

# Word Representation for Biomedical Textmining based on NNGM with Image to Text Conversion

Jinsha V. S.<sup>1</sup>, Limna Das .P<sup>2</sup>

<sup>1</sup>Calicut University, M.Dasan Institute of Technology, Nanminda, Calicut -673613, Kerala, India

<sup>2</sup>Calicut University, M.Dasan Institute of Technology, Calicut, Kerala, India

**Abstract:** For capturing semantic regularities some of the distributed word representations are succeeded, but most of them are shallow window based model. To represent deeper information propose NNGM with image to text by using the concept of OCR (Optical character recognition). OCR is the mechanical or electronic conversion of images in to text format. This method considers dependency relation and context relation and clearly expresses the semantic regularity to emerge in word relation and represent the words using deeper information. In our method, performance measured on Protein-Protein Interaction Extraction and Word analogy task. Compare to other method, our word representation method perform better.

**Keywords:** Natural Language Processing, Machine learning, Connectionism and neural nets, Object representation

## 1. Introduction

Words in the text should be represented into real values or real valued vectors so that the power of mathematics can be used to solve Biomedical Natural Language Processing (Bio-NLP) problems. The most commonly used representation method for categorical features is the one-hot coding, by which each word is represented as a vector with only one 1 and many 0s, e.g., suppose a dataset having only a single categorical feature "type", with values "hormone", "kinase" and "receptor", the corresponding one-hot vectors are [1, 0, 0], [0, 1, 0] and [0, 0, 1] respectively.

The experimental results show that the one-hot coding works well with learning algorithms such as Logistic Regression and Support Vector Machine. However, the pair wise Euclidean distances between them are all equal to 2, and which lacks the capacity to capture the semantic regularities of words. Worse still, one-hot coding is a disaster for Euclidean distance based algorithms such as K-means. so more powerful distributed representation models are motivated

For most machine learning based Bio NLP tasks, such as Name Entity Recognition, Protein-Protein Interaction Extraction, Drug-Drug Interaction Extraction, Event Extraction, Ontology Curation, the word representation methods are most important. It has been proved that reasonable distributed word representations can help improving the performance of those Bio NLP tasks. To represent words by using deeper information introduce the NNGM with image uploading by using the concept of OCR(optical character recognition).compared to other method it has high performance.

## 2. Problem Definition

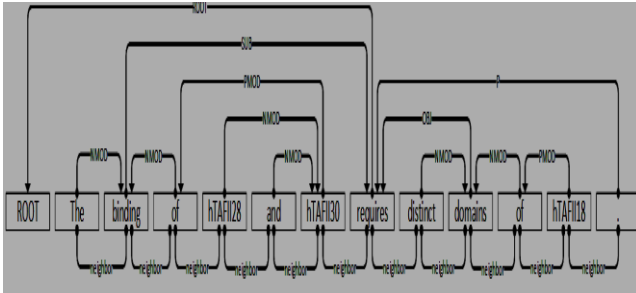
The main problem with existing approach is they only consider the statistical information between words (terms) and documents, where richer information such as context

words, dependency links are not considered. Besides, on large training corpus, the size of vectors may be too large for the memory to load, and in this case, dimension reduction strategies should be applied. The available methods are relatively poorly on the word analogy task i.e., they are not good enough for measuring the meaning of words. Due to these reasons, in this work, try to improve the word representation method based on neural network leveraging richer information, to make the word vectors more powerful to express the semantics. The new neural network-based word representation model overcoming three shortages stated above: 1) it leverages dependency and context relations rather than only considering context window, and 2) it is task irrelevant rather than joint training, which can be integrated into any NLP task and 3) it is trained with high efficiency, for example, the word representations for PPIE can be trained within 15 minutes.

## 3. Proposed system

### 3.1 Proposed Outline

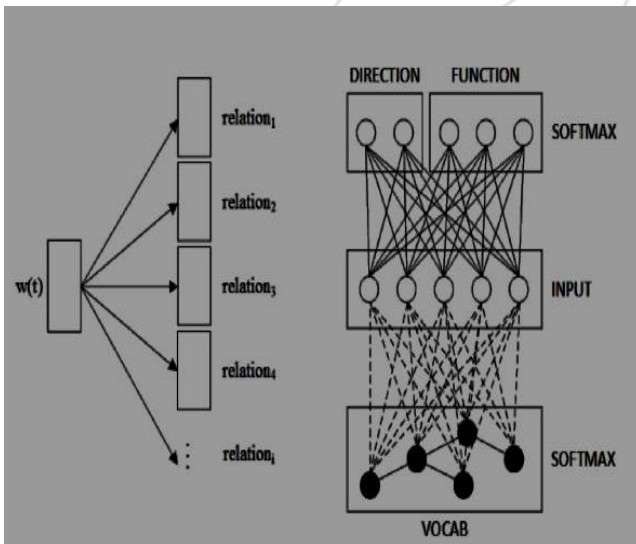
An "isolated word" that would not have any practical meaning and if it had no relations with any other words, think the essence of semantic meaning lies in the relation are the key for computers to understand the meaning of words is relation. For example, the relation between "binding" and "requires" as shown in Fig. 1, means not only that there is a connection between "binding" and "requires", but also that "binding" and "requires" confirm to subject-predicate structure. Without the relation type "SUB" and "OBJ", the algorithms cannot distinguish "binding", "domains" when training "requires". Focus on how to explain what and how other words are linked with it to capture the semantic meaning of a word. Formalized the semantic information of a word as what and how other words are related with it. Previous methods such as Neural network graph model and Continuous bag of words trained a word by context words, but we train each target word by relations according to graph.



**Figure 1:** An example of relation graph. The upper part is dependency relations, and the lower part is context relations.

### 3.2 Neural Network Graph Model

The Fig. 2 shows the neural network based architecture. Establish some notation first. The matrix of word representation is  $X$ , which is randomly initialized and  $X_t$  is the corresponding vector of the  $t$ th word. Let all links in the corpus be denoted by  $L$ . Let  $P_{j,t} = P(y=L_j/X_t)$  be the probability that the  $j$ th link appears in the context of the  $t$ th word. and finally let the  $j$ th link  $L_j$  from a target word  $t$  to the  $w$ th be denoted by  $L_{tw}$  which can be formalized as a tuple (vocab <sub>$w$</sub> , function <sub>$t,w$</sub> , direction <sub>$t,w$</sub> ), or  $(V_w, F_{t,w}, D_{t,w})$ .



**Figure 2:** The two-layer Neural Network Graph Model and the detailed learning architecture. Three-top-layer.

A simple example that shows cases how certain aspects of meaning can be extracted directly from links. Consider a biomedical term  $V_t$ , suppose we are interested in whether it is a protein or not, for which we might take  $V_t = \text{cofilin}$ . The meaning of cofilin can be recognized by studying the probabilities with its relations. For link  $L_j$  related to cofilin,  $L_j = (\text{cofilin}, \text{phosphorylate}, \text{OBJ}, \square)$ . The conditional probability  $P(y=L_j / X_t)$  will be large, because phosphorylation of proteins is an important regulatory mechanism, and large conditional probability of  $L_j$  given  $X_t$  proves that  $V_t$  is a protein that can be phosphorylated.

$$p_{jt} = p_{j,t}^V + p_{j,t}^F + p_{j,t}^D \quad (1)$$

The above argument suggests that the appropriate starting point for word vector learning should be with the conditional probabilities. The links implicate the words meaning by,  $V_w$  (the word  $V_w$  that  $V_w$  links to),  $F_{tw}$  (the grammatical function

of the link) and  $D_{tw}$  (the direction of the link). The target word vector  $X_t$ , we want our hypothesis to estimate the probability  $P_{j,t}$  for each link by the three parts. Thus, our hypothesis will output a  $|V|+|F|+|D|$  dimensional vector whose elements sum to 3. Concretely, the conditional distribution takes the form where  $\lambda_V, \lambda_F, \lambda_D$  are the weight matrices of softmax layer for vocab, function and direction respectively, and the transpose of the input matrix  $X$  is  $X^T$ . Parameters  $\lambda_V, \lambda_F, \lambda_D$  and word vector  $X_t$  are trained to minimize the cost function

$$J = \sum_{C \in \{VFD\}} \sum_{K \in C} 1\{Y_C = C_K\} \log p_{j,t}^D \quad (2)$$

The indicator function is  $1\{\cdot\}$ ,  $1\{\text{a true statement}\}=1$ , and  $1\{\text{a false statement}\}=0$ . To solve for the minimum of  $J$  and train the word vectors  $X_t$ , we resort to gradient descent, the widely used iterative optimization algorithm.

$$\lambda_{LKC} \leftarrow \lambda_{LKC} - \alpha \nabla \lambda_{LKC} J \quad (3)$$

$$X_t \leftarrow X_t - \alpha \nabla X_t J \quad (4)$$

Where  $\alpha$  is the learning rate. Taking derivatives, the gradients are

$$\nabla \lambda_{LKC} = -X_t (1 - y_C = C_k - P_k, tC) \quad (5)$$

$$\nabla X_t J = - \sum_{C \in \{V, F, D\}} \sum_{K \in C} 1\{y_C = C_k\} \quad (6)$$

### 3.3. Hierarchical Softmax

The two-layer architecture seems to be shallow, to deal with millions of sentences; it has tremendous computing and updating workload of forward-propagation and back-propagation so it is not efficient enough. Morin and Bagnio introduced a hierarchical decomposition of the conditional probabilities that yielded a speed-up of about 200 built by using WordNet [11]. Mikolov use the vocabulary was represented as a Huffman binary tree, it is one of the similar strategy based on the observations that the frequency of words worked well for obtaining classes in neural net language models [12].

Complete vocabulary of training corpus does not contain the Word Net, therefore we choose Mikolov's version. With binary tree representations of the vocabulary, the number of prediction units that need to be evaluated can go down to around  $\log_2 |V|$ .

We first build a Huff-man binary tree according to the frequency of words. To adopt hierarchical softmax. Concretely, sort the word list by frequency and make the two lowest elements into leaves, creating a parent node with a frequency that is the sum of the two words' frequencies, repeat this step until all words are included in the tree. All words are leaf nodes, and words with low frequency have high depths and long binary codes.

The hierarchical softmax is a special multiclass logistic regression. Let  $N$  be the nodes on the path from the root to target word  $t$ , during the training of  $X_t$ , only a sub matrix of weight matrix  $\lambda$  is needed for forward-propagation and back-propagation, denoted by  $\lambda_N$ . The probability takes the form

$$P_i(X_t) = \frac{1}{1 + e^{-X_t \lambda_i}} \quad i \in N \quad (8)$$

In the same way as softmax regression, the cost function of, hierarchical softmax is minimized by means of gradient descent,

$$\lambda_i \leftarrow \lambda_i - \alpha \sum_{i \in N} X_t (y_i - P_i(X_t)) \quad (9)$$

$$X_t \leftarrow X_t - \alpha \sum \lambda_i (y_i - P_i(X_t)) \quad (10)$$

## 4. Implementation Details

### 4.1 File Storage

Upload the images and files. Image is uploaded based on the concept of OCR (optical character recognition). Optical character recognition is the mechanical or electronic conversion of images in to text format. First keyword analyzer analyzes the structure of images. Then extract the text from image by using xAlgent tool. Then convert the image in to text by using Interop.MODI.dll (NLP tool). This tool easily integrated OCR functionality to our application.

### 4.2 Word to Vector Conversion

The definition of term depends on the application. Typically terms are single words, keywords, or longer phrases. If words are chosen to be the terms, the dimensionality of the vector is the number of words in the vocabulary. Here we find each keywords with their vector values. It is not sufficient method compared with neural network graph model.

### 4.3 Hierarchical Softmax based regression

The NNGM include 2 softmax layer and one input layer. The softmax layer done the binary tree representation by using mikolov version (Sort the words based on frequency.). Then sort the words by using frequency, if the frequency is low it is considered as leaves and highest words is parent node. the softmax layer is meaningfully arranged the words. Softmax regression is a generalization of logistic regression to the case where we want to handle multiple classes. In this section include the related graph tree creation based on frequency. Then soft the word list by using frequency. Then parent node creation based on frequency.

### 4.4 Extract the feature term.

Based on Hierarchical softmax regression, frequency of each words are calculated from that we can extract the feature terms. Split the keywords and stored in to arraylist. This feature term saved in to CSV (comma separated value) format.

### 4.5 Mutual score calculation.

From the feature terms we can calculate the mutual score by using probability calculations. From that we can get the exact term pairs with its mutual score values.

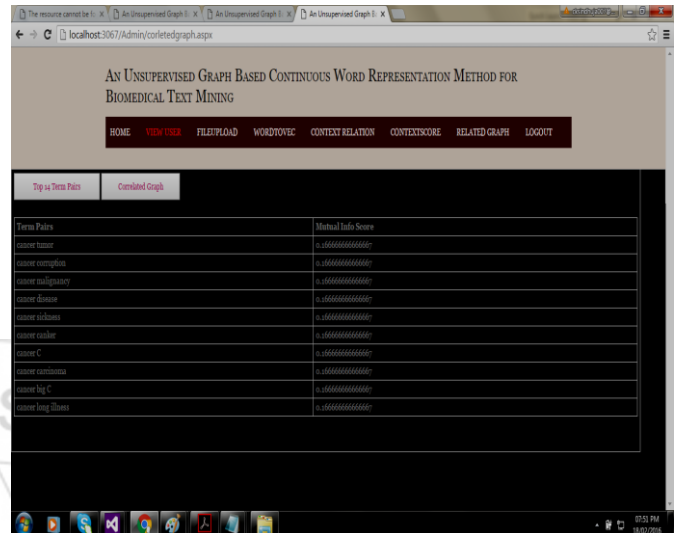
### 4.6 Neural Network Graph model

Then generate the correlated graph by using term pair and mutual score. We can get the straight-line graph with same value.

## 5. Result and discussions

### 5.1 Results

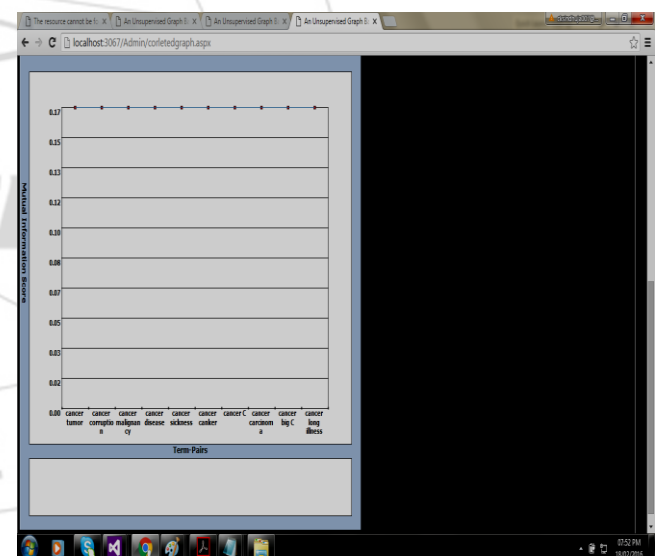
The fig 3 shows the term pairs with their mutual scores for the particular input. Fig 4 shows the correlated graph of NNGM with respect to mutual score and term pairs.



The screenshot shows a web browser window with the URL 'localhost:3067/Admin/colatedgraph.aspx'. The page title is 'AN UNSUPERVISED GRAPH BASED CONTINUOUS WORD REPRESENTATION METHOD FOR BIOMEDICAL TEXT MINING'. There are navigation links: HOME, VIEW DATA, FILEUPLOAD, WORDTOVEC, CONTEXT RELATION, CONTEXTSCORE, RELATED GRAPH, and LOGOUT. Below these links, there are two tabs: 'Term Pairs' and 'Correlated Graph'. The 'Term Pairs' tab is active, displaying a table with two columns: 'Term Pairs' and 'Mutual Info Score'.

Term Pairs	Mutual Info Score
cancer tumor	0.0000000000000000
cancer carcinoma	0.0000000000000000
cancer malignancy	0.0000000000000000
cancer disease	0.0000000000000000
cancer sickness	0.0000000000000000
cancer cancer	0.0000000000000000
cancer C	0.0000000000000000
cancer carcinoma	0.0000000000000000
cancer lung C	0.0000000000000000
cancer lung illness	0.0000000000000000

**Figure 3:** Term pairs with its mutual score from image uploading.



**Figure 4:** Correlated graph from image uploading.

### 5.2 Performance Evaluation

#### 1) Word Analogy

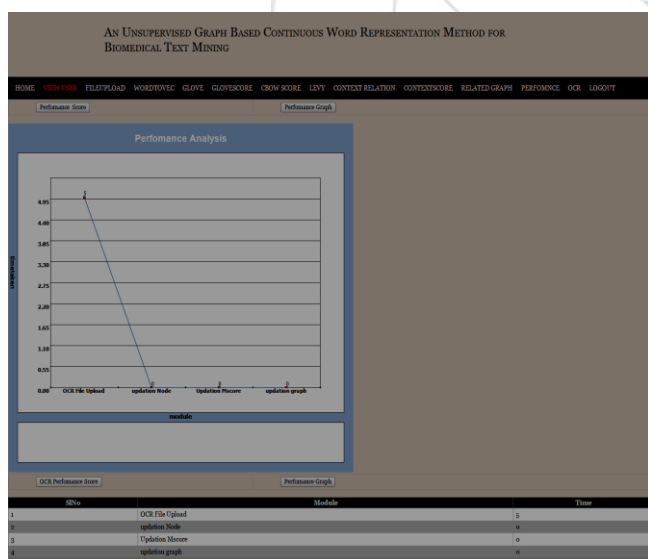
It can refer to relation between the source and target themselves. It consider the similarity between the words. The word analogy task consists of questions like, —a is to b as c is to d. The dataset contains 19544 such questions, divided into 14 categories, and each category contains a semantic subset and a syntactic subset. The semantic questions are typically analogies about people or places, like Athens is to Greece as Berlin is to \_\_?. The syntactic questions are typically analogies about verb tenses or forms of adjectives, for example —dance is to dancing as fly is to \_\_?. To correctly answer the question, the model should uniquely



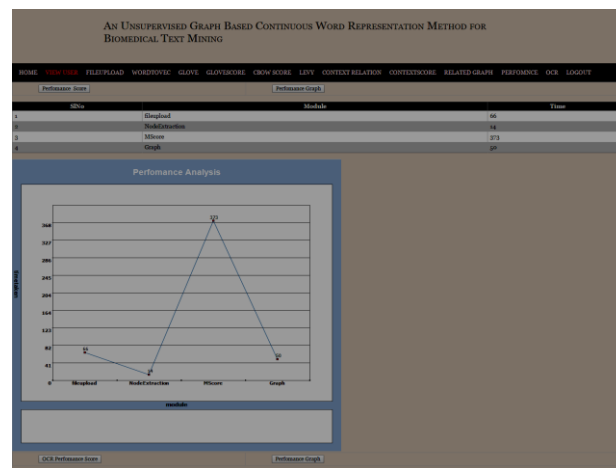
identify the missing term, with only an exact correspondence counted as a correct match.

## 2) PPIE

Large number of protein components organized by their PPIE. Some time the variation lead to disease. PPIE aims to find a criteria to judge whether a pair of proteins actually implies interaction or not according to the biomedical text that mentions them. Five publicly available PPI corpora extracted from Medline contain interaction annotation: AImed , BioInfer, HPRD50, IEPA , LLL . For example, according to a description from AImed, —The binding of hTAFII28 and hTAFII30 requires distinct domains of hTAFII18l, one can infer that hTAFII28 interacts with hTAFII30. To automatically extract protein interactions, the model should classify the PPI candidates into two groups, positive ones and negative ones. In order to evaluate the performance of word representation, reducing the influence by crafted features and tricks like kernel methods, we extract shallow word features and use a L1 regularized logistic regression (L1-LR) based binary classification model to address the problem. All evaluation results will be reported using the F-score. For PPIE, we perform pair-wise 10-folds cross-validation (randomly partitioned) on each corpus and report the macro-average F-score. Precision (P) is the ratio between the number of PPIs correctly detected and the total number of PPIs that were found by the system. Re-call (R) is the ratio between the number of PPIs correctly detected and the total number of PPIs in the gold standard. F-score is the harmonic mean of precision and recall.



**Figure 5: OCR performance**



**Figure 6: NNGM performance**

## 6. Conclusion

Introduced a new unsupervised neural network graph model with image to text by using the concept of OCR(Optical character recognition.).Our model leverages the relations between words, including dependency relations and context relations. State-of-the-art performance on word analogy task and PPIE task achieve our word representation. The major contributions can be summarized as follows:

- 1)We present a.Neural network graph model with image to text by using the concept of OCR(Optical character recognition.) which is task-irrelevant and can be embedded into any biomedical text mining tasks.
- 2)Model considers richer information, i.e., dependency relations and context relations among words, and it outperforms other representation methods on word analogy task, especially on semantic subtask.
- 3)Many biomedical text mining applications the word vectors are trained by NNGM with image uploading using OCR. The evaluation results on PPIE shows better performance with other word representation models when trained on large corpora, while outperforms other models when trained on small corpus.
- 4)The PPIE task also suggests that NNGM with image uploading using OCR is not sensitive to vector dimension, which makes it a good choice for saving computational cost, and more training iterations can improve the performance of NNGM.

## References

- [1] G. Salton, A. Wong, and C. S. Yang, —A Vector Space Model for Automatic Indexing,|| *Commun. ACM*, vol. 18, no. 11, pp. 613–620, 1975.
- [2] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, —Indexing by latent semantic analysis,|| *J. Am. Soc. Inf. Sci.*, vol. 41, no. 6, pp. 391–407, 1990.
- [3] R. Socher, J. Pennington, E. H. Huang, A. Y. Ng, and C. D. Manning, —Semi-supervised Recursive Autoencoders for Predicting Sentiment Distributions,|| in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2011, pp. 151–161. J.

- Yu, R. Hao, F. Kong, X. Cheng, J. Fan, and Y. Chen, "Forwardsecure identity-based signature: Security notions and construction," *Inf. Sci.*, vol. 181, no. 3, pp. 648–660, 2011
- [4] Katrin Fundel, Robert Kuffner and Ralf Zimmer-RelEx—Relation extraction using dependency parse trees| on November 28, 2006
- [5] Tomas Mikolov, Ilya Sutskever, Kai Chen - Distributed Representations of Words and Phrases and their Compositionality|16 october2013.
- [6] Angus Roberts, Robert Gaizauskas, Mark Hepple - Extracting Clinical Relationships from Patient Narratives Ohio, USA, June 2008. c 2008 Association for Computational Linguistics
- [7] Isabel Segura-Bedmar, Paloma Mart'inez, Mar'ia Herrero-Zazo - SemEval-2013 Task 9 : Extraction of Drug-Drug Interactions from Biomedical Texts (DDIExtraction 2013)| June 14-15, 2013. c 2013 Association for Computational Linguistics
- [8] Tomas Mikolov, Wen-tau Yih, Geoffrey Zweig, Linguistic Regularities in Continuous Space Word Representations"| Atlanta, Georgia, 9–14 June 2013.
- [9] Changqin Quan, Fuji Ren –Gene–disease association extraction by text mining and network analysis.| April 26-30 2014
- [10] Omer Levy and Yoav - Dependency-Based Word Embeddings,| - European Community's Seventh Framework Programme 2013
- [11] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, —A Neural Probabilistic Language Model,| *J. Mach. Learn. Res.*, vol. 3, pp. 1137–1155, 2003.
- [12] T. Mikolov, K. Chen, G. Corrado, and J. Dean, —Efficient Estimation of Word Representations in Vector Space,| *arXiv Prepr. arXiv1301.3781*, 2013

### Author Profile



**Jinsha V.S** is pursuing her M.Tech degree in Computer Science and Engineering from M Dasan institute of technology, Calicut University. She obtained her B.Tech Degree in Computer Science and Engineering from Anna university chennai, in 2014.