Unsupervised Identification of Relevant Cases & Statutes Using Word Embeddings

Soumil Mandal[1] and Sourya Dipta Das[2] *

[1] SRM University, Chennai, India
[2] Jadavpur University, Kolkata, India
{soumil.mandal, dipta.juetce}@gmail.com

Abstract. In this paper, we have described the systems that we submitted as team JU_SRM for FIRE 2019 track on Artificial Intelligence for Legal Assistance (AILA 2019). The two tasks in this track were 1) identifying relevant prior cases and 2) identifying relevant statutes. For both of these tasks, we took an unsupervised approach using pre-trained word-embeddings for encoding texts and calculating relevance using cosine-similarity between the query and target documents.

Keywords: Artificial Intelligence · Legal Assistance · Sent2Vec · FastText · BERT

1 Introduction

Similar to a lot of other practical domains, the domain of law and legality is gradually incorporating automated methods as well, especially after the rapid growth and development in machine learning and information retrieval models. To encourage researchers in delving into such automated methods, FIRE 2019 included a track named Artificial Intelligence for Legal Assistance (AILA) [11]. In a lot of countries, when a lawyer is presented with a case, the final verdict is generally based on two things, 1) statutes (established laws) and 2) precedents (prior cases). The statutes informs the lawyer regarding applying legal principals based on a certain situation, while precedents informs the lawyer about how similar cases were dealt with in the past. If this pipeline of collecting relevant statutes and precedents can be automated as a information retrieval based model, this will not only help the lawyer but as well as several other people including the clients and subjects. Motivated by this, the organizers of AILA added two tasks, 1) identifying relevant prior cases for a given situation and 2) identifying most relevant statutes for a given situation. The goal was to given a case description as query, rank the target documents prior to relevancy. The datasets which the organizers provided consisted of 2914 prior cases, 197 statues and 50 queries, which were summarized case descriptions. To test our model prior to the final run submissions, the organizers provided us with the task 1 and task 2 outputs of the first 10 queries. To build our systems, we took an unsupervised approach using text-embeddings and cosine-similarity. The primary motivation behind this was semantic level modelling and better scaling. Before building our systems, we performed some basic NER removal using Spacy [3] with the following tags PERSON, ORDINAL, CARDINAL, WORK_OF_ART, TIME, PERCENT, QUANTITY.

2 Related Work

In the legal domain, researchers have contributed in several problems like text classification, text summarising and information mining Gonalves et.al [2] showed how some linguistic techniques like lemmatization and POS identification can be used to increase accuracy of the model for classification of legal texts with low dimensional feature vector. As we know, legal documents follow a certain structure of information written with formal languages and defined terminologies, researchers have tried to use these prior structural information to summarize the data which can be useful for other tasks like case recommendations, document classification and labeling. Saravanan et.al [3][6] have contributed in legal document summarizing in their subsequent works by using graphical models and CRFs. Conrad et.al [5] introduced a query based sentiment summarization for legal texts which can be very useful for mining the opinions. In the legal document labeling problem, Schweighofer et.al [1] proposed a solution by using hierarchical self-organizing map and Mencia et.al [4] introduced a multilabel classification using one vs all classifiers with efficient perceptron algorithms.

3 Task 1 - Identifying Relevant Prior Cases

For this task, the goal was to identify the relevant prior case from a collection of past cases. Here, we have used two types of word embeddings, namely pretrained Sent2Vec [9] and FastText [10] trained on the prior 2914 cases. On a whole, we created three models. For the first two models, we used a simple algorithm where the queries and precedents were encoded using pretrained word-embeddings. Instead of encoding whole cases or queries as a single long vector, we extracted sentences of max 20 tokens, then encoded each of them, and finally calculated the average of all these vectors to get the vector of the respective case or query of size 20. Then, for each query-case pair, we computed the cosine-similarity score and the ranked them accordingly. The only difference was that for the first model, we used Sent2Vec, while in the second we used FastText. We tested both of these models on the training data. The Sent2Vec model secured an BPREF of 0.0215 while the FastText model secured an BPREF of 0.0124. Using these values, we calculated weights of the models. For Sent2Vec, it was 0.0215/(0.0215 + 0.0124) = 0.63 and for FastText it was 0.0124/(0.0215 + 0.0124) = 0.36. With

[3] https://spacy.io/api/annotationnamed-entities

these values, we created our third model, which was a weighted voting ensemble model. The performance metrics [4] of all of these models on the testing data is shown below in Table 1. 1/Ro1R denotes 1/(rank of first relevant document).

Model	**P@10**	**MAP**	**BPREF**	**1/Ro1R**
Sent2Vec	0.0250	0.0478	0.0284	0.131
FastTex	0.0175	0.0228	0.0163	0.065
Ensemble	0.0200	0.0181	0.0060	0.044

Table 1. Evaluation results for task 1 systems.

4 Task 2 - Identifying Relevant Statutes

In this task, the goal was to identify relevant statutes given a summarized version of a case as input. To do this, we first extracted key-phrases from the queries and the statutes using the rake-nltk [5] library. For statutes, we further performed some manual augmentation as well as removal of key-phrases based on relevance. Example, for statute S10, "equality of opportunity in matters of public employment", the library didn't select "discriminated" as a keyword so it was manually added while "fifty per cent", which was picked up was removed. Finally, we encoded each of these key-phrases using a pretrained BERT [8] model. Using these encoding vectors, we created three models. For the first model, we computed the cosine similarity scores between each of the key-phrase pairs of every query-statute pairs. Then, for each of the query-statute pair cosine similarity scores, we took the max and second-max values and multiplied them to get the final rank determining score. For the second model, we took a similar approach as the first one, but this time, took an average of the key-phrase cosine similarity scores. In the third model, we used the product of the scores calculated for the first and the second model to get the final relevance score, i.e. the product of the max, second-max and the average score. The performance metrics of all of the models on the testing data is shown in Table 4.

Model	**P@10**	**MAP**	**BPREF**	**1/Ro1R**
M*SM	0.0600	0.0767	0.0309	0.1460
Average	0.0600	0.0918	0.0402	0.2010
Ensemble	0.0600	0.0831	0.0285	0.1620

Table 2. Evaluation results for task 2 systems.

[4] https://trec.nist.gov/pubs/trec15/appendices/CE.MEASURES06.pdf
[5] https://pypi.org/project/rake-nltk/

5 Conclusion & Future Work

We have demonstrated that satisfactory results in both of the tasks can be achieved by taking a simple and fast unsupervised approach using pre-trained embeddings and cosine-similarity scores. Our Sent2Vec and average based system got a rank of 7 and 4 in task 1 and task 2 respectively based on the metric Ro1R. In the future, we would like to collect more legal data and annotate them to build supervised classification models.

Acknowledgement

The authors would like to thank Silversparro Pvt, Ltd. for providing necessary support & computational resources to complete this work. We particularly extend our gratitude to Mr. Ankit Agarwal, CTO and Mr. Ravikant Bhargav, R&D head for their constant support and encouragement.

References

1. Schweighofer, Erich, Andreas Rauber, and Michael Dittenbach. "Automatic text representation, classification and labeling in European law." In Proceedings of the 8th international conference on Artificial intelligence and law, pp. 78-87. ACM, 2001.
2. Gonalves, Teresa, and Paulo Quaresma. "Is linguistic information relevant for the classification of legal texts?." In Proceedings of the 10th international conference on Artificial intelligence and law, pp. 168-176. ACM, 2005.
3. Saravanan, M., Balaraman Ravindran, and S. Raman. "Improving legal document summarization using graphical models." Frontiers in Artificial Intelligence and Applications 152 (2006): 51.
4. Mencia, Eneldo Loza, and Johannes Frnkranz. "Efficient pairwise multilabel classification for large-scale problems in the legal domain." In Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 50-65. Springer, Berlin, Heidelberg, 2008.
5. Conrad, Jack G., Jochen L. Leidner, Frank Schilder, and Ravi Kondadadi. "Query-based opinion summarization for legal blog entries." In Proceedings of the 12th International Conference on Artificial Intelligence and Law, pp. 167-176. ACM, 2009.
6. Saravanan, M., Balaraman Ravindran, and S. Raman. "Automatic identification of rhetorical roles using conditional random fields for legal document summarization." In Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-I. 2008.
7. Bansal, Trapit, David Belanger, and Andrew McCallum. "Ask the gru: Multi-task learning for deep text recommendations." In Proceedings of the 10th ACM Conference on Recommender Systems, pp. 107-114. ACM, 2016.
8. Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).
9. Pagliardini, Matteo, Prakhar Gupta, and Martin Jaggi. "Unsupervised learning of sentence embeddings using compositional n-gram features." arXiv preprint arXiv:1703.02507 (2017).

10. Joulin, Armand, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hrve Jgou, and Tomas Mikolov. "Fasttext. zip: Compressing text classification models." arXiv preprint arXiv:1612.03651 (2016).
11. P. Bhattacharya, K. Ghosh, S. Ghosh, A. Pal, P. Mehta, A. Bhattacharya., P. Majumder, Overview of the Fire 2019 AILA track: Artificial Intelligence for Legal Assistance. In Proc. of FIRE 2019 - Forum for Information Retrieval Evaluation, Kolkata, India, December 12-15, 2019.