Actes de la 6e conférence conjointe Journées d'Études sur la Parole (JEP, 33e édition),
Traitement Automatique des Langues Naturelles (TALN, 27e édition),
Rencontre des Étudiants Chercheurs en Informatique pour le Traitement Automatique des Langues (RÉCITAL, 22e édition)
Nancy, France, 08-19 juin 2020

DEFT 2020 : détection de similarité entre phrases et extraction d'information

Mike Tapi Nzali[1]
(1) Reezocar, 20 Rue d'issy 92100 Boulogne-Billancourt, France
mike@reezocar.com

RÉSUMÉ

Ce papier décrit la participation de Reezocar à la campagne d'évaluation DEFT 2020. Cette seizième édition du challenge a porté sur le calcul de similarité entre phrases et l'extraction d'information fine autour d'une douzaine de catégories dans des textes rédigés en Français. Le challenge propose trois tâches : (i) la première concerne l'identification du degré de similarité entre paires de phrases ; (ii) la deuxième concerne l'identification des phrases parallèles possibles pour une phrase source et (iii) la troisième concerne l'extraction d'information. Nous avons utilisé des méthodes d'apprentissage automatique pour effectuer ces tâches et avons obtenu des résultats satisfaisants sur l'ensemble des tâches.

ABSTRACT

DEFT 2020 : sentence similarity detection and information retrieval

This paper describes Reezocar's participation in the DEFT 2020 evaluation campaign. This sixteenth edition of the challenge focused on the calculation of similarity between sentences and the extraction of fine information around a dozen categories in texts written in French. The challenge proposes three tasks : (i) the first concerns the identification of the degree of similarity between pairs of sentences ; (ii) the second concerns the identification of possible parallel sentences for a source sentence ; and (iii) the third concerns the extraction of information. We used machine learning methods to perform these tasks and obtained satisfactory results on all tasks.

MOTS-CLÉS : détection de similarité sémantique, extraction d'information, apprentissage automatique.

KEYWORDS: semantic similarity detection, information extraction, machine learning.

1 Introduction

L'édition 2020 du DÉfi Fouille de Textes (DEFT) porte sur l'exploration des cas cliniques rédigés en langue française (Cardon *et al.*, 2020). Le challenge de cette édition a pour nature l'extraction d'information fine autour d'une douzaine de catégorie, mais aussi la détection de similarité sémantique entre phrases. Le défi repose sur deux corpus. Le premier corpus CAS est constitué de cas cliniques, de plusieurs pays francophones, concernant diverses spécialités médicales (cardiologie, urologie, oncologie, obstétrique, pulmonaire, gasto-entérologie, etc.) (Grabar *et al.*, 2018). Le deuxième corpus est issu du projet CLEAR, il contient des textes issus de trois sources différentes (articles

Atelier DÉfi Fouille de Textes, pages 91–96.
HAL : hal-02784745.

d'encyclopédie, notices de médicaments, et résumés Cochrane) (Grabar & Cardon, 2018). Chaque source met à disposition des versions techniques et simplifiées sur un sujet donné en français.

Dans cet article, nous présentons les méthodes utilisées lors de l'édition 2020 du DEFT. Nous avons participé aux 3 tâches proposées. La première tâche consiste à identifier le degré de similarité entre paires de phrases parallèles et non-parallèles. Cette tâche a pour objectif de déterminer le niveau de similarité entre paires de phrases sur une échelle de 0 à 5. La deuxième tâche consiste à identifier les phrases parallèles possibles pour une phrase source, son objectif étant, pour une phrase source donnée et plusieurs phrases cibles fournies, d'identifier parmi les phrases cibles celle qui est parallèle. Ces deux tâches s'effectuent sur le corpus issu du projet CLEAR. La troisième tâche est une tâche d'extraction d'information, son objectif étant de repérer dans les cas cliniques les informations fines telles que les *pathologies*, les *signes* ou les *symptômes cliniques*. . .Cette tâche s'effectue sur le corpus CAS.

Le reste de l'article sera organisé comme suit. La section 2 décrit l'approche utilisée et présente les résultats obtenues sur les tâches 1 et 2. La section 2 décrit l'approche utilisée et présente les résultats obtenues sur la tâche 3. Enfin, la section 4 conclut ce travail.

2 Méthodes pour les tâches 1 et 2

2.1 Approches

Les tâches 1 et 2 sont respectivement des tâches de similarité entre deux phrases et d'appariements entre une phrase source et plusieurs phrases cibles. Les méthodes proposées ici sont basées sur les vecteurs de poids tf-idf des mots du corpus et sur les vecteurs de poids des mots selon leur contexte. Le principe général de la méthode est de créer un ensemble de descripteurs. Ces derniers, sont les paramètres du modèle d'apprentissage. Nous avons créé plusieurs descripteurs statistiques à partir des phrases du corpus.

Vecteur de poids tf-idf. Pour chaque phrase du corpus, nous créons son vecteur tf-idf. Ensuite, nous calculons les distances entre les vecteurs des phrases pour créer nos descripteurs. Ici, les distances utilisées sont les suivantes : la *similarité cosinus*, la *distance de Manhattan*, la *distance euclidienne* et la *distance de Jaccard*. À l'issue de cette étape, nous avons donc 4 descripteurs : $cosine_{tfidf}$, $manhattan_{tfidf}$, $euclidienne_{tfidf}$, $jaccard_{tfidf}$.

Vecteur de poids selon le contexte. Nous avons aussi utilisé la méthode de transformeurs proposée dans (Reimers & Gurevych, 2019) pour transformer chaque phrase en vecteur de poids suivant le contexte des mots dans les phrases. Pour ce faire, nous avons utilisé BERT (Devlin *et al.*, 2018) et Roberta (Liu *et al.*, 2019). Ensuite, nous avons utilisé les mêmes distances présentées précédemment pour créer des descripteurs supplémentaires. À l'issue de cette étape, nous avons obtenus les 4 descripteurs suivants : $cosine_{transformer}$, $manhattan_{transformer}$, $euclidienne_{transformer}$, $jaccard_{transformer}$

Descripteurs supplémentaires. À ces 8 descripteurs, nous avons ajouté les descripteurs suivants : la longueur de chaque phrase, la différence de longueur entre les phrases et le rapport de longueurs entre les deux phrases. Nous prenons ensuite les descripteurs obtenues, et les utilisons afin d'entraîner nos modèles. Ainsi, nous allons prédire *la valeur de similarité* pour la tâche 1 et *la cible* pour la tâche 2 avec nos descripteurs.

Choix des classifieurs. Comme méthode d'apprentissage, nous avons utilisé XGBoost (*eXtreme Gradient Boosting*) (Chen & Guestrin, 2016). Il s'agit d'une implémentation open source optimisée de l'algorithme d'arbres de boosting de gradient qui utilise des approximations plus précises pour trouver le meilleur modèle d'arbre. Il utilise un certain nombre d'astuces qui lui confèrent un succès exceptionnel, en particulier avec des données structurées.

2.2 Résultats et discussions

Pour évaluer nos modèles, nous avons utilisé l' EDRM (Exactitude en Distance Relative à la solution Moyenne) (Grouin *et al.*, 2013) pour la tâche 1 et la MAP (*Mean Average Precision*) pour la tâche 2. Par rapport à d'autres participants, nous sommes au-dessus des moyennes et des médianes.

Sur la tâche 1, nous obtenons le meilleur score avec le *run* 2, son score EDRM de 81% (voir table 1). La médiane de l'EDRM de l'ensemble des systèmes soumis est de $79,5\%$. Cependant, le meilleur système du défi sur cette tâche a obtenu une valeur de $82,2\%$.

Sur la tâche 2, notre meilleur score est obtenu par le *run* 1 (voir table 2) avec une MAP de 98.7%, la médiane de la MAP de l'ensemble des systèmes soumis est de $98,68\%$. Cependant, le meilleur système du défi sur cette tâche a obtenu une valeur de $99,06\%$.

	Tâche 1		
	EDRM	Spearman Correlation	p-value
Run 1	0.792	0.706	4.15e-63
Run 2	**0.810**	**0.735**	**6.60e-71**
Run 3	0.802	0.708	1.28e-63

TABLE 1 – Résultats des différents *runs* soumis au défi pour la tâche 1

	Tâche 2
	MAP
Run 1	**0.987**
Run 2	0.981
Run 3	0.985

TABLE 2 – Résultats des différents *runs* soumis au défi pour la tâche 2

On remarque que sur la 2ème tâche, les résultats des différents *runs* 2 sont assez proches, cela s'explique par le fait que la différence entre les runs se situe au niveau du paramétrage des modèles.

3 Tâches 3

3.1 Approches

Pour effectuer cette tâche, nous avons utilisé les champs aléatoires conditionnels (CRF (Lafferty *et al.*, 2001)). Il s'agit de l'une des approches les plus efficaces pour l'étiquetage supervisé de séquences. Ils ont été appliqués avec succès pour des tâches telles que l'étiquetage morphosyntaxique (Lafferty *et al.*, 2001), l'extraction d'entités nommées (McCallum & Li, 2003).

Les CRF sont des modèles probabilistes graphiques non dirigés, conçus pour définir une distribution de probabilités conditionnelles sur des séquences d'étiquettes, étant donnée des séquences observées. Cette nature conditionnelle démarque les CRF des modèles qui nécessitent une hypothèse d'indépendance des variables, tels que les modèles de Markov cachés (HMM) (Blunsom, 2004). En pratique, une qualité des modèles CRF est leur robustesse sur des ensembles de données de petite taille. Nous avons appliqué plusieurs pré-traitements comme effectué dans (Tapi Nzali *et al.*, 2015). Nous avons créé un modèle CRF pour cette tâche et avons extrait plusieurs traits d'ordre morphologique, syntaxique et sémantique pour l'apprentissage. Les différentes descripteurs construits sont les suivantes : *Capitalisation du token*, *Longueur du token*, *Présence d'un chiffre dans le token*, *Présence de ponctuation dans le token*.

3.2 Résultats et discussions

Nous présentons les résultats obtenus dans les tables 3 et 4. Nous remarquons que, sur certaines entités nommées, nous obtenons de bon résultats comparés à d'autres. C'est le cas de *dose* et *traitement* pour lesquelles le modèle est moins performant.

	TP	FP	FN	Precision	Recall	F1
pathologie	73	79	93	0,4803	0,4398	0,4591
sosy	500	418	779	0,5447	0,3909	0,4552
Overall	573	497	872	0,5355	0,3965	0,4557

TABLE 3 – Résultats du *run 3* soumis au défi pour la tâche 3.1

	TP	FP	FN	Precision	Recall	F1
anatomie	684	214	436	0,7617	0,6107	0,6779
dose	10	15	42	0,4000	0,1923	0,2597
examen	342	301	475	0,5319	0,4186	0,4685
mode	42	11	47	0,7925	0,4719	0,5915
moment	93	36	72	0,7209	0,5636	0,6327
substance	172	93	141	0,6491	0,5495	0,5952
traitement	99	105	205	0,4853	0,3257	0,3898
valeur	302	53	130	0,8507	0,6991	0,7675
Overall	1744	828	1548	0,6781	0,5298	0,5948

TABLE 4 – Résultats du *run 3* soumis au défi pour la tâche 3.2

Sur la tâche 3, nous avons fournit 3 *runs* et ne présentons ici que celui avec lequel nous avons le meilleur résultat.

4 Conclusion

Nous avons participé aux 3 tâches proposées dans ce DÉfi Fouille de Textes (DEFT2020). Globalement, nos résultats sont assez satisfaisants, car nous sommes tout proches des meilleurs du challenge. Nous ne sommes malheureusement pas parvenus à tester des approches supplémentaires et performantes sur la tâche 3 du défi DEFT 2020 par manque de temps. Nos résultats proviennent d'approches classiques et conduisent à des résultats sans surprise. Pour améliorer les résultats de la tâche 3, Il aurait été intéressant de combiner les approches LSTM (Long Short-Term Memory) et les CRF (Huang *et al.*, 2015).

Références

BLUNSOM P. (2004). Hidden markov models. *Lecture notes, August*, **15**(18-19), 48.

CARDON R., GRABAR N., GROUIN C. & HAMON T. (2020). Présentation de la campagne d'évaluation deft 2020 : similarité textuelle en domaine ouvert et extraction d'information précise dans des cas cliniques. In *Actes DEFT*.

CHEN T. & GUESTRIN C. (2016). Xgboost : A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, p. 785–794.

DEVLIN J., CHANG M.-W., LEE K. & TOUTANOVA K. (2018). Bert : Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv :1810.04805*.

GRABAR N. & CARDON R. (2018). Clear-simple corpus for medical french.

GRABAR N., CLAVEAU V. & DALLOUX C. (2018). Cas : French corpus with clinical cases. In *Proceedings of the Ninth International Workshop on Health Text Mining and Information Analysis*, p. 122–128.

GROUIN C., ZWEIGENBAUM P. & PAROUBEK P. (2013). Deft2013 se met à table : présentation du défi et résultats. *Actes du neuvième DÉfi Fouille de Textes*, p.$\tilde{2}$.

HUANG Z., XU W. & YU K. (2015). Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv :1508.01991*.

LAFFERTY J., MCCALLUM A. & PEREIRA F. C. (2001). Conditional random fields : Probabilistic models for segmenting and labeling sequence data.

LIU Y., OTT M., GOYAL N., DU J., JOSHI M., CHEN D., LEVY O., LEWIS M., ZETTLEMOYER L. & STOYANOV V. (2019). Roberta : A robustly optimized bert pretraining approach. *arXiv preprint arXiv :1907.11692*.

MCCALLUM A. & LI W. (2003). Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, p. 188–191 : Association for Computational Linguistics.

REIMERS N. & GUREVYCH I. (2019). Sentence-bert : Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing* : Association for Computational Linguistics.

TAPI NZALI M. D., TANNIER X. & NÉVÉOL A. (2015). Automatic extraction of time expressions across domains in french narratives.