Understanding Legal Documents: Classification of Rhetorical Role of Sentences Using Deep Learning and Natural Language Processing

Rameel Ahmad
IBM, Pakistan
syed.rameel.ahmad@ibm.com

Deborah Harris
Elmhurst College, Chicago
harrisdeborahanne@gmail.com

Mohammad Ibrahim Sahibzada
IBM, Pakistan
ibrahim.sahibzada@ibm.com

***Abstract*— Automatically extracting the patterns of reasoning from exhaustive legal documents can make legal systems more effective and increase public access to justice. The vital task to achieve this is to automatically classify sentences in legal documents into categories based on their content. In this paper, we propose a deep learning model that breaks down legal documents and classify the rhetorical types of sentences. We also test out a hypothesis, that using small set of labelled data, we can build deeper and accurate models for processing of legal documents. This will quicken and automate the processing of legal documents hence decreasing, and ultimately eliminating the backlog that currently exists throughout the legal systems. This work can be generalized for legal appeals cases in diverse fields. Breakthroughs in the processing of these documents will decrease appeal timeline. We compare the various configurations used to train our LSTM-RNN model along with a variety of embeddings and show that our results obtained a higher accuracy compared to previous techniques used for semantic understanding of law related documents.**

***Keywords*—Automate; LSTM-RNN; Embeddings; Rhetorical Role.**

I. Introduction

Post-traumatic Stress Disorder (PTSD) is a mental disorder triggered by a traumatic part or singular event in a person's life [1], it has to last for over a month from its offset and it can cause the person to relive the event through recurring flashbacks, disturbed dreams and emotional thoughts of that particular time even though the person wants to block any recollection of what has happened.

The United States provides a variety of benefits for veterans suffering from PTSD, which developed during or as a result of their armed service, determined by the Department of Veteran Affairs' (VA). The benefits include tax-free cash payments [2], free or low-cost mental health treatment and other healthcare [3], vocational rehabilitation services [4], employment assistance [5], and independent living support [6][7].

The number of PTSD claims and appeals by veterans in the United States is rising sharply. The ability to process these legal documents in an automated way is thus crucial to breaking the logjam in the Department of Veteran Affairs', currently backlogged by 400,000 cases [8][9]. In this paper, we propose a neural network approach for breaking down the document and predicting the rhetorical type of sentences for faster understanding of the appeals. Breakthroughs in the processing of these documents will help decrease the 4-year average VA appeal timeline. We cast the problem as a multi-class classification problem.

Using Bidirectional LSTMs model, which learns based on sequential context, the scope of our work is to tokenize a dataset of VA claims and appeals, predicting the rhetorical type of each sentence.

II. Data

For model training we utilized the open-source repository titled Veteran Claims [10]. This repository contains a total of 6,153 sentences, which are part of the analyzed disability-claim decisions issued by the BVA of the U.S. Department of Veterans Affairs. We divide the dataset as 85:15 for training and testing.

This dataset was constructed for public use i.e. for research purposes by the Research Laboratory for Law, Logic and Technology in New York, USA. They extracted from each decision the specific sentences that addressed the fact-based issues which directly correlate to the claim for PTSD or a jointly related psychiatric disorder. Within this dataset there are five key components:

- **docID** - citation number of the decision.
- **sentences** - array contains the subset of classified or annotated sentences from the decision that pertain to the claim for disability benefits due to service-connected PTSD.
- **ruleTree** - representation of the logic of the substantive rules applicable to the case.
- **text** - plain text of the initial, entire decision.
- **metadm** - contains selected metadata about the decision.

In addition to the above, each sentence is further defined into six different Rhetorical Types:

- **Sentence**
- **FindingSentence** - the factual finding.
- **ReasoningSentence** - the premise.
- **EvidenceSentence** - evidence used in the case.
- **LegalRuleSentence** - legal rule without applying it to the case.
- **CitationSentence** - quotes regarding the legal cases and materials.

In the current problem, we will be predicting the Rhetorical type of a given sentence.

III. Previous Works

In 'Automatic Classification of Rhetorical Roles for Sentences: Comparing Rule-Based Scripts with Machine Learning' researchers used an annotated dataset of U.S. decisions, investigated a methodology for qualitatively examining a very limited sub-sample of such decisions [11]. They used the same dataset and tried to classify rhetorical roles using data mining techniques. They compared those outcomes against the performance of standard supervised machine learning algorithms trained on large samples from the same dataset. The algorithms chosen for this study were Naive Bayes, Logistic Regression and support vector machines. While the general accuracy for both the multi-class and two-class experiments appeared to be acceptable, there were substantial deficiencies in this classifier, especially for the important two-class case. The multi-class accuracy was 81.7%. Their results also showed that Logistic Regression was an acceptable classifier for the problem. The multi-class accuracy score of 85.7% was better than that of the Naive Bayes classifier. The performance of the SVM classifier with a linear kernel was similar to that performance of the Logistic Regression classifier. This was true for both the multi-class and the two-class experiments. There was still substantial divergence in the top features chosen by the two algorithms. The features in common were "board find", "thus" and "whether". One hypothesis was that most of the top features were used to decide the Non-Finding class labels, and the Finding class arose as a default class. Several of the highest ranked features seemed to be specific for PTSD cases. Also, as with the Logistic Regression classifier, the confusion matrix for the multi-class SVM did not indicate any dominant source of classification error. The class-wise results for the SVM model, which according to the research, showed the best results are shown in Table I.

TABLE I: Results of SVM Model

Class	**Precision**	**Recall**	**F1-Score**
CitationSentence	98	96	98
EvidenceSentence	88	94	91
FindingSentence	82	78	80
LegalRuleSentence	90	90	90
ReasoningSentence	65	53	58
Sentence	63	63	62

Our approach was inspired by the work in 'C-LSTM Neural Network for Text Classification' [12]. In this paper, a combination of Convolutional Neural Network (CNN) and Long Short-Term Memory Units (LSTM) are used for text classification. C-LSTM was able to capture local features, as well as global and temporal sentence semantics. The proposed architecture was evaluated on sentiment classification and question classification tasks. One-dimensional convolution involves a filter vector sliding over a sequence and detecting features at different positions. Recurrent neural networks (RNNs) are able to propagate historical information via a chain-like neural network architecture. However, standard RNNs become unable to learn long-term dependencies as the gap between two-time steps becomes larger. To address this issue, LSTM was first introduced in [13]. The model achieved the fourth best published result for sentiment analysis of the 5-class classification task on the dataset at the time.

IV. Methodology

In the proposed approach, we used the Bi-directional LSTMs. Bi-directional LSTMs run in two directions, from the past to future and from the future to the past. They are known to show great results by understanding sequential context and semantics better. Compared to traditional LSTMs, Bi-directional LSTMs help improve model performance on sequence classification problems. In problems where all timesteps of the input sequence are available, Bi-directional LSTM train two instead of one LSTMs on the input sequence. Firstly, on the input sequence as-is and then on a reversed copy of the same input sequence. This approach provides further context to the network and results in a faster, fuller learning of the problem. We experimented our model with LSTMs and Bi-directional LSTMs.

Although Bi-LSTMs are deeper than LSTMs, the model was trained in a smaller number of epochs in comparison to LSTMs. The accuracy of Bi-LSTMs model was also 3% higher than the LSTM models making them the preferred choice for our model.

LSTMs when used for Natural Language processing, require word embeddings. Word embeddings are a set of language and feature learning techniques that map words from a vocabulary to a vector space based on the contextual and semantic similarity of the words. Word embeddings are generated using neural networks [14], dimensionality reduction on the word co-occurrence matrix [15][16][17], probabilistic models [18], explainable knowledge base method [19]. Word embeddings, when used as the fundamental input, have been shown to enhance the performance of NLP tasks [20].

Embeddings can be trained along with the model, based on the dataset provided. There are also many pre-trained embeddings available which are trained on very large datasets. We experimented various word embeddings for our model. Following are some of the pre-trained word embeddings that we experimented with to prepare our model:

- **GloVe** is a distributed word representation model for obtaining vector representations for words implementing an unsupervised learning algorithm. It maps words into a vector space where the distance between words is related to semantic similarity [21]. It was developed as an open-source project at Stanford [22]. As log-bilinear regression model for unsupervised learning of word representations, it combines the features of two model families, namely the global matrix factorization and local context window methods [23]. GloVe can discover relations between synonyms, company - product relations, cities, zip codes, etc. It has in addition been used to

detect psychological distress in patient interviews [24].

- **Fastext** is a library developed by Facebook's AI Research Lab (FAIR) [25] for learning of word embeddings and text classification. FastText utilizes a neural network for word embedding. The model creates an unsupervised or supervised learning algorithm for obtaining vector representations for words. Facebook provides pre-trained models for 294 languages.
- **Law2vec** embeddings were created using a considerable number of legal corpora from various public sources in English. The corpus consists of a sum of 123,066 documents 492M individual words [26]. The corpus discards non-UTF8 encoded characters. All words were lower-case, and all numerical digits have been normalized. The embeddings were trained using word2vec models, instead of the most recent fasttext, reason being that word2vec seems to provide better semantic representation than fasttext. The documents used to train embeddings were as follows:
    - 53,000 pieces of UK legislation
    - 62,000 pieces of European legislation
    - 5,500 pieces of Canadian legislation
    - 1,150 pieces of Australian legislation
    - 780 pieces of English-translated Legislation from Japanese
    - 68 bound volumes of the US Supreme Court decisions from 1998 to 2017.
    - 54 titles of the most recently updated U.S. Code.

The final architecture of our network is as follows:

- 300-dimension embeddings layer (GloVe 300d vectors)
- 128 Bidirectional LSTMs
- 256 Fully Connected Layer with RELU activation Function
- 0.5 Dropout
- Fully Connected Layer with Sigmoid activation function

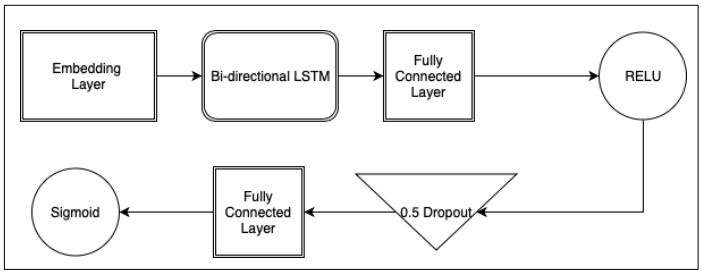

Fig.4 Architecture of Neural Network Used.

V. Results

The results of our model, using the embeddings discussed above are shown in the Table II:

TABLE II Accuracies using different embeddings.

Embedding Type	Accuracy	Precision	Recall	F1-Score
GloVe-300	91	88	86	87
Fasttext	87	83	80	81
Self-learned Embeddings	85	74	78	76
Law2Vec	83	81	73	77

Using the GloVe embeddings with 300-dimension space had the best accuracy of 91%. Experiments with lower dimension of GloVe were also performed however 300-dimension GloVe embeddings gave the best results. GloVe embeddings also performed better than the fasttext embeddings. Law2Vec embeddings gave an accuracy of 83%. However, it had a good precision but very less recall. We also built a model where custom embeddings were trained with a dimension of 300. Each epoch took longer to complete, and the model took very long to train as compared to the pre-trained embeddings. The accuracy was 85%. The lower performance compared to other models must have been due to the small dataset.

Table III show the class-wise scores of our best model trained using the GloVe 300 embeddings.

TABLE III Result of CNN-LSTM

Class	Precision	Recall	F1-Score
CitationSentence	99	96	98
EvidenceSentence	92	96	94
FindingSentence	77	73	75
LegalRuleSentence	88	94	91
ReasoningSentence	71	62	66
Sentence	81	77	79

The Bi-LSTM performs significantly better compared to the SVM model in [12]. It particularly performed very well on the "ReasoningSentence" and "Sentence" classes, where it shows a vast improvement as shown in Table.1. The precision recall and F1-scores for "ReasoningSentence" improved to 71, 62 and 66, respectively. While the precision, recall and F1-scores for "Sentence" class showed vast improvements of several points to 81,77 and 79 respectively. Precision of "FindingSentence" class improved from 77 to 82 and its recall went up to 78.

VI. Other Experiment

We also implemented the approach suggested in CNN-LSTM approach as suggested in 'C-LSTM Neural Network for Text Classification' [12]. CNN was implemented over the embeddings layer and the output of CNN was input to the LSTM layer. The idea was to capture local features and temporal sentence semantics. However, the model took longer to train, and the accuracy was 76%, much less than the Bi-LSTMs. CNN resulted in loss of information of the sequence instead of learning temporal semantics, due to which there was a fall of accuracy compared to Bi-LSTMs directly connected to the embeddings layer.

The complete results are shown in Table IV.

TABLE IV Result of CNN-LSTM

Model	Accuracy	Precision	Recall	F1-Score
CNN-LSTM	76	74	72	73

VII. Conclusion

In this paper, we compared the results of deep learning models for classification of rhetorical type of sentences in legal documents. In previous research [12], similar work was done using rule-based scripts and basic machine learning algorithms. By using LSTMs, we were able to bring significant improvement in the accuracy of classification of rhetorical roles. We also compared various embeddings for this task and got the most efficient results with GloVe 300-d embeddings. This research will facilitate the automation of the VA appeals and can be used for the processing of other legal documents hence, decreasing, and ultimately eliminating the backlog that currently exists throughout the legal systems. The experiments also reiterate the fact that many automation use-cases can be implemented in the field of justice using small set of labelled data.

References

[1] American Psychiatric Association (2013). Diagnostic and Statistical Manual of Mental Disorders (5th ed.).

[2] "VA Compensation Rate Table". Department of Veterans Affairs. Archived from the original on 3 November 2012. Retrieved 20 October 2012.

[3] "Access VA Health Benefits". Department of Veterans Affairs. Archived from the original on 16 October 2012. Retrieved 20 October 2012.

[4] "VA Vocational Rehabilitation". Department of Veterans Affairs. Archived from the original on 19 October 2012. Retrieved 20 October 2012.

[5] "VetSuccess". Department of Veterans Affairs + State Government Veterans Agencies. Archived from the original on 19 October 2012. Retrieved 20 October 2012.

[6] "Independent Living Support for Veterans". Department of Veterans Affairs. Archived from the original on 24 October 2012. Retrieved 20 October 2012.

[7] "Veterans Benefits". Veterans Benefits Administration. Archived from the original on 26 November 2012. Retrieved 30 November 2012.

[8] New VA Secretary Faces 400,000-Case Appeals Backlog, IT Delay by Richard Sisk of Military.com - 31 July 2018

[9] Department of Veterans Affairs (VA) Strategic Plan to Eliminate the Compensation Claims Backlog by U.S Department of Veteran Affairs - 25 January 2013

[10] Veterans Claims Project by Law, Logic and Technology Research Lab, Hempstead, New York, USA.

[11] Automatic Classification of Rhetorical Roles for Sentences : Comparing Rule-Based Scripts with Machine Learning by Vern R. Walker, Krishnan Pillaipakkamnatt, Alexandra M. Davidson, Marysa Linares, Domenick J. Pesce [2019]

[12] A C-LSTM Neural Network for Text Classification Chunting Zhou, Chonglin Sun, Zhiyuan Liu, Francis C.M. Lau.

[13] Long Short Term Memory by Sepp Hochreiter and Jurgen Schmidhuber, 1997.

[14] Mikolov, Tomas ; Sutskever, Ilya ; Chen, Kai ; Corrado, Greg ; Dean, Jeffrey (2013). "Distributed Representations of Words and Phrases and their Compositionality"

[15] Lebret, R√©mi ; Collobert, Ronan (2013). "Word Embeddings through Hellinger PCA". Conference of the European Chapter of the Association for Computational Linguistics (EACL). 2014

[16] Levy, Omer ; Goldberg, Yoav (2014). Neural Word Embedding a Implicit Matrix Factorization

[17] Li, Yitan; Xu, Linli (2015). Word Embedding Revisited : A New Representation Learning and Explicit Matrix Factorization Perspective (PDF). Int'l J. Conf. On Artificial Intelligence

[18] Globerson, Amir (2007). "Euclidean Embedding of Co-occurrence Data" (PDF). Journal of Machine Learning Research.

[19] Qureshi, M. Atif ; Greene, Derek (2018-06-04). "EVE : explainable vector based embedding technique using Wikipedia". Journal of Intelligent Information Systems.

[20] Socher, Richard ; Perelygin, Alex ; Wu, Jean ; Chuang, Jason ; Manning, Chris ; Ng, Andrew ; Potts, Chris (2013). Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank

[21] Abad, Alberto ; Ortega, Alfonso; Teixeira, Ant√≥nio; Mateo, Carmen; Hinarejos, Carlos; Perdig√£o, Fernando; Batista, Fernando; Mamede, Nuno (2016). Advances in Speech and Language Technologies for Iberian Languages: Third International Conference, IberSPEECH 2016, Lisbon, Portugal, November 23-25, 2016,

[22] GloVe: Global Vectors for Word Representation,Jeffrey Pennington, Richard Socher, Christopher D. Manning,Computer Science Department, Stanford University, Stanford, CA 94305

[23] Kalajdziski, Slobodan (2018). ICT Innovations 2018. Engineering and Life Sciences.

[24] Singh, Mayank; Gupta, P. K.; Tyagi, Vipin; Flusser, Jan; √ñren, Tuncer I. (2018). Advances in Computing and Data Sciences : Second International Conference, ICACDS 2018, Dehradun, India, April 20-21, 2018, Revised Selected Papers. Singapore.

[25] Fastext Library by Facebook.Inc

[26] Law2Vec : Legal Word Embeddings by Ilias Chalkidis