Domain Question Answering (ODQA) représente déjà un challenge car il est nécessaire

d'identifier un ou plusieurs passages pertinents (Manning et al., 2008) d'abord, puis de résoudre la tâche de question-réponse en domaine fermé qui en découle. Néanmoins, on trouve des travaux qui ajoutent des éléments de complexité supplémentaires comme l'aspect conversationnel (Qu et al., 2020 ; Siblini et al., 2021), multilingue (Liu et al., 2019), ou génératif (Muller et al., 2021).

Dans ce travail, notre objectif est de s'attaquer à plusieurs de ces challenge à travers une stratégie de combinaison et d'affinage (fine-tuning) de modèles sur différents jeux de données. On considérera les challenges génératif, unifié (capacité à répondre à plusieurs types de question), en domaine ouvert, et multilingue. A notre connaissance, il n'existe par exemple aucun papier proposant une approche à la fois générative et unifiée. Notre proposition s'appuie sur un modèle basé Transformer (BART, Lewis et al. (2019)) et repose sur les contributions suivantes : (i) un travail important de préparation et d'augmentation de données pour disposer, dans deux langues cibles (anglais et français), de données correspondant à plusieurs types de questions, dans un même format (ii) la proposition d'un mécanisme d'entraînement, de fine-tuning et d'inférence pour obtenir un système répondant à l'ensemble des challenges mentionnés (iii) d'une sélection pertinente et argumentée de briques de la littérature pour obtenir des résultats empiriques prometteurs. Nous proposons un processus complet permettant d'appliquer nos modèles pour répondre à tout types de questions sur les Wikipedia français et anglais et d'obtenir des réponses générées bien formulées. Nous l'implémentons finalement dans un agent dialoguant automatique et montrons un exemple d'interaction. S'agissant de travaux encore préliminaires, seuls quelques résultats quantitatifs sont donnés et nous indiquons en conclusion des directions pour nos travaux futurs.

2 TALN et Question-Réponse

Le Traitement Automatique du Langage Naturel est un sous-domaine de la linguistique, de l'informatique et de l'intelligence artificielle qui s'intéresse aux interactions entre les ordinateurs et le langage humain. La tâche de question-réponse (ou Question Answering noté QA) est une partie du TALN dont l'objectif est la construction de systèmes répondant automatiquement à des questions posées par des humains. Au cours de la dernière décennie, les approches utilisées ont progressivement évolué depuis des méthodes probabilistes et classiques (TF-IDF, Naive Bayes, SVM) vers des méthodes à base de réseaux de neurones récurrents (LSTM, GRU) puis enfin vers le deep learning avec des mécanismes d'attention comme dans le Transformer (Vaswani et al., 2017).

Un Transformer est un réseau de neurones multi-couches comportant deux parties : un encodeur et un décodeur. L'encodeur prend en entrée une séquence de tokens (mots) et les transforme en une séquence de vecteurs qui dépendent chacun du token d'entrée, de sa position et des autres tokens de la séquence (via le mécanisme d'auto-attention). L'encodeur est souvent utilisé seul (Cf modèle BERT, Devlin et al. (2019)) avec une couche finale de classification pour résoudre des tâches discriminantes. Le décodeur est un modèle génératif. Il suit un processus itératif dans lequel il prend en entrée la sortie de l'encodeur ainsi que les tokens qu'il a déjà généré à l'itération précédente, puis utilise un mécanisme d'attention pour enfin prédire le token suivant. Il existe aujourd'hui des dizaines de modèles inspirés de tout ou partie du Transformer, qui produisent des

performances surpassant parfois l'humain sur des tâches de TALN (Devlin et al., 2019 ; Lewis et al., 2019), notamment relatives au QA.

Le QA extractif Il consiste à extraire pour une question et un contexte (paragraphe de texte) donnés, une partie courte du contexte qui répond à la question. De nombreuses approches ont été proposées pour résoudre cette tâche, en utilisant notamment le populaire jeu de données SQuAD (Rajpurkar et al., 2016). L'approche la plus emblématique aujourd'hui est BERT. D'abord, la question, le contexte, et des séparateurs sont concaténés. La séquence résultante est fournit en entrée de BERT. Les vecteurs finaux obtenus pour chaque token du contexte (contextualisés par les tokens de la question) passent ensuite dans un classifier (couche de neurones) qui prédit pour chacun la probabilité d'être le début et la fin de la réponse.

Le QA génératif Dans ce paradigme, la réponse fournie par le modèle est une séquence de mots générée. Cette séquence peut correspondre à une partie du contexte, mais ne s'y restreint pas. Elle peut également être un texte libre, ou une combinaison entre du texte libre et un ou plusieurs passages du contexte. Les approches de QA génératif s'appuient généralement sur des architecture encodeur-décodeur ou la première partie encode la question et le contexte et la seconde génère la réponse. C'est le cas par exemple de S-NET (Tan et al., 2017) ou PALM (Bi et al., 2020).

QA unifié On considère qu'il existe plusieurs types de questions. Les questions **extractives** peuvent trouver une réponse par l'extraction directe d'une partie du contexte. Les questions **abstractives**, à l'inverse, nécessitent de générer une réponse qui n'en fait pas directement partie. Il existe aussi des questions avec des formats particuliers comme le format **multi-choix** ou **booléen** dont la réponse est soit oui soit non. L'objectif du QA unifié est de proposer un unique système capable de répondre à ces différents types de questions. UnifiedQA (Khashabi et al., 2020) répond à cet objectif avec brio et un entraînant un unique modèle simultanément sur plusieurs jeux de données associés à différents types de questions.

QA multi-lingue La grande majorité des ressources (e.g. jeux de données) disponibles aujourd'hui pour la tâche de QA sont en anglais. Ainsi, développer des systèmes dans d'autres langues comme le français voire des systèmes multilingues ou crosslingues (question et contexte dans des langues différentes) est une tâche complexe. Il existe des solutions comme l'utilisation de la traduction automatique (Asai et al., 2018), ou de modèles de langage pré-entraînés sur plusieurs langues qui permettent d'effectuer du transfert few-shot ou zero-shot de l'anglais vers d'autres langues (Siblini et al., 2019).

QA en domaine ouvert Le QA en domaine ouvert (Chen et al., 2017 ; Siblini et al., 2020) consiste à répondre à des questions en s'appuyant non pas sur un petit passage de texte mais sur une grande collection de documents traitants de sujets divers (e.g. Wikipedia). Il s'appuie généralement sur deux briques élémentaires. L'étape de **recherche d'information ad-hoc** avec un modèle *retriever* qui identifie les documents pertinents pour la question considérée. Ensuite, le *reader*, qui résout la tâche de QA

sur le domaine fermé résultant. Le retriever consiste en l'application d'une fonction de similarité entre la question et les documents, sur la base d'une représentation de ces derniers qui peut être parcimonieuse (TF-IDF, BM25 (Robertson et al., 1995)) ou dense (USE (Cer et al., 2018), Word2Vec (Mikolov et al., 2013), BERT). Parfois le retriever est décomposé en deux parties, une partie qui pré-sélectionne une centaine de documents et qui repose sur une approche simple et rapide (e.g. BM25) puis une partie *reranker* qui repose sur une approche plus complexe comme BERT.

3 QA Multilingue, Génératif et Unifié en Domaine Ouvert

Dans cette section, on introduit notre proposition pour un système de QA en domaine ouvert (ODQA) toujours plus complet, avec la capacité de traitement de l'anglais, du français et de plusieurs types de questions et la capacité de génération d'une réponse bien formée.

3.1 Retriever

Nous choisissons la stratégie populaire *retriever + reader* pour répondre au challenge de QA en domaine ouvert. Pour faire le bon choix de retriever, nous en avons considérés plusieurs qui appartiennent aux deux familles évoquées dans la section précédente : BM25 comme *retriever* parcimonieux et DRP (Karpukhin et al., 2020) et ProQA (Xiong et al., 2020) comme *retrievers* denses. Pour les comparer, nous évaluons deux tâches sur quatres jeux de données (TriviaQA, Natural Questions, SQuAD, WebQuestions) et Wikipedia : (1) la tâche de recherche d'information pure et (2) une tâche complète d'ODQA. Pour la tâche (1), l'objectif est de voir dans quelle proportion le top 5, 30, 70 et 100 des paragraphes de Wikipedia renvoyés par le retriever contient la réponse à la question donnée en entrée (cela correspond aux précisions P@5, P@30, ...). Dans la tâche (2), on considère le top 100 renvoyé et on évalue la capacité d'un reader (ici BERT) à y extraire une réponse précise (en utilisant la métrique F1 et Exact Match). Les résultats ne sont pas détaillés mais pour la tâche de recherche d'information pure, l'approche la plus précise est DPR sur trois datasets. BM25 est cependant plus efficace sur SQuAD et est environ dix fois plus rapide que les autres méthodes. Malgré l'avantage de DPR sur la partie de recherche d'information, il semble que la combinaison BM25+BERT est globalement plus efficace que DPR+BERT sur la tâche (2), sauf pour le jeu de donnée Natural Questions. Nous retenons donc BM25 pour notre approche qui apparaît comme plus rapide et robuste.

3.2 Reader

Concernant le reader, nous avons sélectionné un modèle ayant une version multilingue, une partie générative, et présentant des performances prédictives prometteuses dans un cadre unifié. Il s'agit du modèle BART avec une architecture encodeur-décodeur inspirée du Transformer, et utilisé dans l'approche UnifiedQA (Khashabi et al., 2020).

Unification et langage Notre modèle unifié est obtenu de façon similaire à UnifiedQA, par entraînement simultané sur plusieurs jeux de données associés à différents types de question. Il donc nécessaire d'identifier préalablement ces jeux de données et d'unifier leur format. Nous complétons et augmentons la liste de Khashabi et al. (2020) (RACE, Arc hard, Arc easy, AI2 science elementary, AI2 science middle, OBQA, MCTest) pour l'aspect multilingue, en ajoutant en particulier des jeux de données en français comme FQuAD (d'Hoffschmidt et al., 2020) et Fr-SQuAD (traduction automatique de SQuAD en français) et d'autres jeux de données en anglais comme Natural Question, TriviaQA et WebQuestions. Concernant le modèle, nous avons choisit une version multilingue de BART, appelée mBART (Liu et al., 2020). Il en existe plusieurs variantes où l'encodeur et le décodeur sont soit l'un (many-to-one), soit l'autre (one-to-many), soit tous les deux multilingues (many-to-many). La version many-to-many étant instable pour le fine-tuning sur la tâche de QA, nous avons décidé d'opter pour les versions many-to-fr et many-to-en (décodeurs monolingues) pour construire des modèles qui répondent respectivement en français ou en anglais. Pour entraîner ces modèles, nous construisons, à partir des données bruts, deux nouveaux ensembles de données : chacun contient l'intégralité des jeux considérés, avec l'entrée qui peut être en français ou en anglais, et avec la sortie (réponse) qui est uniquement en français dans l'un (noté output-fr) et uniquement en anglais dans l'autre (noté output-en). Pour cela, nous utilisons de la traduction automatique avec l'API de Google : par exemple, un jeu de données initialement en français (e.g. FQuAD) reste inchangé dans output-fr mais ses questions et ses réponses (pas les contextes) sont traduites en anglais dans output-en. Le modèle many-to-fr (resp. many-to-en) est entraîné sur l'ensemble de données output-fr (resp. output-en). Dans l'inférence, on utilisera soit l'un soit l'autre de ces modèles en fonction de la langue de la question de l'utilisateur. Pour détecter la langue, on utilise un module de détection de langue pré-entraîné (*langdetect*).

Génération La majorité des jeux de données pour le QA unifié proposent une réponse extraite ou très courte. A l'inverse, les données utilisées pour le QA génératif proposent des réponses bien formées, potentiellement exploitables dans une interface conversationnelle (e.g. chatbot). Pour rendre nos modèles many-to-fr et many-to-en plus verbeux, notre proposition consiste à les affiner sur un jeu de données adapté. On considère ici MS-MARCO (Nguyen et al., 2016), le jeu le plus emblématique dans le domaine du QA génératif. Notons que nous choisissons d'effectuer l'entraînement sur MS-MARCO dans un second temps et pas en simultané de l'entraînement sur les ensembles output-fr ou output-en car nous souhaitons obtenir en définitif un modèle générant des réponses bien formées. Il sera donc important d'évaluer si, après fine-tuning, le modèle conserve sa capacité à répondre correctement à tous types de questions dans les ensembles output-fr ou output-en. L'affinage est réalisé selon les paramètres d'entraînement préconisés dans la littérature pour le modèle BART sur le jeu de données MS-MARCO.

3.3 Architecture Globale

Pour combiner le retriever et le reader, nous sommes confrontés à une contrainte provenant du reader. Celui-ci est limité à une entrée de 512 tokens au plus, ce qui cor-

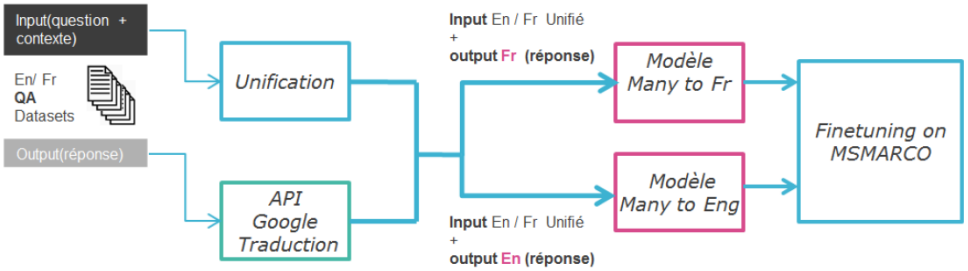


FIG. 1 – *Processus d'augmentation de données et d'entraînement des deux readers mBART many-to-en et many-to-fr.*

respond à environ 5 passages de texte renvoyés par le retriever. Cependant la précision à 5 de BM25 est très limitée (de l'ordre de 30%). Pour y remédier nous adoptons une solution ayant émergé récemment dans la littérature qui consiste à mettre en place un reranker. Plus précisément, on sélectionne dans un premier temps un nombre de passages important (e.g. 100) mais qui reste très négligeable par rapport au nombre de passages total à l'aide un retriever, puis on restreint le nombre de passages (e.g. 5) à l'aide d'un modèle plus complexe.

Reranker Pour le reranker, on utilise l'encodeur du modèle mBART auquel on ajoute une couche de classification. L'objectif est de classer une entrée (question + contexte) en vrai (ou 1) si le contexte contient la réponse, et en faux (ou 0) sinon. Pour construire un jeu de données pour ce reranker, on considère les jeux WebQuestions, Natural Questions, Squad1.1, FQuAD, FrSQuAD, TriviaQA car ils sont adaptés pour la tâche d'ODQA et en français ou en anglais. On considère uniquement les questions et réponses de ces jeux de données (on ignore le contexte) et on recherche avec BM25, pour chaque question, le top 100 des passages dans les Wikipedia français et anglais. A partir de ce top 100, on construit des exemples positifs (avec les passages qui contiennent la réponse) et on échantillonne des exemples négatifs avec une probabilité proportionnelle au score de pertinence donné par BM25. Après entraînement, le reranker obtient une précision à 5 bien meilleure que BM25, de l'ordre de 55%.

Pipeline En résumé, la phase d'entraînement nécessite un travail important de préparation et d'augmentation de données : (1) On considère au départ un nombre important de jeux de données en français et en anglais à uniformiser. (2) Pour six d'entre eux, on recherche, avec BM25 et les questions, le top 100 des passages pertinent dans Wikipedia. On les échantillonne pour créer un jeu d'entraînement pour le reranker. (3) Pour le jeux de données des deux readers, on traduit les questions et réponses dans les deux langues cibles. (4) On prépare le jeu de donnée génératif MS-MARCO sous le même format pour l'affinage. On traduit les réponses pour le reader many-to-fr.

Une fois tous les modèles entraînés, ils sont utilisés dans un processus complet d'inférence illustré par la figure 2. Lorsqu'on pose une question, celle-ci est utilisée par

le retriever et le reranker pour identifier les 5 passages les plus pertinents. En parallèle un module détecte la langue de la question entre anglais et français pour sélectionner un des readers (many-to-fr ou many-to-en). Celui-ci prend finalement en entrée la question et les 5 passages concaténés et génère une réponse pour l'utilisateur.

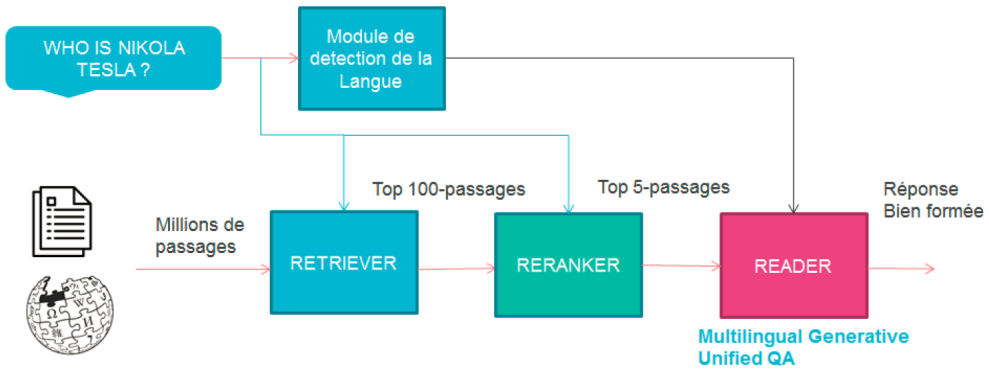


FIG. 2 – *Architecture globale du processus d'inférence proposé.*

Note : les jeux de données utilisés dans ce papier ont des ensembles d'apprentissage allant de 1000 exemples à 100 000 exemples et les caractéristiques détaillées sont disponibles, pour la plupart, dans le Tableau 2 de l'article de Khashabi et al. (2020).

4 Résultats Expérimentaux

Les résultats du retriever et du reranker sont déjà mentionnés dans la section précédente. Ici on se concentre principalement sur les performances du reader. En particulier, on analyse sa capacité à traiter la tâche unifiée et la tâche générative. Dans le tableau 1, on montre les performances de mBART many-to-en globalement sur la tâche unifiée (moyenne sur l'ensemble des jeux de données de output-en) et sur la tâche générative (MS-MARCO). Pour la tâche unifiée, on distingue les performances du modèle entraîné uniquement sur output-en et celle du modèle entraîné sur output-en puis affiné sur MS-MARCO.

TAB. 1 – *Performance globale de mBART many-to-en sur les ensembles "dev" des jeux de données de output-en et sur l'ensemble dev de MS-MARCO, lorsqu'il est uniquement entraîné sur les ensembles "train" des jeux de données de output-en ou également affiné sur l'ensemble "train" de MS-MARCO.*

	Global output-en dev			MS-MARCO dev	
	EM	Rouge-1	Rouge-L	Rouge-1	Rouge-L
Entraîné uniquement sur output-en	**0,443**	0,557	0,554		
Affiné sur MS-MARCO	0,374	**0,569**	**0,562**	0,626	0,579

Notons que les résultats présentés ici sur la partie QA unifié sont similaires à ceux obtenus par l'approche UnifiedQA (Khashabi et al., 2020). Et les résultats génératifs sur MS-MARCO sont comparables à ceux du leaderboard public. Lorsqu'on compare les deux modèles du Tableau 1, on peut remarquer que le premier obtient un meilleur Exact Match (la réponse générée doit correspondre exactement à la réponse attendue) mais les deux obtiennent un Rouge similaire. La comparaison en terme d'exact match n'est pas surprenante car le modèle génératif (affiné sur MS-MARCO) est verbeux alors que les réponses attendues dans output-en sont généralement courtes. Il est donc plus pertinent de regarder le Rouge orienté rappel qui cherche à analyser si la réponse attendue est contenue dans la réponse générée. Ce Rouge suggère que le modèle affiné reste aussi pertinent (voire légèrement plus) que le modèle entraîné sur output-en pour répondre aux questions sur cet ensemble de données.

Pour mieux percevoir la différence entre les deux modèles, on montre dans le Tableau 2 quelques exemples de leurs prédictions de réponse pour des questions de plusieurs types (Multi-Choix, Yes/No, Extractive, Abstractive). Il est intéressant de noter ici que MS-MARCO ne contient pas de questions Multi-Choix ou Yes/No mais par la dynamique d'entraînement sur output-en suivi d'un affinage sur MS-MARCO, le modèle semble capable d'être génératif sur ce type de questions.

Tab. 2 – *Exemples de réponses de mBART many-to-en lorsqu'il est uniquement entraîné sur l'ensemble de données output-en ou lorsqu'il est également affiné (fine-tuning) sur MS-MARCO.*

Type	**Question**	**Context**	**Réponse avant fine-tuning**	**Réponse après fine-tuning**
MC	Which is best conductor ?	(A) iron (B) feather	iron	iron is the best conductor.
MC	What does photosynthesis produce that helps plants grow ?	(A) water (B) oxygen (C) protein (D) sugar	sugar	photosynthesis produces sugar that helps plants grow.
Y/N	Was America the first country to have a president ?	(President) The first usage of the word president to denote the …	no	no, america was not the first country to have a president.
EX	At what speed did the turbine operate ?	(Nikola_Tesla) On his 50th birthday in 1906, Tesla demonstrated…	16,000 rpm	the turbine operated at 16,000 rpm.
EX	what is the size of amir's head ?	His head was so big with a size of 15 cm	15 cm	the size of amir's head is 15 cm.
AB	What does a drink from narcissus's spring cause the drinker to do ?	Mercury has awakened Echo, who weeps for Narcissus, and …	grow increasingly e…	a drink from narcissus's spring causes the drinker to grow…

On montre également dans le tableau 3 des résultats du modèle many-to-fr, qui suggèrent que ce dernier parvient à générer une réponse en français même lorsque la question et le contexte lui sont donnés en anglais.

TAB. 3 – *Exemples de réponses de mBART many-to-fr lorsqu'il est uniquement entraîné sur l'ensemble de données output-fr ou lorsqu'il est également affiné (fine-tuning) sur MS-MARCO.*

Type	**Question**	**Context**	**Réponse avant fine-tuning**	**Réponse après fine-tuning**
Y/N	Was America the first country to have a president ?	(President) The first usage of the word president to denote the highest …	non	non, l'amérique n'était pas le premier pays à avoir un président
EX	At what speed did the turbine operate ?	(Nikola_Tesla) ... Tesla demonstrated ... 16,000 rpm bladeless turbine.	16 000 rpm	la turbine a fonctionné à une vitesse de 16 000 roulements par minute.
MC	What does photosynthesis produce that helps plants grow ?	(A) water (B) oxygen (C) protein (D) sugar	sucre	la photosynthèse produit du sucre qui aide les plantes à grandir.
AB	What does a drink from narcissus's spring cause the drinker to do ?	… a drink from Narcissus's spring causes the drinkers to "Grow dotingly enamored of themselves.	se sentir amoureux de eux-mêmes	une boisson du printemps de narcisse provoque le buveur de se sentir amoureux de eux-mêmes

5 Application

La capacité à traiter le français et l'anglais, à répondre à plusieurs types de questions, en domaine ouvert sur Wikipedia, et à générer des réponses bien formées font du processus proposé un atout intéressant pour une intégration dans un agent dialoguant automatique comme un chatbot. Nous avons donc souhaité implémenter ce processus sous forme d'une API avec la libraire python Flask, et développé une interface conversationnelle qui interroge cette API. Un exemple d'interaction via l'interface est montrée sur la Figure 3. L'API est hébergée sur un serveur avec 32 Gb de RAM et 1 GPU Tesla V100. Le modèle interroge l'intégralité des Wikipédia anglais et français ce qui représente environ 50 millions de paragraphes et la réponse est obtenue en temps réel (environ une seconde).

6 Conclusion

Dans ce papier, nous proposons la première approche de QA qui est à la fois générative, multilingue, unifiée et en domaine ouvert. L'approche repose sur un travail conséquent autour de la préparation de jeux de données, de la sélection d'algorithmes pertinents de la littérature et sur une dynamique d'entraînement et d'affinage adaptée. Elle peut être appliquée pour la recherche d'information en domaine ouvert avec des sources et des questions variées. Dans nos futurs travaux, nous envisageons (1) d'évaluer avec plus de détails cette technique et de la comparer avec d'autres approches du

FIG. 3 – *Exemple d'interaction avec le chatbot basé sur l'approche proposée.*

domaine comme par exemple GenQA (Muller et al., 2021), et (2) d'analyser la dynamique d'entraînement/affinage et d'ajouter des éléments de complexité supplémentaires (conversationnel, multi-hop) avec une méthodologie similaire.

Références

Asai, A., A. Eriguchi, K. Hashimoto, et Y. Tsuruoka (2018). Multilingual extractive reading comprehension by runtime machine translation. *arXiv preprint arXiv :1809.03275*.

Bi, B., C. Li, C. Wu, M. Yan, W. Wang, S. Huang, F. Huang, et L. Si (2020). Palm : Pre-training an autoencoding&autoregressive language model for context-conditioned generation. *arXiv preprint arXiv :2004.07159*.

Cer, D., Y. Yang, S.-y. Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Céspedes, S. Yuan, C. Tar, et al. (2018). Universal sentence encoder. *arXiv preprint arXiv :1803.11175*.

Chen, D., A. Fisch, J. Weston, et A. Bordes (2017). Reading wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pp. 1870–1879.

Devlin, J., M.-W. Chang, K. Lee, et K. Toutanova (2019). Bert : Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186.

d'Hoffschmidt, M., W. Belblidia, T. Brendlé, Q. Heinrich, et M. Vidal (2020). Fquad : French question answering dataset. *arXiv preprint arXiv :2002.06071*.

Karpukhin, V., B. Oğuz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, et W.-t. Yih (2020). Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv :2004.04906*.

Khashabi, D., S. Min, T. Khot, A. Sabharwal, O. Tafjord, P. Clark, et H. Hajishirzi (2020). Unifiedqa : Crossing format boundaries with a single qa system. *arXiv preprint arXiv :2005.00700*.

Lewis, M., Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, et L. Zettlemoyer (2019). Bart : Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv :1910.13461*.

Liu, J., Y. Lin, Z. Liu, et M. Sun (2019). Xqa : A cross-lingual open-domain question answering dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 2358–2368.

Liu, Y., J. Gu, N. Goyal, X. Li, S. Edunov, M. Ghazvininejad, M. Lewis, et L. Zettlemoyer (2020). Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics 8*, 726–742.

Manning, C. D., H. Schütze, et P. Raghavan (2008). *Introduction to information retrieval*. Cambridge university press.

Mikolov, T., I. Sutskever, K. Chen, G. S. Corrado, et J. Dean (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pp. 3111–3119.

Muller, B., L. Soldaini, R. Koncel-Kedziorski, E. Lind, et A. Moschitti (2021). Cross-lingual genqa : A language-agnostic generative question answering approach for open-domain question answering. *arXiv preprint arXiv :2110.07150*.

Nguyen, T., M. Rosenberg, X. Song, J. Gao, S. Tiwary, R. Majumder, et L. Deng (2016). Ms marco : A human generated machine reading comprehension dataset. In

CoCo@ NIPS.

Qu, C., L. Yang, C. Chen, M. Qiu, W. B. Croft, et M. Iyyer (2020). Open-retrieval conversational question answering. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pp. 539–548.

Rajpurkar, P., J. Zhang, K. Lopyrev, et P. Liang (2016). Squad : 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2383–2392.

Robertson, S. E., S. Walker, S. Jones, M. M. Hancock-Beaulieu, M. Gatford, et al. (1995). Okapi at trec-3. *Nist Special Publication Sp 109*, 109.

Siblini, W., M. Challal, et C. Pasqual (2020). Delaying interaction layers in transformer-based encoders for efficient open domain question answering. *arXiv preprint arXiv :2010.08422*.

Siblini, W., C. Pasqual, A. Lavielle, et C. Cauchois (2019). Multilingual question answering from formatted text applied to conversational agents. *arXiv preprint arXiv :1910.04659*.

Siblini, W., B. Sayil, et Y. Kessaci (2021). Towards a more robust evaluation for conversational question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2 : Short Papers)*, pp. 1028–1034.

Tan, C., F. Wei, N. Yang, B. Du, W. Lv, et M. Zhou (2017). S-net : From answer extraction to answer generation for machine reading comprehension. *arXiv preprint arXiv :1706.04815*.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, et I. Polosukhin (2017). Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008.

Xiong, W., H. Wang, et W. Y. Wang (2020). Progressively pretrained dense corpus index for open-domain question answering. *arXiv preprint arXiv :2005.00038*.

Summary

QA systems have evolved in several directions over the past years: extractive QA, open domain QA, multilingual QA, unified QA, etc. In this paper, our objective is to bring together these directions by proposing an approach that behaves satisfyingly for each of them. More precisely, we propose the first unified generative multilingual open domain question answering system. It is based on the selection of appropriate existing building blocks, on the preparation and augmentation of datasets, and on the design of a global training dynamic. It obtains a competitive performance on several subtasks and provides an interesting behavior on the crossing of some tasks. We finally implement it in a dialog agent whose objective is to answer to user questions from Wikipedia.

Index