Project Report on

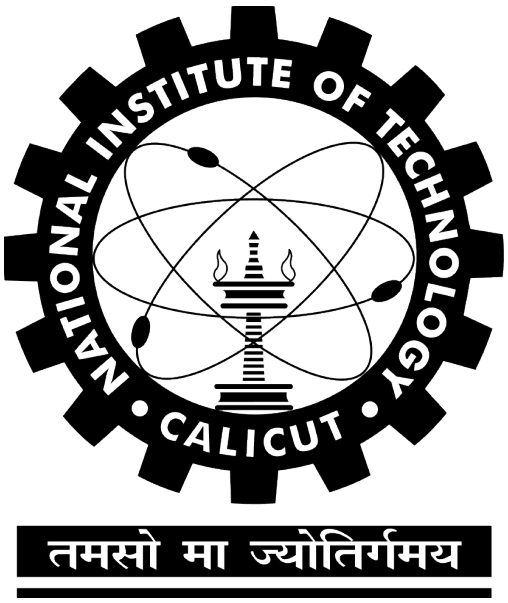
Biomedical Text Mining for Drug Repurposing Using a Word Embedding Implementation With SVM and Deep Neural Networks

Submitted by

Sruthi P. G. **B150254CS**
Shravani Sridhar **B150062CS**
Aditya Kumar **B150476CS**
Ahmed Bilal **B150363CS**
R. Swathy **B140394CS**

Under the Guidance of

Dr. K A Abdul Nazeer



Department of Computer Science and Engineering
National Institute of Technology Calicut
Calicut, Kerala, India - 673 601

March 28th, 2019

Biomedical Text Mining for Drug Repurposing Using a Word Embedding Implementation With SVM and Deep Neural Networks

Sruthi P. G. Shravani Sridhar Aditya Kumar Ahmed Bilal R. Swathy

Abstract: Drug repurposing is the process of applying an existing drug to treat a different disease than the one it was originally used for. It is becoming a common replacement to drug discovery as it has been found to be far more efficient, less costly, less time-consuming, and less risky than drug discovery. One major challenge in drug repurposing is finding new drug-disease relationships. One approach for this is text mining, which is increasingly being used to identify and extract relationships between biological entities in literature. Most strategies that are used represent such units as vectors, but a disadvantage of this is that the vectors generated have too large a number of dimensions and are highly sparse. To counter this, recently a new approach has been tested, namely, word embedding, which generates relatively short numerical vectors as representations of word sense. Thus, word embedding proved to be an efficient encoding system that allows the corpus to be processed in moderate computational space and time and that also generates word vectors that are reasonably semantically equivalent to the words themselves. The final classification model, which was based on SVM (Support Vector Machine), attained a good accuracy and was successfully able to discover new drug-disease relationships; concatenating the vectors of drugs and diseases in the discovered relationships could then be used to identify candidate drugs for repurposing. In this project, we aim to utilise word embedding, and a deep neural network for the prediction of novel drug-disease relationships from a large amount of cancer-related biomedical literature, and use the relationships to identify possible drugs that could be repurposed. We will also use an SVM approach for classification, then compare its results with that of the DNN approach. [5]

1 Problem Definition

We aim to repurpose existing and approved drugs using biomedical text mining of a large cancer-related corpus, specifically using a word embedding implementation to generate semantically equivalent representations of words in the text and then a deep neural network for predicting drug-disease relationships from the representations. We will also use an SVM for the task of prediction and then compare the results with that from using a DNN.

2 Literature Survey

2.1 Case 1: Computational Method for Drug Repurposing

Work Done A data mining process using publicly available gene expression datasets associated with a few diseases and drugs was carried out to identify existing drugs that could be used to treat genes causing lung cancer and breast cancer.

Results Three strong candidates for repurposing were identified: Letrozole and GDC-0941 against

lung cancer, and Ribavirin against breast cancer. Letrozole and GDC-0941 are drugs currently used in breast cancer treatment and Ribavirin is used in the treatment of Hepatitis C. [1]

How our project will be different A more recent, alternative approach to drug repurposing is text mining. We will use biomedical text mining, specifically.

2.2 Case 2: Network-based Approach for Drug Repurposing

Work Done A network-based approach for drug repurposing was utilised that takes into account the human interactome network, proximity measures between drug targets and disease-associated genes, potential side-effects, genome-wide gene expression, and disease modules that emerge through pertinent analysis.

Results Network-based drug-disease proximity was found to offer a novel perspective to a drug's therapeutic effect. In addition, the network-based approach was found to provide a fast and efficient way to determine likely candidates for drug repurposing and understand their underlying mechanisms, with far-reaching applications to various diseases beyond Rheumatoid Arthritis (RA). [2]

How our project will be different Refer the same section for Case 1.

2.3 Case 3: Biomedical Text Mining Using a Full Parser

2.3.1 Full Parser

In the task of text mining, parsing of sentences is done before generating computational representations of linguistic units in text. One method of parsing is using a full parser. A full parser is domain independent and scrutinizes the complete structure of the given data. It converts the basic structure of the given data into an argument structure for further analysis.

Work Done A system which uses a full parser for analyzing biomedical text was developed. A preprocessor was used to partially overcome the shortcomings of full parsing.

Results The developed system could be maintained easily and could adapt itself for a particular domain. In the primary experiment, out of 131 argument structures extracted from 96 sentences, 32 were extractable without ambiguity, 33 with ambiguity, and 66 (non-extractable) for which the partial result was determined. A total of 99 argument structures were predicted for extraction, with the presence of a post-processor and a disambiguation module. The full parsing technique was found to be viable to the application of IR (Information Retrieval) systems. [3]

Limitations A full parser requires an oversized memory size, and it is slower in execution.

How we aim to overcome the limitations We will use POS tagging on the sentences for parsing them, as it is much faster than full parsing.

2.4 Case 4: Text Mining using Bag-of-Words (BoW) Model for Prediction

2.4.1 Bag of Words (BoW) Model

It is a technique for computationally representing linguistic units in text like sentences, words, and documents. Each document is represented as a vector of word frequencies in it. For word representation, only the neighboring words in the same sentence are counted. Stop-words can be removed and raw frequencies can be modified by term weighting for better analysis. The constructed vectors are then used to evaluate the characteristics of the units and similarities between them.

Work Done An easy-to-use framework was developed for accelerated usage of the BoW model in text mining and processing.

Results The developed framework was generic and was shown to be able to be easily replicated across many other related domains as well, highlighting its utility.

Limitations Due to the large dimensionality and high sparseness of the vectors that the BoW model tends to generate, other approaches such as word embedding have been proposed. Traditional algorithms for dimension reduction or compression, like Principal Component Analysis (PCA) and Latent Semantic Analysis (LSA), have been used to attempt to solve the large dimensionality problem, but unsuccessfully.

How we aim to overcome the limitations 1. We use word embedding for computationally representing words. The word embedding technique is based on a neural network algorithm and generates relatively short numerical vectors as representations of word sense, thus avoiding large dimensions.
2. In BoW model, the frequency of a word follows Zipf's law, which states that given a large sample of words, the frequency of any word is inversely proportional to its rank in the frequency table. Due to this law, most of the millions of words in a text will only occur a few times, making the frequency vectors by BoW model sparse. Word embedding does not make use of frequencies, thus reducing the risk of generating highly sparse vectors.
3. BoW model does not look at the context of words when constructing their representations. Word embedding does, leading to more accurate and semantically equivalent representations of words. [4]

2.5 Case 5: Comparing Machine Learning and Deep Learning Approaches for Drug Repurposing

Work Done Various machine learning approaches were applied for prediction in the task of identifying repurposing opportunities for treating schizophrenia and depression/anxiety disorders. The approaches tried consisted of SVM and Deep Neural Networks (DNNs), among others, and they were then compared based on their performances.

Results The performance of the five approaches did not differ substantially; though SVM slightly outperformed the others. The relatively modest sample size of the dataset used was suspected to have been the reason that DNN was limited against achieving the optimal predictive ability, as deep learning techniques tend to work best (and better than machine learning approaches) with large datasets. However, using larger samples may lead to greater computational costs. The study overall showed that deep learning can achieve reasonable performance in drug repurposing (DNN had achieved the best ROC-AUC for depression/anxiety disorders in the weighted analysis.).

Limitations The dataset used in the study had been relatively small, most likely leading DNN to achieve a lower performance than SVM. But given the rapid growth in the area, deep learning approaches are definitely worthy of further investigations.

How we aim to overcome the limitations We plan to use a larger dataset to test the effectiveness of our deep neural network in predicting drug-disease relationships from the word sense representations. We will also use an SVM approach for the same. Then we will compare the performances of both and see if the former's is better.

3 Project Design

The following will be done/included in our project:

1. **word2vec** algorithm for word embedding
2. POS tagging for sentence parsing
3. Comparison of the performances of Continuous Bag-of-Words (CBOW) and skip-gram training algorithms in **word2vec**
4. Comparison of the performances of DNN and SVM in the prediction of drug-disease relationships using our large dataset
5. Python 3.x as our programming language
6. TensorFlow library for numerical computation that makes deep learning easier

3.1 Dataset

We will use a set of cancer-related reports downloaded from PubMed as our raw corpus.

3.2 Structure

Our project will be divided into three modules as follows (refer Figure 1):

3.2.1 Information Extraction

First, the sentences from our biomedical text corpus will be extracted and parsed as part of Part-of-Speech (POS) tagging. To simplify input to our word embedding algorithm, all words except for nouns, adjectives, adverbs, and verbs will be removed from the sentences; the remaining words will then be converted into their base forms. Next, we will perform Named Entity Recognition on the sentences.

3.2.2 Word Embedding

Word embedding is a technique based on neural networks which outputs relatively short numerical vectors for all words in a text. It has been proven that, unlike previous techniques like BoW model, the created vector space by word embedding represents word senses and distances, i.e, similarities, between the word senses reasonably well. We plan to use the **word2vec** algorithm, which is a de facto standard word embedding algorithm, for generating the word embedding from the text. Generally, **word2vec** can use either of two training algorithms to learn the embedding: Continuous Bag-of-Words (CBOW) and skip-gram. We will use both and compare the results.

After generating the word vector representations, we will bind each of the vectors with an appropriate word class. The knowledge of which classes to use for the vectors can be obtained from drug and disease datasets. Then we will filter the bindings to allow only those word vectors of drugs and diseases and that are related to cancer.

3.2.3 Classification and Prediction

A model (here, SVM or DNN) will be used on the vector-class bindings and, making use of a set of correct drug-disease relations from an appropriate dataset, predict new drug-disease relations. Concatenating the vectors of drugs and diseases in the discovered relationships will then be used to identify candidate drugs for repurposing.

3.3 Method

Our corpus will be divided into two sets: a testing set consisting of roughly one-thirds of the original data and a training set consisting of the remaining data.

3.3.1 Training

After executing the first two modules on the training set, we will train our models separately on the output of the second module and obtain the results.

3.3.2 Testing

We will do the same as above for the testing set and obtain the results.

We will compare the results of the performances of using SVM and DNN and conclude.

4 Pseudocode

The following is the general algorithm for classifying data using SVM:
1. Import the dataset
2. Encode the target features as vectors
3. Split the dataset into training and testing
4. Perform feature scaling
5. Fit SVM to the training set
6. Predict the test set results
7. Visualize the training set results
8. Visualize the test set results

We will use the **word2vec** software for word embedding implementation.

5 Project Implementation

5.1 Information Retrieval

In the initial phase of Information Retrieval, the biomedical, cancer-relevant research papers were extracted sentence-wise and subjected to Part-Of-Speech (POS) tagging. An off-the-shelf POS tagger using the Penn Treebank tagset was used for this purpose. Tagged tokens, i.e., tokens associated with tags that represent their part of speech, were encoded as tuples in the form of (tag, token). The tags for nouns, adjectives, adverbs, and verbs begin with 'NN', 'JJ', 'RB', and 'VB' respectively. This property was exploited to remove all the stop-words and to ensure that the tokens remaining were just nouns, adverbs, adjectives, and verbs.

To convert the tokens to their base forms, we used one of the most popular stemming algorithms, **Porter Stemmer**, which comes as a part of the **nltk.stem** package using its **PorterStemmer** class. Stemming is the process of decreasing inflections or variations in words to their root forms, such as mapping a group of words to the same stem even if the stem itself is not a valid word in the language. For Named Entity Recognition, an effort was made to use **spaCy**, an open-source software library for advanced natural language processing, written in the programming languages Python and Cython.

5.2 Word Embedding

The phase of Word Embedding maps words to relatively short vectors of real numbers. We used the **word2vec** algorithm, which is for generating the word embedding of all extracted tokens resulting from the first phase. The Word2Vec models are shallow two-layer neural networks having one input layer, one hidden layer, and one output layer for generating word embedding. The two architectures used by Word2Vec, Continuous Bag-of-Words (CBOW) model and skip-gram, were used separately to perform word embedding. Another open source Python library for natural language processing, **Gensim**, was used as well to provide the **Word2Vec** class for working with a Word2Vec model.

Once the word embedding for the tokens were generated, classical projection methods of **scikit-learn** i.e., PCA (Principal Component Analysis), were used to reduce the high-dimensional word vectors to two-dimensional scatter plots using **matplotlib** and were then visualized on a graph. The resulting projection was plotted using **matplotlib** as follows: the two dimensions were pulled out as x and y coordinates. Thus a 2-dimensional PCA model of the word vectors using the scikit-learn PCA class was created on which classification and prediction were done.

For each of the generated vector representations of the tokens, an appropriate word class (drug or disease) was identified and bound with it. For this vector-class binding, we make use of drug and disease datasets. The drug dataset has 200 drugs, and the disease dataset has 50 diseases in total.

5.3 Classification and Prediction

Using SVM and DNN models and exploiting a dataset of correct drug-disease relationships, we predicted drug-disease relations from the vector-class bindings. The testing set consisted of roughly one-thirds of the original data, and the training set consisted of the remaining data.

6 Results

6.1 Performance Comparison Between DNN and SVM

Overall, the Deep Neural Network gave better performance in predicting drug-disease relationships from the word vector representations than the SVM did. The DNN had a higher classification accuracy than SVM.

In general, deep learning works better than machine learning for larger datasets, and we had used a relatively large dataset of cancer-related reports as our corpus to test this. Thus, we can say that our hypothesis had been proven true.

6.2 Performance Comparison Between Continuous Bag-of-Words (CBOW) and Skip-Gram Training Algorithms in word2vec

With a smaller dataset, both training algorithms gave equivalent outputs, i.e, equivalent word vector representations of input words. As we increased the size of our dataset, the discrepancy between their outputs increased. We observed the following using our final corpus: skip-gram was better able to understand words that were infrequent (with respect to the corpus) than CBOW was; hence the accuracy of the former was higher. This result can be better understood after we discuss the principles behind both training algorithms. There are some important differences between them, though both algorithms generally look at a window of words for each target word to provide context and hence word meanings.

6.2.1 Continuous Bag-of-Words

This algorithms learns to predict a word given a context, or find the word with the maximum probability given a context. This automatically becomes an issue for infrequent words, since they dont appear very often in a given context. As a result, the model will assign these words low probabilities and thus make them harder to predict. [8]

6.2.2 Skip-Gram

The skip-gram algorithm learns to predict the context given a word. Thus, two words (one infrequent and the other frequent) are treated the same. Hence, the model will learn to understand even rare words. [8]

7 Conclusion

In this project, our aim was to demonstrate by implementation the effectiveness of different word embedding training algorithms namely CBOW model and Skip-Gram for representing senses of all words in a large amount of cancer-related biomedical literature and thus also in the discovery of novel drug-disease relationships from it for drug repurposing. It was also concluded that out of Support Vector Machine and Deep Neural Network models used for prediction of new drug-disease relationships, Deep Neural Networks provided higher classification accuracy. With regards to the different word embedding training algorithms we used,the accuracy of the skip-gram was higher compared to CBOW for larger dataset.

In the future, our project can be extended and improved using various feature selection or data dimension reduction techniques. In this manner, the few most important variables or parameters which help in predicting the outcome are identified, which aids in giving better results in predictive analytics. Oversampling technqiues, like SMOTE and ADASYN, which are used to adjust the class distribution of a given dataset, can also be employed as an improvement to deal with data imbalance by introducing a bias.

References

[1] K. M. Shabana, K. A. A. Nazeer, M. Pradhan, and M. Palakal. A computational method for drug repositioning using publicly available gene expression data. *BMC Bioinformatics*, 2015, vol. 16, https://doi.org/10.1186/1471-2105-16-S17-S5.

[2] K. Kim, N. Rai, M. Kim, and I. Tagkopoulos. A network-based model for drug repurposing in Rheumatoid Arthritis. University of California, California, US, 2018, https://doi.org/10.1101/335679.

[3] P. Govindarajan, K. S. Ravichandran. Text mining from biomedical domain using a full parser. *2016 International Conference on Inventive Computation Technologies*, 2016, vol. 3, https://doi.org/10.1109/INVENTIVE.2016.78248

[4] D. S, P. Raj, and S. Rajaraajeswari. A Framework for Text Analytics using the Bag of Words (BoW) Model for Prediction. *International Journal of Advanced Networking Applications*, Raja Rajeswari College of Engineering, Bangalore, India. https://doi.org/10.1109/BIBM.2015.7359756.

[5] D. L. Ngo, N. Yamamoto, V. A. Tran, N. G. Nguyen, D. Phan, F. R. Lumbanraja, M. Kubo, K. Satou. Application of Word Embedding to Drug Repositioning. *J Biomedical Science and Engineering*, 2016, vol. 9, pp. 7–16, https://doi.org/10.4236/jbise.2016.91002.

[6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa. Electron spectroscopy studies on magneto-optical media and plastic substrate interface. *IEEE Transl. J. Magn. Japan*, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[7] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[8] "In Word2vec, why is the skip-gram model preferred over CBOW?" quora.com. https://www.quora.com/In-Word2vec-why-is-the-skip-gram-model-preferred-over-CBOW (accessed Mar. 28, 2019).