Comparaison de l'efficacité des Word Embeddings dans le domaine juridique

Kervin Prinville 2016202, Patrick Cobanovic 1931859
Polytechnique de Montréal

Abstract

Les documents juridiques utilisent une terminologie spécifique qui peut porter des significations distinctes de celles dans un contexte non spécialisé. Ceci rend leur traitement automatique plus complexe. En effet, cette terminologie nécessite des approches informatiques adaptées pour une analyse efficace. Cette recherche démontre l'application des word embeddings, notamment FastText et Word2Vec, visant à améliorer la compréhension et le traitement des textes juridiques.

1 Introduction

L'analyse automatique des documents juridiques est de plus en plus importante dans ce domaine. Celle-ci est devenue nécessaire à cause de la croissance du volume de textes juridiques numériques ainsi que l'avantage qu'apporte les outils de recherches dans ces millions de documents.

2 Travaux connexes

Le domaine du traitement automatique du langage naturel est en évolution rapide. Avoir une compréhension des travaux existants est important pour débuter de nouvelles recherches. Dans cette optique, nous examinons deux travaux connexes qui illustrent l'utilisation de word embeddings dans notre domaine de recherches et qui nous a inspiré dans notre enquête sur l'application des word embeddings aux documents juridiques.

2.1 Word and Sentence Embeddings in Legal NLP

Le domaine juridique présente des défis uniques en matière de traitement du langage naturel en raison de son vocabulaire spécialisé et de ses structures de phrases complexes. Jayasinghe et coll. (2022) explorent les intégrations de phrases dans le domaine juridique à l'aide d'un modèle d'auto-encodage en combinaison avec des intégrations spécifiques au droit [1]. Leurs travaux mettent en évidence la nécessité de techniques adaptées au domaine pour améliorer la fonctionnalité des intégrations dans les applications juridiques, en s'alignant étroitement sur FastText et Word2Vec[1].

2.2 Transformers and Legal Text Analysis

Oliveira et Nascimento (2023) étudient l'utilisation de modèles basés sur des transformateurs tels que BERT, GPT-2 et RoBERTa, spécifiquement pré-entraînés et affinés sur les documents juridiques du système judiciaire brésilien [2]. Ils démontrent les performances supérieures de ces modèles dans la détection des similitudes entre les documents juridiques, suggérant le potentiel des techniques avancées de PNL pour améliorer le traitement des textes juridiques. Cette recherche souligne la pertinence d'adapter les modèles de PNL au domaine juridique[2].

3 Approche Théorique

Les word embeddings sont le résultat de modèles de traitement du langage naturel qui mappent des mots dans un espace vectoriel où

les mots de contextes similaires sont proches les uns des autres, selon certaines mesures de similarité. La figure 1 montre un exemple de vecteurs de mots dans un espace 3d. On peut voir que man→woman est très similaire à king→queen.

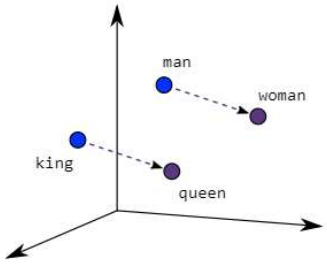

Figure 1: Exemple de vecteur[3]

3.1 Word2Vec et FastText

Le modèle Word2Vec a été développé par Google[4]. Celui-ci apprend à associer des mots à des vecteurs en utilisant deux architectures principales, Skip-gram et Continuous Bag of Words (CBOW)[4]. Le Skip-gram cherche à maximiser la classification correcte d'un mot contextuel *wi* basé sur le mot source *wt*, utilisant le produit de similarité point entre les vecteurs correspondants comme exprimé dans la fonction de probabilité conditionnelle suivante :

$$p(w_i|w_t) = \frac{\exp(\mathbf{v}_{w_i}^\top \mathbf{v}_{w_t})}{\sum_{j=1}^{V} \exp(\mathbf{v}_{w_j}^\top \mathbf{v}_{w_t})}$$

où v_{wi} et v_{wt} sont les vecteurs d'embeddings pour les mots contextuels et le mot cible, respectivement, et *V* est le vocabulaire. CBOW, quant à lui, prédit un mot cible en fonction du contexte constitué par un ensemble de mots environnants[4]. Le modèle CBOW prend les vecteurs d'embeddings pour les mots contextuels, les combine en les moyennant, et utilise ce vecteur résultant pour prédire le mot cible. La probabilité d'un mot cible *wt* étant donné le contexte *C* est définie par :

$$p(w_t|C) = \frac{\exp({\mathbf{v}'_{w_t}}^\top \cdot \mathbf{h})}{\sum_{j=1}^{V} \exp({\mathbf{v}'_{w_j}}^\top \cdot \mathbf{h})}$$

où **h** est le vecteur caché obtenu en moyennant les vecteurs d'embedding des mots de contexte *C*[4]. Le vecteur h est donnée par:

$$\mathbf{h} = \frac{1}{2m} \sum_{w \in C} \mathbf{v}_w$$

La figure 2 suivante explique les deux architectures de façon simple.

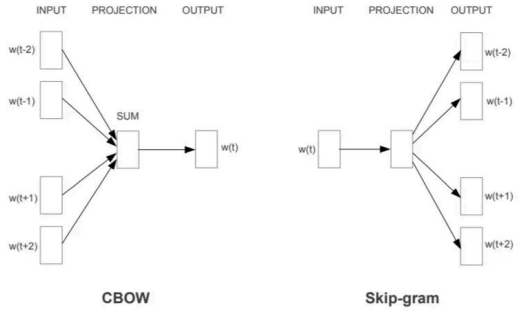

Figure 2: Architectures de Word2Vec[4]

FastText, élaboré par Facebook AI Research, continue sur l'approche Word2Vec en prenant en compte les sous-chaînes de caractères d'un mot, capturant ainsi la morphologie des mots. Un mot est représenté comme un sac de ses n-grammes de caractères, et un vecteur pour un mot est obtenu par l'agrégation des vecteurs de ses n-grammes.

3.2 Préparation du Dataset

Pour cette recherche, nous avons utilisé le Case Law Access Project qui offre un accès aux décisions de justice historiques à travers les États-Unis. Nous avons séparé la préparation du texte en plusieurs étapes différentes.

Premièrement, nous avons choisi 1 volume au hasard des Appellate Division Reports de New York, totalisant des milliers d'opinions de la cour et fournissant ainsi un corpus focalisé et pertinent pour l'analyse juridique[5].

Ensuite, nous avons appliqué certaines expressions pour remplacer les dates complètes et les identifiants de cas par des marqueurs génériques. Cela nous a permis de standardiser les références temporelles et éventuelles qui pourraient brouiller l'analyse. La ponctuation est ensuite supprimée pour ne laisser que des mots, des chiffres et des caractères spéciaux qui pourraient être significatifs.

La tokenisation est alors effectuée pour découper le texte en mots individuels, et ces tokens sont tous convertis en minuscules pour garder une cohérence dans le texte.

Les stopwords fréquents sont ensuite retirés puisqu'ils ne portent pas beaucoup d'informations pertinentes pour notre analyse. L'étape suivante consiste à appliquer le stemming, où les mots sont réduits à leur racine ou forme de base.

Enfin, nous éliminons les mots peu fréquents, conservant uniquement ceux qui apparaissent plus souvent que le seuil minimal que nous avons défini. Cette filtration affine le corpus en se concentrant sur le vocabulaire pertinent.

Une fois le corpus prétraité à l'aide des étapes précédentes, nous entraînons les modèles Word2Vec et FastText. On peut ensuite utiliser ces modèles entraînés pour notre analyse sémantique sur les textes juridiques.

4 Résultats

Une analyse des paires de mots a été effectuée sur chaque modèle grâce à la fonction evaluate_word_pairs() fournie par gensim. Nous avons utilisé l'ensemble de test wordsim353.txt aussi fourni pour la validation des résultats. Cette analyse retourne le coefficient de corrélation Pearson, ainsi que la P-value qui est associée.

Word2Vec	Values	Significance
Pearson Correlation	0.321074	0.243268
P-Value	0.35094	0.199655
Confidence (%)	95.7507	

Figure 3: Correlation de Pearson du modèle Word2Vec

FastText	Values	Significance
Pearson Correlation	0.366311	0.17931
P-Value	0.247091	0.374628
Confidence (%)	95.75071	

Figure 4: Correlation de Pearson du modèle FastText

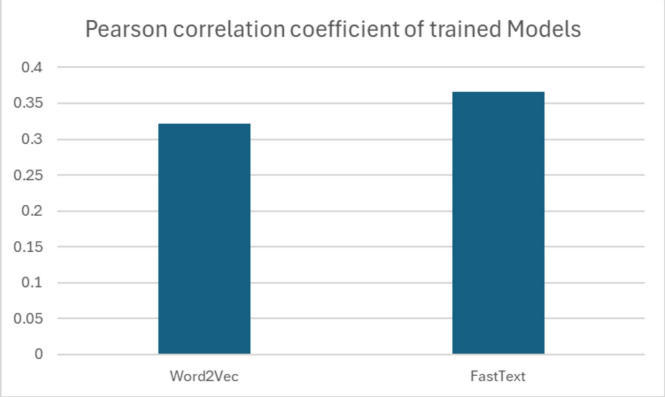

Figure 5: Comparaison des corrélations de Pearson des modèles entraînés

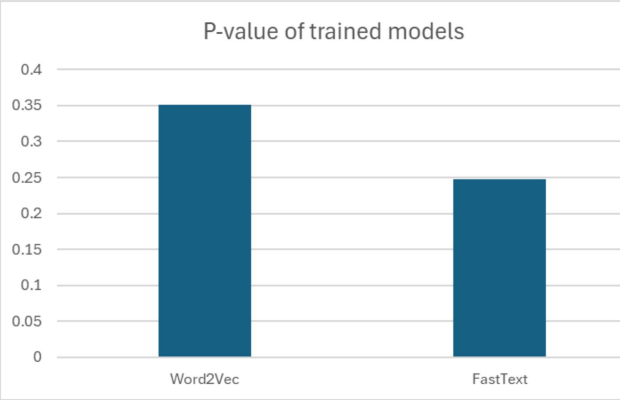

Figure 6: Comparaison des P-value de Pearson des modèles entraînés

De plus, nous avons effectué des tests sur certains ensembles de mots à titre démonstratif pour montrer les 3 meilleurs résultats de ces manipulations de vecteurs.

A	B	C	W2V	FastText
complaint	hospit	caus	victim,use,condit	accid,use,obtain
defend	plaintiff	judgment	counterclaim,reinstat,judgment	insofar,ensu,reinstat
recommend	warn	damag	damag,recov,malpractic	injur,sustain,causal
file	divorc	statement	evid,view,suffici	favor,identif,suffici
motion	appeal	suprem	westchest,queen,king	queen,king,peter

Figure 7: Exemples d'observations sémantiques.

5 Analyse

Plusieurs facteurs ont limité notre approche méthodologique. Nous avons omis d'évaluer GloVe, qui est un autre modèle de word embedding. Sa comparaison aurait pu nous offrir une perspective plus complète sur la performance de ceux que nous avons utilisés. De plus, nous nous sommes limités à 1 volume de cas juridiques. Nous aurions pu, par exemple, utiliser les 157 volumes de Appellate Division Reports de New York pour enrichir davantage nos résultats. Nous nous sommes aussi concentrés sur l'état de New York, mais il y avait des millions de textes pour les différents états. Ces éléments soulignent la nécessité d'un corpus plus étendu et d'une exploration comparative plus large pour des recherches futures, afin de saisir pleinement la portée des word embeddings dans le domaine légal.

6 Conclusion

Pour conclure, cette recherche a exploré la capacité des modèles Word2Vec et FastText à naviguer dans le domaine juridique avec l'aide de textes d'opinion de la cour. Les résultats indiquent que l'utilisation d'embeddings entraînés avec ces textes offre un avantage pour le traitement de la terminologie et concept légaux. FastText a montré une supériorité lègère, mais cela pourrait être dû au aspects introduit dans l'analyse. Une recherche plus poussée pourrait trouver des systèmes de recherches juridiques plus précis ou des outils dont les professionnels du droit pourraient prendre avantage.

7 Références

[1] S. Jayasinghe, L. Rambukkanage, A. Silva, N. de Silva, S. Perera, and M. Perera, "Learning Sentence Embeddings In The Legal Domain with Low Resource Settings," in *Proceedings of the 36th Pacific Asia Conference on Language, Information and Computation*, S. Dita, A. Trillanes, and R. I. Lucas, Eds., Manila, Philippines: Association for Computational Linguistics, Oct. 2022, pp. 494–502. Accessed: Apr. 29, 2024. [Online]. Available: https://aclanthology.org/2022.paclic-1.55

[2] R. S. de Oliveira and E. G. S. Nascimento, "Analysing similarities between legal court documents using natural language processing approaches based on Transformers." arXiv, May 11, 2023. Accessed: Apr. 29, 2024. [Online]. Available: http://arxiv.org/abs/2204.07182

[3] D. Jatnika, M. A. Bijaksana, and A. A. Suryani, "Word2Vec Model Analysis for Semantic Similarities in English Words," *Procedia Comput. Sci.*, vol. 157, pp. 160–167, Jan. 2019, doi: 10.1016/j.procs.2019.08.153.

[4] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient Estimation of Word Representations in Vector Space.” arXiv, Sep. 06, 2013. Accessed: Apr. 29, 2024. [Online]. Available: http://arxiv.org/abs/1301.3781
[5] “Read Caselaw | Caselaw Access Project.” Accessed: Apr. 29, 2024. [Online]. Available: https://case.law/caselaw/