Legal Knowledge and Information Systems
A. Wyner and G. Casini (Eds.)

doi:10.3233/978-1-61499-838-9-125

A Semi-Supervised Training Method for Semantic Search of Legal Facts in Canadian Immigration Cases

Isar NEJADGHOLI [a,1], Renaud BOUGUENG [a] and Samuel WITHERSPOON [a]
[a] *IMRSV Research Lab, Miralaw Inc., Ottawa, Canada.*

Abstract.
A semi-supervised approach was introduced to develop a semantic search system, capable of finding legal cases whose fact-asserting sentences are similar to a given query, in a large legal corpus. First, an unsupervised word embedding model learns the meaning of legal words from a large immigration law corpus. Then this knowledge is used to initiate the training of a fact detecting classifier with a small set of annotated legal cases. We achieved 90% accuracy in detecting fact sentences, where only 150 annotated documents were available. The hidden layer of the trained classifier is used to vectorize sentences and calculate cosine similarity between fact-asserting sentences and the given queries. We reached 78% mean average precision score in searching semantically similar sentences.

Keywords. semantic modeling, automatic annotation, semantic similarity search

1. Introduction

Systemic barriers prevent some Canadians from having adequate access to the legal system and a growing number of Canadians represent themselves in court because of the high cost of retaining a lawyer [1].

In this work, we proposed an immigration-specific search algorithm to make legal research more efficient, thorough, and user-friendly. Search engines available in Canada today are not always effective because they merely match keywords to their results and require users to use refining tools to their searches.

In our approach, which is semantic search, the meaning of words and similarity Semantic search will allow users to input natural language queries without the need to be familiar with the jargon used in legal documents and will also respond to queries semantically which includes synonyms and relevant concepts besides exact matches.

Moreover, we designed this system to find sentences that assert a fact of the case and limit the search to only these sentences. The greater the similarity between the facts of any two cases, the more likely the legal outcome, or judgment, will be similar. Thus, older cases can be used to predict new cases. By identifying fact sentences, and com-

[1]Corresponding Author: The Head of Machine Learning Research, IMRSV Research Lab, Miralaw Inc., 100 Sparks, Ottawa, Canada; E-mail: isar@miralaw.ca

paring input queries exclusively to other fact sentences, we believed we could increase the predictive accuracy of our results. Matching fact sentences with different sentence types, such as a sentence which demonstrates reasoning, or a sentence where the litigants state their positions could lead to misleading results. For example, a court may discuss a hypothetical situation to illustrate some auxiliary point. This sentence would have little to do with the facts of the case itself, and would be a poor predictor of legal outcome. Matching these sentences, which are of different types, would be misleading, because, although they are good matches, their ranking would not correlate with the cases' predictive ability, which after all, is the purpose of most legal research.

In this work, we use embedding models to capture not only the semantic meaning of words, but to model the meaning of variable-length word sequences, such as sentences, phrases or combination of keywords. We use a large corpus of immigration law cases to capture the meaning of words in the context of immigration law. This knowledge is used to train a fact-detecting classifier in a supervised manner with a relatively small set of annotated sentences. The resulted model is shown to be able to detect fact-asserting sentences of the whole corpus and the feasibility of semantic search is exhibited.

This paper is organized as following. Section 2 reviews the previous works that are related to our research problem. Section 3 explains the general approach that has been taken in this work to identify fact-giving sentences and search similar sentences to a given query. Section 4 describes the Canadian immigration law corpus that has been used in this work as well as the annotation process. In Section 5, we highlight the details of the models and training methods that have been applied in this work. Section 6 explains our evaluation methods and shows the obtained results. Section 7 summarizes the work and discusses the advantages and limitations of the proposed method.

2. Background

The idea of computer-assisted search for legal cases goes back to 1970s when Lexis legal search and Westlaw were introduced to the public [2]. Traditional legal search is based on finding exact matches to a given combination of keyword queries in a corpus of legal cases. With relentless improvement of software, hardware and natural Language processing techniques, many research efforts have been put to improve the efficiency and accuracy of finding the relevant legal cases and evaluating the results. Extracting different characteristics from a set of legal cases and weighing these characteristics was one of the approaches to improve legal search [3]. With improved calculation capabilities high-dimensional numeric features such as term-frequencies were used to represent legal texts. For example q-grams have been used to calculate similarities between investment treaties [4]. Citation analysis was one of the other advancements that was applied to improve legal information retrieval, by measuring the strength of a case based on how much other cases rely on it [5,6].

Despite all the improvements in keyword search methods, change from the use of keywords to semantics is a recent trend in legal search systems [2,7]. Finding relevant linguistics and semantic patterns has been applied for similarity search among vaccine injury decisions and was a successful step towards semantic search [8]. With the rise of word embedding models as the state-of-the-art semantic representation of words and texts, they have been used by researchers to improve search and navigation of legal data.

As examples, word embeddings have been used for detecting evidence for claims [9], argumentation mining [10] and full-text legal search [11]. Word embedding models are designed to represent a word through its contexts. In this approach, words are described in a dense and low-dimensional vector space in comparison to statistical representations of documents which describe each document as a term-frequency vector. Also, words that appear in the same context will be represented with similar vectors in word embedding models [12]. Moreover, it is impossible to represent Out-Of-Vocabulary (OOV) words in terms of term-frequency vectors [13]. However, character level word embeddings are able to assign vectors to OOV words [14].

3. General Approach

This work investigates the feasibility of semantic search among fact-asserting sentences of the legal cases. One of the challenges of training a "*legal fact detecting classifier*" is the need for adequate training data consisting of a set of legal cases annotated at the sentence level. Such an annotation process can be very costly since it can be done best by expert lawyers. We take a semi-supervised approach and show the feasibility of detecting fact sentences when only 150 annotated cases are available. Although our classifier is trained in a supervised manner, its training is initiated with the word embedding model that is trained on a large corpus in an unsupervised way.

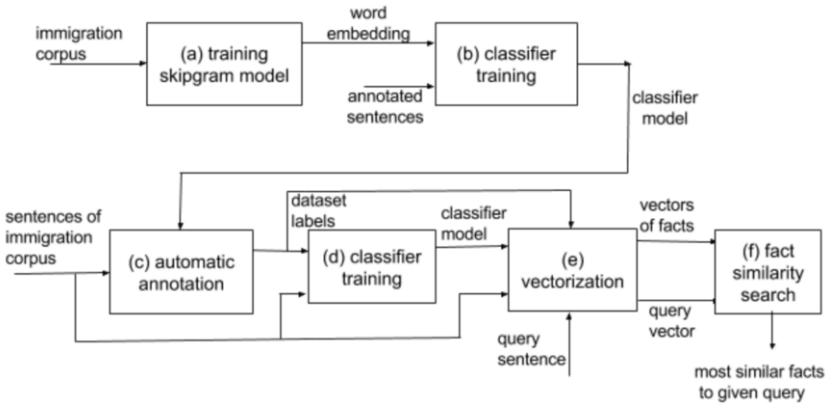

Figure 1. Steps of the proposed semi-supervised method

Figure 1 shows different steps of the proposed semi-supervised method. First, the skip-gram model [15] is trained using a large corpus of immigration cases to capture the meaning of words based on their context in immigration cases (block (a)). A small set of annotated sentences are then used to train a binary classifier that is able to distinguish between facts and non-facts (block (b)). The immigration word embedding is used as initial word representation for classifier training. The trained classifier is then used to automatically label all the sentences of immigration corpus as facts and non-facts (block (c)). This classifier is a shallow neural network with one hidden layer, the values of which can be used as vector representations of sentences in order to calculate sentence similarity. The network is fully connected and the number of neurons of hidden layer equals to the

dimension of vector space which is 100 in this work. The classifier is re-trained with the sentences of automatically annotated immigration corpus in order to capture the meaning of all words of the vocabulary and improve sentence vector representations (block (d)). The re-trained classifier is then used to get the vectors of all the fact sentences of the corpus (block (e)). For a given query, the hidden layer of the trained classifier is used to calculate the vector representation of the query (block (e)). The similarity between the query and the fact sentences of the corpus is then calculated and the most similar facts to the query are returned to the user along with a link to each of the corresponding legal cases (block (f)).

4. Dataset

We use a dataset of 46000 immigration and refugee cases available on Canada's Federal and Supreme Court websites. The HTML documents are first converted to text. Most of the documents contain headers and footers which provides specific information about the case such as date, case name, etc. Documents are processed and headers and footers are removed. The documents are then parsed to sentences. Sentences with less than 20 characters or more than 1000 characters are removed as most of them are a result of wrong sentence spiting and are only 1% of the sentences. We used the Spacy package for sentence splitting and created a set of rules to improve the quality of the sentences splitting considering the specific structure of legal documents such as paragraph numbering, titles, etc. Sentences were tokenized and the punctuations were removed. The cleaned corpus contains more than 136M words, 4549809 sentences and vocabulary size of 125846.

Table 1. Annotation Scheme

Tag	Freq.	Description
Procedure	1%	The nature of the case which is the description of the appeal and the case's procedural history and how the case was treated at previous court levels and/or tribunals.
Fact	46%	The applicant's background information, the applicant's account of his or her story, and the findings made by the tribunal member or previous judge. In the context of immigration this encompasses everything that happened in the administrative tribunal.
Party Position	13%	The applicant and respondent's respective arguments, what they were seeking, and their interpretations of the facts.
Issue	2%	The legal questions the judge must answer/decide upon including the issues ultimately not answered.
Analysis	28%	The judge's decision making process, why and how the judge came to his or her conclusions including any reference to previously decided cases.
Conclusion	6%	The sentences that provide the judge's answer to the issues.
Judgment for Appellant	1%	Statements indicating that the judge decided in favor of the applicant including both orders & holdings.
Judgment for Respondent	1%	Statements indicating that the judge decided in favor of the applicant including both orders & holdings.

4.1. Manual Annotation of Sentences

Two law students manually parsed 150 random cases (each annotator 75 cases) to sentences and annotated the sentences using eight different sentence tags. The detailed de-

scription of the tags as well as their frequency in the 150 annotated cases are provided in Table 1. Sentences that did not fit in any of these tags remained unannotated. The sentences that could viably correspond to more than one potential tag, were tagged according to their most dominant characteristics. From Table 1, we can observe that about half of the sentences are tagged as fact. Therefore, this dataset is balanced for training a binary classifier to detect facts.

5. Training Methods and Procedures

5.1. Unsupervised Semantic Modeling of Legal Words

Distributed word representations or word embeddings are introduced to capture the semantic meaning of words by assigning vectors to each word. Word embeddings have been vastly studied and used in NLP applications [15,16,17,18]. One popular example of building word embeddings is the skip-gram model introduced in [15], where the distributed representations are trained to predict words that appear as their neighbors in the training corpus. The objective function to be maximized during training is:

$$\sum_{t=1}^{M} \sum_{i\neq t, i=t-N}^{t+N} \log p(v_{di}|v_{dt}) \tag{1}$$

summed across all words v_{dt} in all documents v_d, where M is the number of words and N is the length of skip-gram window and p is the probability of occurrence of v_{di} as a neighbor of v_{dt}. We trained a skip-gram model using the dataset described in Section 4 to build a word embedding specialized in immigration law. The dimension of embedding vectors is 100. The embedding features are word n-grams where $n = 1, \ldots, 4$.

Although pretrained word embeddings are available and provided by NLP tools, for semantic search in immigration legal corpus, a word embedding model trained on the same corpus is preferred because it captures the legal meaning of terms. For example, in an immigration law word embedding, the word ***immigration*** is found to be close to word ***FCJ***, which is a frequently occurring Federal Court citation component and ***IRPA***, which is the short form of the *Immigration and Refugee Protection Act*. However, in a general word embedding (provided by Spacy library), ***immigration*** is closest to general terms such as ***reform*** or ***citizenship***. This is because our legal word embedding is trained to convey specific legal meaning of the words in the context of Canadian immigration law whereas the general word embeddings trained on general corpora carries the general meaning of immigration. Another example is the term ***allowed*** which locates near words ***dismissed, costs, ordered, assessment*** in immigration law word embedding and near words ***allow, allowing, not, unless*** in general word embedding provided by Spacy. Among different tools that are available for training a skip-gram embedding space, we chose fastText; developed and implemented by Facebook research team [14]. FastText is very fast in training in comparison to other implementations of skip-gram model and achieves almost the same accuracy. More importantly, it provides vectors for OOV words, since it is trained in character level and uses character n-grams to calculate word vectors.

The quality of a word embedding model is often evaluated in calculating word analogies besides finding most similar words to a query. Table 2 shows some of the interesting

analogies that were produced by trained word embedding model and shows examples of similar pairs of words in the context of Canadian immigration law. Word pairs (w_1, w_2) and (w_3, w_4) are shown in the same row of this table, if $w_1 - w_2 = w_3 - w_4$, where w_i, represents a word vector. [2]

Table 2. Analogues word pairs found from Canadian immigration law word embedding.

Pair1	Pair2
China - Chinese	Sri Lanka - Sri Lankan
Colombian - FARC	Somalian -Alshabab
Roma - Hungarian	Bahai - Iranian
Palestine - Hamas	Lebanon - Hezbollah
PRRA - Preremoval	RPD - posthearing

5.2. Automatic detection of fact-asserting sentences through supervised learning

We used the annotated dataset, described in Section 4.1, to train a binary classifier that automatically detects fact-asserting sentences in immigration cases. We used the supervised model from the fastText library to achieve this classification task. Figure 2 shows the model architecture of fastText supervised classifier for a sentence with N n-gram features, $w_1, \ldots, w_N$ [19] where w_i is the embedding vectors corresponding to i^{th} feature. In this model, the text representation is a hidden variable which can be potentially used as a text representation in other tasks. This architecture is similar to the Continuous Bag of Words (CBOW) model of [15], where the middle word is replaced by a label. The hidden variable is then mapped to the class label through a softmax output layer with the number of neurons equal to the number of classes. The softmax function is used to compute the probability distribution over the predefined classes. For a set of D documents,the following negative likelihood function is minimized.

$$-\frac{1}{D}\sum_{d=1}^{D} y_d \log(f(UWv_d) \tag{2}$$

where v_d is the normalized vector representation of the d^{th} document, y_d represents the label, U and W are the weight matrices. We trained this model on a 4-core CPU machine. The cost function is optimized using stochastic gradient descent and a learning rate that decreases linearly. We used the trained embedding model, described in section 5.1, to initialize the embedding layer of this classifier. In theory, the embedding layer of this model can be initialized either with random vectors or a pre-trained word embedding. we only have 12220 annotated sentences (from 150 cases), which might not be enough to train both embeddings and classifier layer of the structure shown in Figure 2. Therefore, we initialize the first layer with the embedding vectors trained on the immigration law corpus. The trained classifier is used to automatically tag all the sentences of the corpus as facts or non-facts.

[2]In Table 2, PRRA is an acronym for Pre-removal Risk Assessment which is an unsuccessful refugee claimant's last chance to avoid deportation. RPD (Refugee Protection Division) is the body that hears Refugee matters in Canada and posthearing is the descriptor of the consequences of applying.

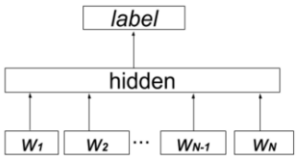

Figure 2. fastText supervised classification model

5.3. Database vectorization and similarity search

Although the classifier explained in Section 5.2 is primarily designed and applied for classification of short documents such as sentences, the values of the hidden layer of the trained model can be used to calculate a vector representation for a given sentence. However, the size of the vocabulary of the annotated documents is much less than the size of the vocabulary of the whole corpus. Therefore only a small portion of word vectors are updated during training of the classifier. To improve the effectiveness of sentence vectorization, we use the automatically annotated sentences of the legal corpus to retrain the classifier. In this way, we update the word vectors of the classifier for all the words of the corpus. After re-training of the classifier, the values of its hidden layer are taken as vector representation of all the sentences present in the corpus. These vectors are stored for similarity search purpose only if the assigned label to the sentence is fact.

For a given query, which can be a sentence, a phrase or a combination of keywords, the re-trained classifier is used to label and vectorize it. We compute the cosine similarity between the query vector and vector of each fact sentence in the corpus. The resulting similarity scores are sorted in a descending order and the three most similar sentences are found and their corresponding documents are returned. The intuition is that sentences which are most similar to the query sentence should rank at the top of the retrieval results. The similarity score can also be understood as an estimate of the relevance of a sentence with respect to the query.

6. Evaluation and results

6.1. Evaluation of the classifier

Although the classifier was trained using sentences parsed by human, we valuated the trained classifier on sentences that were parsed automatically, since that is the ultimate performance of the classifier that the user experiences. 300 automatically parsed sentences were randomly selected from the corpus (annotated documents described in section 4.1 were excluded). These 300 sentences were manually annotated by experts to be used as the test set for classification. In this test dataset, 47% of the sentences are stating a fact and the rest are tagged as non-facts.

In order to compare the classification accuracy of the classifier described in Section 5.2, with benchmark classifiers, we trained six different binary classifiers to detect fact sentences using the training dataset described in Section 4.1 and tested using the test set. The description of theses classifiers as well as the obtained classification accuracies for the test set are given in Table 3. These results show that the proposed semi-supervised method outperforms commonly used classifiers in detecting facts when the amount of training data is relatively small.

Table 3. Classification results for detecting fact sentences using various binary classifiers.

Classifier Description	Acc.
Sentences are represented with *term frequency-inverse document frequency* (tfidf) features. A SVM binary classifier is trained.	81%
Sentences are represented by averaging of word vectors from the embedding space trained in Section 5.1. A SVM binary classifier is trained.	83%
Sentences are represented by tfidf weighted averaging of word vectors from the embedding space trained in Section 5.1. A SVM binary classifier is trained.	84%
A fastText supervised model is trained with random initial word embeddings.	83%
A fastText supervised model is trained with pre-trained word vectors (provided by fasteText) [20] as initial values of embedding layer.	86%
A fastText supervised model is trained with immigration law word vectors (described in Section 5.1) as initial values of embedding layer (proposed semi-supervised method).	90%

6.2. Evaluation of the proposed similarity score

The goal of this evaluation is to measure the algorithm's ability to return semantically relevant sentences as top results, given a query.

We used the Mean Average Precision (MAP) metric, which is a standard comparative evaluation metric for search engines [21] and indicates how precisely the relevant sentences can be ranked on top, in a set of candidate sentences, based on their similarity score to the query.

Table 4. Candidate sentences, corresponding human judgments (HJ) and calculated similarity scores (SS) for the query *"Applicants PRRA rejected despite his fear of persecution and violence in Sri Lanka."*

Sentence	HJ	SS
He claimed to have a well-founded fear of persecution and argued that Sri Lanka was violent, nevertheless, the PRRA rejected the application.	R	0.96
The PRRA Officer reviewed the Applicants immigration history and quoted extensively from his statutory declaration dated March 26, 2014 including the Applicants claim that, even though the war in Sri Lanka has ended, the situation there is worsening in many ways and that, if returned, he would face discrimination and harassment due to his ethnicity and would be targeted because he has family overseas.	R	0.93
The applicant is a member of Tamil and claims fear of persecution.	R	0.85
With respect to the male Applicant's claim, the Board held that he did not have a well-founded fear of persecution because he was not really wanted by the Iranian authorities.	N	0.74
Specifically, the Applicant argues the RPD erred by: making plausibility findings without specific reference to the evidence to support such findings; making an overall credibility finding before independently assessing his corroborative evidence; discounting the psychiatrists report; and, failing to address his claim that he was kidnapped by authorities in 2013.	N	0.12

We designed an evaluation dataset with human judgments on semantic similarity. The evaluation dataset is a collection of 15 queries, crafted by legal experts, each targeting one important area within Canadian Immigration Law. For each query, a set of 5 candidate sentences was built which was a mix of sentences handpicked from the CanLii website [22] and sentences handcrafted by the evaluators. The evaluators assessed the relevance of a sentence with respect to its query by marking it as "Relevant" or "Not Relevant". The candidate sentences were ranked based on their similarity score to the query and Average Precision (AP) was calculated for each query to measure how precisely the

"Relevant" candidates are ranked higher than "Not Relevant" candidates. As an example, Table 4 shows the candidate sentences, human assessments and similarity scores for the query *"Applicants PRRA rejected despite his fear of persecution and violence in Sri Lanka."*. We simply calculated the mean of all APs over all queries and obtained the MAP score of 78%.

7. Discussion, limitations and scope of use

We showed the feasibility of detecting fact-asserting sentences and searching for semantically similar facts in a large Canadian immigration law corpus when only 0.3% of the corpus is manually annotated. Figure 3 shows a screenshot of a system developed based on the proposed method.

Our evaluation show that a supervised fastText classifier that is initiated with immigration law word embedding is more effective than benchmark classifiers in identifying fact sentences (see Table 3). Evaluation of the proposed semantic similarity score has been carried out using a small hand-crafted evaluation dataset and an acceptable MAP score is acquired. The main advantage of the proposed similarity score is that it automatically limits the search to facts given a fact query, since the similarity score between a fact sentence and a non-fact sentence calculated by the proposed method is very low even if the two sentences share semantically similar words. The other advantage of this method is that slight misspelling of words does not change the results, since fastText is a character level word embedding. An alternative method of semantic score calculation such as tfidf weighting of word vectors will completely ignore misspelled words, since they are not included in the vocabulary of tfidf calculator. A more rigorous quantitative evaluation of this search method and comparing it with other alternatives remains as a focus of future work due to challenges of designing a comprehensive evaluation dataset.

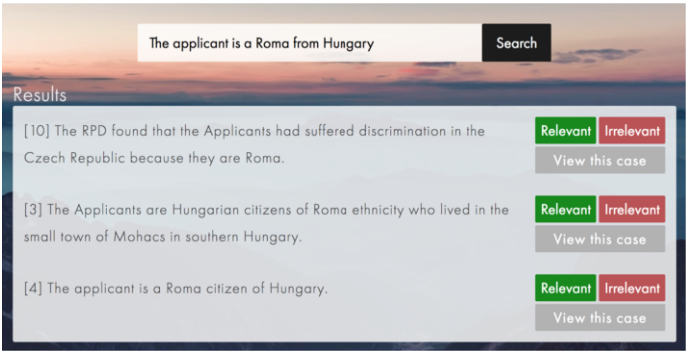

Figure 3. Example of the results returned by the developed system.

Acknowledgement

The authors would like to thank Ms. Chelsea Kirsch and Mr. Micheal Elharrar for annotation, their enormous help in formulating the research question in the context of Canadian legal environment and also evaluating the results. This research was made possible in part with funding from the Canadian Industrial Research Assistance Program (IRAP).

References

[1] R. Birnbaum, N. Bala, and L. Bertrand, The rise of self-representation in canadas family courts: The complex picture revealed in surveys of judges, lawyers and litigants, Canadian Bar Review **91**, 2013.

[2] J. O. McGinnis and R. G. Pearce, The great disruption: How machine intelligence will transform the role of lawyers in the delivery of legal services, Fordham L. Rev. **82**, (2014), 3041-3050.

[3] Q. Lu and J. G. Conrad, Next generation legal search-its already here, Vox Populii blog, Legal Information Institute (LII), Cornell University, 2013.

[4] W. Alschner and D. Skougarevskiy, Consistency and legal innovation in the bit universe, Stanford Public Law Working Paper No. 2595288, 2015.

[5] J. H. Fowler, T. R. Johnson, J. F. Spriggs, S. Jeon, and P. J. Wahlbeck, Network analysis and the law: Measuring the legal importance of precedents at the us supreme court, Political Analysis **15** (2007), 324346.

[6] R. Winkels, A. Boer, B. Vredebregt, and A. van SOMEREN, Towards a legal recommender system, JURIX **271** (2014), 169178.

[7] E. Francesconi, S. Montemagni,W. Peters, and D. Tiscornia, Semantic processing of legal texts: Where the language of law meets the law of language, Lecture Notes in Computer Science **6036**, Springer, 2010.

[8] M. Grabmair, K. D. Ashley, R. Chen, P. Sureshkumar, C. Wang, E. Nyberg, and V. R. Walker, Introducing LUIMA: an experiment in legal conceptual retrieval of vaccine injury decisions using a UIMA type system and tools, ACM (2015), 6978.

[9] R. Rinott, L. Dankin, C. A. Perez, M. M. Khapra, E. Aharoni, and N. Slonim, Show me your evidence: an automatic method for context dependent evidence detection, EMNLP (2015), 440450.

[10] N. Naderi and G. Hirst, Argumentation mining in parliamentary discourse, International Workshop on Empathic Computing (2014), 1625.

[11] J. Landthaler, B.Waltl, P. Holl, and F. Matthes, Extending full text search for legal document collections using word embeddings. JURIX (2016), 7382.

[12] K. Erk, Vector space models of word meaning and phrase meaning: A survey, Language and Linguistics Compass **6** (2012), 635653.

[13] F. Huang and A. Yates, Distributional representations for handling sparsity in supervised sequence labeling, ACL-AFNLP **1**, Association for Computational Linguistics (2009), 495503.

[14] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, Enriching word vectors with subword information, Transactions of the Association for Computational Linguistics **5**, (2017), 135146.

[15] T. Mikolov, K. Chen, G. Corrado, and J. Dean, Efficient estimation of word representations in vector space, arXiv preprint arXiv:1301.3781, 2013.

[16] D. E. Rumelhart, G. E. Hinton, R. J. Williams, et al., Learning representations by back-propagating errors, Cognitive modeling **5**, 1988.

[17] F. Morin and Y. Bengio, Hierarchical probabilistic neural network language model, AIStats **5** (2005), 246252.

[18] J. Pennington, R. Socher, and C. D. Manning, Glove: Global vectors for word representation., EMNLP **14** (2014), 15321543.

[19] A. Joulin, E. Grave, and P. B. T. Mikolov, Bag of tricks for efficient text classification, EACL (2017), 427-430.

[20] Pretrained fasttext word vectors. *https://github.com/facebookresearch/fastText/blob/master/pretrainedvectors.md*, Accessed at 24-08-2017.

[21] A. Turpin and F. Scholer, User performance versus precision measures for simple search tasks, ACM (2006), 1118.

[22] Canlii website, *https://canlii.org*, Accessed: 2017-06-30.