

Deep Neural Network and Pseudo Relevance Feedback Based Query Expansion

Abhishek Kumar Shukla* and Sujoy Das

Department of Mathematics, Bio-Informatics and Computer Applications, Maulana Azad National Institute of Technology Bhopal, Bhopal, Madhya Pradesh, 462003, India

*Corresponding Author: Abhishek Kumar Shukla. Email: abhishekshukla94@gmail.com

Received: 06 August 2021; Accepted: 13 September 2021

Abstract: The neural network has attracted researchers immensely in the last couple of years due to its wide applications in various areas such as Data mining, Natural language processing, Image processing, and Information retrieval etc. Word embedding has been applied by many researchers for Information retrieval tasks. In this paper word embedding-based skip-gram model has been developed for the query expansion task. Vocabulary terms are obtained from the top “k” initially retrieved documents using the Pseudo relevance feedback model and then they are trained using the skip-gram model to find the expansion terms for the user query. The performance of the model based on mean average precision is 0.3176. The proposed model compares with other existing models. An improvement of 6.61%, 6.93%, and 9.07% on MAP value is observed compare to the Original query, BM25 model, and query expansion with the Chi-Square model respectively. The proposed model also retrieves 84, 25, and 81 additional relevant documents compare to the original query, query expansion with Chi-Square model, and BM25 model respectively and thus improves the recall value also. The per query analysis reveals that the proposed model performs well in 30, 36, and 30 queries compare to the original query, query expansion with Chi-square model, and BM25 model respectively.

Keywords: Information retrieval; query expansion; word embedding; neural network; deep neural network

1 Introduction

Over the years the web has growing exponentially and it has become difficult to retrieve the relevant documents as per the user query. The information retrieval system tries to minimize the gap between the user query and relevant documents. Various phases of the retrieval process are affected by the vagueness of the user query. For example novice user during the formulation of the query, might be uncertain in selecting the keyword to express his/her information need. The user has only a fuzzy idea about what he/she is looking for. Due to this retrieval system retrieves irrelevant documents along with relevant documents. Query expansion appends additional terms to the original query and helps



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

in retrieving those additional relevant documents that were left out. Query expansion technique tries to minimize the word mismatch problem. Generally, queries are categorized into the following three main categories [1]

- (1) Navigational queries
- (2) Informational queries
- (3) Transactional queries

Navigational queries are those queries that are searching a particular URL or website. Informational queries are those which search a broad area of the given query and may contain thousands of documents. Transactional queries are those which search user intention to execute some task like downloading or buying some items. In information retrieval, one method of query expansion could be the use of semantically similar terms to the original query. WordNet [2] based methods are one of the oldest methods for query expansion. It is a semantic-based approach that finds semantically similar terms of the original query terms by using synonyms, hyponyms and, meronyms of the query terms. Word embedding is a technique to find similar terms to the original query. Word2vec [3] and Glove [4] are the two well-known word embedding techniques to find the semantically similar terms to the original query terms for query expansion. Word2vec and Glove learns the word embedding vector in an unsupervised way using a deep neural network. Word2vec and Glove find the semantically similar term of original query terms using global document collection or external resources such as Wikipedia [5] or similarity thesaurus [6]. The local method of query expansion searches the similar term of the original query using the Pseudo relevance feedback method. The pseudo relevance feedback method assumes that top “k” retrieved documents are relevant to the original query. It is observed that the local method of query expansion performs better than the global method of query expansion [7].

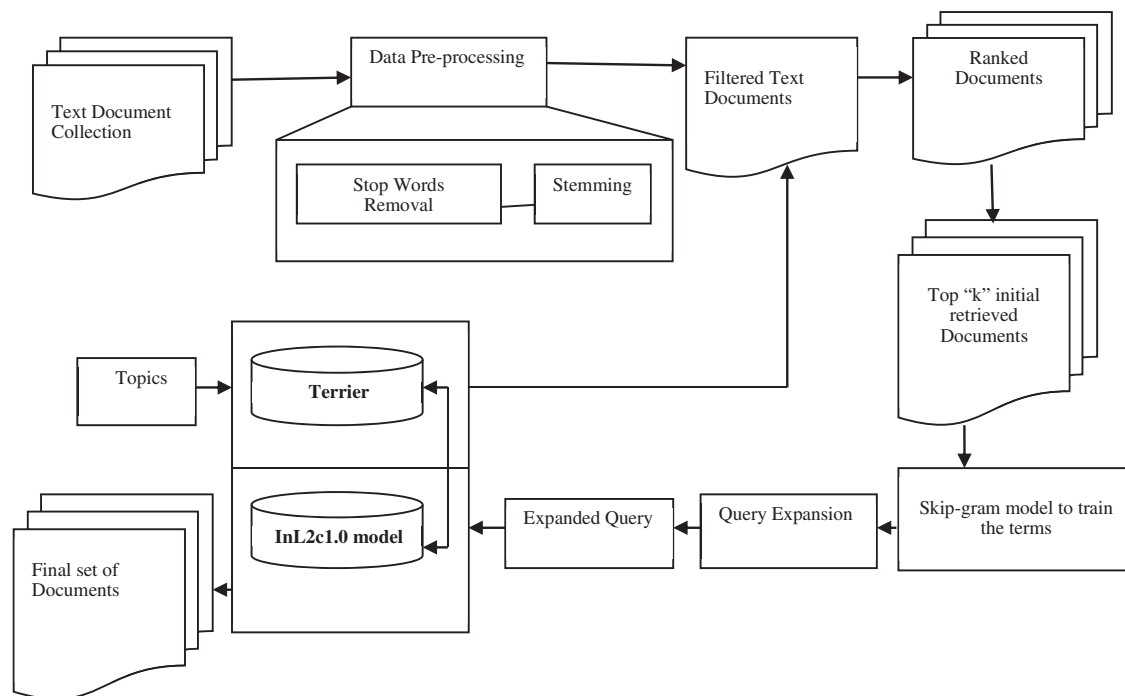


Figure 1: Architecture of proposed model

The proposed method uses a deep neural network-based query expansion method using the skip-gram model. In the proposed method of query expansion semantically similar terms of the original query are retrieved from top “k” initial retrieved documents using the Pseudo relevance feedback method. Semantically similar terms are retrieved by training the terms in top “k” initially retrieved documents using the skip-gram model. In the skip-gram model, we predict the context word of the given center word. The Skip-gram model uses an un-supervised deep neural network-based training method that successively updates the weight between two successive layers. The Skip-gram model assigns each term to a lower-dimensional vector compare to the vocabulary size, in a semantic vector space. The proposed method predicts the context word of each query term and then finds the union of these context words. The combined context words are treated as expansion terms for the given query terms. Fig. 1 shows the architecture of the proposed model.

2 Related Work

Query expansion plays important role in improving the performance of the retrieval system. The most common method of query expansion is to extract the expansion terms from an external data collection such as Anchor text, Query log, and external corpus. References [8,9] used anchor text as a data source. References [10,11] used query log for query expansion. They applied correlation between query terms and documents term. They collected data source from click-through of documents on URL. Reference [12] used query log as a bipartite graph where query nodes are connected to URL nodes by click edges and they showed an improvement of 10%. Reference [13] proposed co-occurrence-based document-centric probabilistic model for query expansion. A continuous word embedding-based technique for the document was proposed by [14]. They reported that their model performs better than LSI based model but does not outperform TF-IDF and divergence from the randomness model. Reference [15] proposed supervised embedding-based term weighting technique for language modeling. Reference [16] proposed semantic similarities between vocabulary terms to improve the performance of the retrieval system. Reference [17] proposed word embedding technique in a supervised manner for query expansion. Reference [18] proposed word embedding-based word2vec based model for expanding the query terms. Using this model they extracted similar terms of the query terms using the K-nearest neighbor approach. They reported considerable improvement on TREC ad-hoc data. Reference [19] used Word2Vec and Glove for query expansion for ad hoc retrieval. Reference [20] used fuzzy method to reformulate and expand the user query using pseudo relevance feedback method that uses top ‘k’ ranked document as a data source. Reference [21] proposed a hybrid method that uses both local and global methods as a data source. The proposed method used a combination of external corpus and top ‘k’ ranked documents as a data source. Reference [22] used a combination of top retrieved documents and anchor text as a data source for query expansion. Reference [23] used query log and web search result as a data source for query reformulation and expansion. Reference [24] used Wikipedia and Freebase to expand the initial query. Reference [25] used a fuzzy-based machine learning technique to classify the liver disease patients. Reference [26] proposed a machine learning technique that diagnoses breast cancer patients using different classifiers.

3 Query Expansion Using Deep Learning

Deep learning is a technique that is used in almost every area of computer science to learn something. In information retrieval, continuous word embedding is widely used to improve the mean average precision (MAP). There are following two deep learning approaches of word embedding technique

- (1) The Continuous Bag of Words model (CBOW) [27]
- (2) The Skip-gram model

Continuous bag of word model and Skip-gram model is widely used in query expansion method [28,29]. A continuous bag of word model is used to predict the center word of given context words. The Skip-gram model is just the opposite of the CBOW model. The Skip-gram model predicts the context word of a given center word. In this paper Skip-gram model is used to expand the query. The proposed method predicts context words for each query term and then they are combined and treated as expansion terms.

The proposed model is having three-layer of architectures, Input layer, hidden layer, and output layer. The proposed model used both the feed-forward network and the back-propagation method to predict the context word of a given center word. In the skip-gram model architecture, each query word is represented as one-hot encoding at the input layer. In one hot encoding representation if vocabulary size is 7000 words then in a 7000X1 vector is created and 0 is put at each index except at the index containing the center word. "1" is put at the index of the center word. The architecture of the skip-gram model is shown in Fig. 2. In the following diagram, the weight matrix is initialized randomly. Hidden layer is used to represent the one hot encoding into a dense representation. This is achieved through the dot product of the hot vector and weight matrix. At the next layer, we initialize another weight matrix with random weights. Then the dot product of a hidden vector and newly weighted matrix is obtained. At the next layer, activation function softmax is applied to the output value of the product of the hidden vector and newly assigned weight matrix. In the mid of the training, we have to change the weight of both the matrix so that the words surrounding the context words have a higher probability at the softmax layer. Let N represents the number of unique terms in our corpus of text, X represents the one hot encoding of our query word at input layer, N' number of neurons in the hidden layer, W(N'XN) weight matrix between the input layer and hidden layer, W'(NXN') weight matrix between the hidden layer and output layer, and Y a softmax layer having probabilities of every word in vocabulary, then using feed-forward propagation we have

$$h = w^T \cdot x$$

and

$$u = w'^T \cdot h$$

Let u_j be the j th neuron of layer u, w_j be the word in our vocabulary where j is any index, and V_{w_j} be the j th column of matrix W' then we have

$$u_j = V_{w_j}^T \cdot h$$

$$y = \text{softmax}(u)$$

$$y_j = \text{softmax}(u_j)$$

Y_j denote the probability that w_j is a context word

$$P(w_j|w_i) = y_j = \frac{e^{u_j}}{\sum_{j'}^N e^{u_{j'}}$$

$P(w_j|w_i)$ is the probability that w_j is a context word w_i is the input word. The goal is to maximize $P(w_{j^*}|w_i)$ where j^* represents the indices of context words. We have to maximize

$$\prod_{c=1}^C \frac{e^{u_{j_c^*}}}{\sum_{j'}^N e^{u_{j'}}} \tag{1}$$

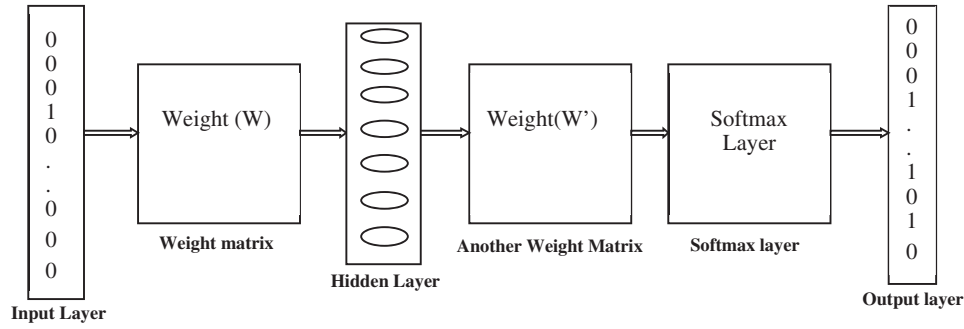


Figure 2: Deep skip-gram model architecture

where y_c^* are the vocabulary indices of context words. Context words are range from $c = 1, 2, 3, \dots, C$. The loss function E is defined as negative log of Eq. (1) as

$$E = -\log \left\{ \prod_{c=1}^C \frac{e^{u_{j_c^*}}}{\sum_{j'}^N e^{u_{j'}}} \right\}$$

or

$$E = -\sum_{c=1}^C u_{j_c^*} + C \cdot \log \left\{ \sum_{j'=1}^N e^{u_{j'}} \right\}$$

Using Back Propagation we have

$$\frac{\partial E}{\partial W'_{ij}} = \frac{\partial E}{\partial u_j} \cdot \frac{\partial u_j}{\partial W'_{ij}}$$

$$\begin{aligned} \frac{\partial E}{\partial u_j} &= -\sum_{c=1}^C 1 + C \cdot \frac{1}{\sum_{j'=1}^N e^{u_{j'}}} \frac{\partial}{\partial u_j} \left\{ \sum_{j=1}^N e^{u_j} \right\} \\ &= -\sum_{c=1}^C 1 + \sum_{j=1}^N y_j \end{aligned}$$

$$\frac{\partial E}{\partial u_j} = y_j - t_j = e_j \tag{2}$$

$$\frac{\partial E}{\partial W'_{ij}} = e_j \cdot \frac{\partial u_j}{\partial W'_{ij}} = e_j \cdot \frac{\partial W'_{ij} \cdot h_i}{\partial W'_{ij}} = e_j \cdot h_i$$

Now
$$\frac{\partial E}{\partial W_{ij}} = \frac{\partial E}{\partial u_j} \cdot \frac{\partial u_j}{\partial W_{ij}}$$

$$\begin{aligned}
&= \frac{\partial E}{\partial u_j} \cdot \frac{\partial u_j}{\partial h_i} \cdot \frac{\partial u_j}{\partial W_{ij}} \\
&= e_j \cdot W'_{ij} \cdot \frac{\partial W_{ij} \cdot x_i}{\partial W_{ij}} \\
\frac{\partial E}{\partial W_{ij}} &= e_j \cdot W'_{ij} \cdot x_i
\end{aligned} \tag{3}$$

The loss function is propagated from output layer to hidden layer and hidden layer to input layer from Eqs. (2) and (3). The weight W and W' is updated as

$$W'_{ijNew} = W'_{ij} - e_j \cdot h_i \tag{4}$$

$$W_{ijNew} = W_{ij} - e_j \cdot W'_{ij} \cdot x_i \tag{5}$$

where W_{ijNew} and W'_{ijNew} are the updated weights between input layer and hidden layer and hidden layer and output layer respectively. The algorithm of the proposed method is

Algorithm 1: SKIP-GRAM BASED QUERY EXPANSION

1. Create hot vector X , **for each** query term “ t ” for user query Q .
 2. Initialize skip_window_size=1, epoch=k, voc_size=N, hid_size=N' weight_matrix1=W(N'XN), weight_matrix2=W'(NXN').
 3. Using Feed Forward:
 - 3.1 Compute $h = w^T \cdot x$, $u = w'^T \cdot h$
 - 3.2 Compute $softmax(u_j) = \frac{e^{u_j}}{\sum_{j'=1}^N e^{u_{j'}}$
 - 3.3 $y = softmax(u)$
 - 3.4 **for** $j \leftarrow 1$ **to** N :
 - 3.4.1 $y_j \leftarrow softmax(u_j)$
 4. Using Back Propagation:
 - 4.1 **for** $i \leftarrow 1$ **to** N' :
 - 4.1.1 **for** $j \leftarrow 1$ **to** N :
 - 4.1.1.1 Compute $e_j \leftarrow y_j - t_j$
 - 4.1.2 Compute $e_{ji} = e_j \cdot h_i$
 - 4.1.3 Compute $e'_{ij} \leftarrow e_j \cdot W'_{ij} \cdot x_i$
 - 4.2 **for** $m \leftarrow 1$ **to** epoch:
 - 4.2.1 **for** $I \leftarrow 1$ **to** N' :
 - 4.2.1.1 **for** $j \leftarrow 1$ **to** N :
 - 4.2.1.1.1 $W'_{ji} \leftarrow W'_{ji} - e_{ji}$
 - 4.2.1.1.2 $W_{ij} \leftarrow W_{ij} - e'_{ij}$
 5. len $\leftarrow Length(Q)$
 6. leq $\leftarrow \lceil l/len \rceil$
 7. **for each** t in query Q retrieve the indices at y which has top “leq” values.
 8. **for each** t in query Q retrieve the words corresponding to indices at y and merge them to Q_m
 9. Append these words to $exp \leftarrow Q + Q_m$
 10. **return** exp
-

4 Experimental Results and Discussion

Precision and recall are the two metrics to check the performance of the retrieval system. A retrieval system with high precision and recall gives an implication to the evaluators that the proposed system is highly significant. Precision is defined as

$$precision = \frac{|{\{relevant\ documents\}} \cap {\{retrieved\ documents\}}|}{|{\{retrieved\ documents\}}|}$$

and recall as

$$recall = \frac{|{\{relevant\ documents\}} \cap {\{retrieved\ documents\}}|}{|{\{relevant\ documents\}}|}$$

Mean average precision (MAP) defined as

$$MAP = \frac{1}{N} \sum_{j=1}^N \frac{1}{Q_j} \sum_{i=1}^{Q_j} P(doc_i)$$

where

Q_j : number of relevant documents for query j

N : number of queries

$P(doc_i)$: precision at ith relevant document

We have performed our experiment on *FIRE 2011 English test collection* [30]. The dataset is of size 1.1 GB containing 392577 documents. We have used *terrier3.5* [31] search engine as retrieval engine. The documents are pre-processed through the following steps.

- o **Text Segmentation:** To split the text into sentences and then to split the sentence into tokens.
- o **Stop Word Removal:** This step remove all the stop words containing the documents.
- o **Stemming:** This step stem the root words of all the terms containing the documents.

We have performed pre-processing on the underlying dataset by applying *Porter stemmer and Stopword* to stem the root word and to remove the stop words respectively. In the proposed method documents are retrieved using InL2c1.0 model. We have performed our experiment on 50 queries. The mean average precision value of the proposed method, query expansion with the Chi-Square model, BM25 model, and the original query are 0.3176, 0.2912, 0.2970, and 0.2979 respectively. An improvement of 6.61% to the original query is observed. The performance of the retrieval system on original query, Chi-squared based query expansion, query expansion using the proposed model and BM25 model are shown in [Tabs. 1–4](#) respectively. The performance of original query vs. query expansion using Chi-Square, original query vs. proposed model, Chi-Square vs. proposed model, and original query vs. Chi-Square vs. proposed model is shown in [Figs. 3–6](#) respectively. From [Tabs. 1–4](#) it is observed that the proposed model outperforms to original query model, query expansion with the Chi-Square model, and BM25 model respectively. The MAP improvement of the proposed model to the original query and query expansion with Chi-square is 6.61% and 9.07% respectively. [Figs. 3–6](#) query by query analyses reveal that the proposed model retrieves 84 and 35 more relevant documents in comparison to the original query and query expansion with the Chi-Square model. The proposed model also performs well on 30 queries in comparison to and in 36 queries in comparison to the original query and query expansion with the Chi-Square model respectively. The sample query and their expansion terms using the proposed model are shown in [Tab. 6](#). In the Following Figures x-axis represents query number and y-axis represents MAP value respectively.

Table 1: Original query

Number of queries	50
Retrieved	50000
Relevant	2761
Relevant retrieved	2322
MAP	0.2979
R precision	0.3188
P@5	0.4480
P@10	0.4280
P@20	0.3900
P@30	0.3553
P@50	0.3152
P@100	0.2620

Table 2: Query expansion using Chi Square model

Number of queries	50
Retrieved	50000
Relevant	2761
Relevant retrieved	2381
MAP	0.2912
R precision	0.3083
P@5	0.4160
P@10	0.3880
P@20	0.3650
P@30	0.3500
P@50	0.3096
P@100	0.2516

Table 3: Query expansion using proposed model

Number of Queries	50
Retrieved	50000
Relevant	2761
Relevant retrieved	2406
MAP	0.3176

(Continued)

Table 3: Continued

R precision	0.3457
P@5	0.5160
P@10	0.4640
P@20	0.4190
P@30	0.3847
P@50	0.3424
P@100	0.2700

Table 4: BM25 model

Number of Queries	50
Retrieved	50000
Relevant	2761
Relevant Retrieved	2325
MAP	0.2970
R precision	0.3230
P@5	0.4400
P@10	0.4240
P@20	0.3860
P@30	0.3533
P@50	0.3168
P@100	0.2612

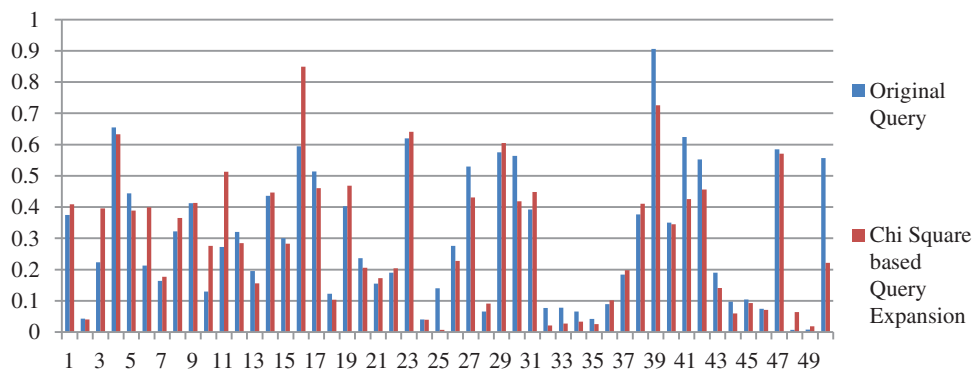


Figure 3: Performance of Chi Square based query expansion vs. original query

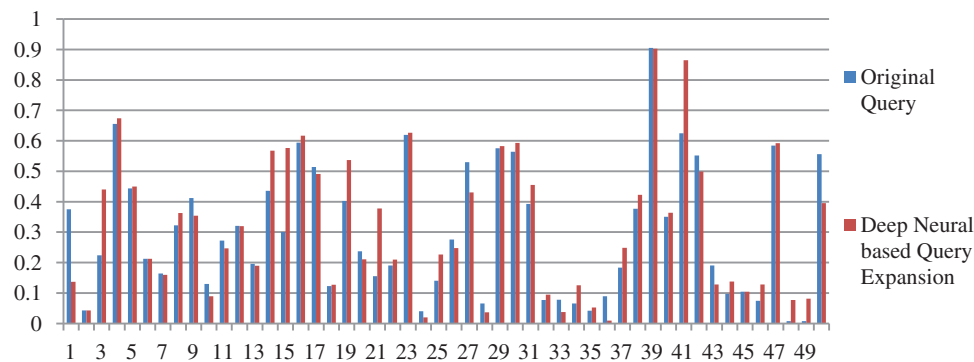


Figure 4: Performance of proposed model vs. original query

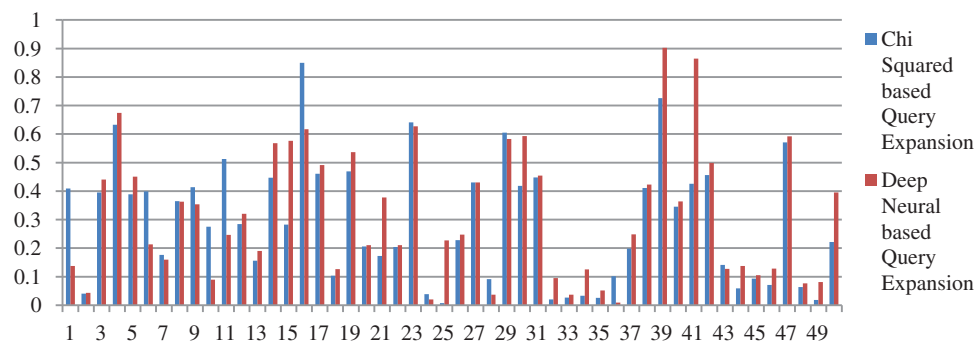


Figure 5: Performance of proposed model vs. Chi Square model

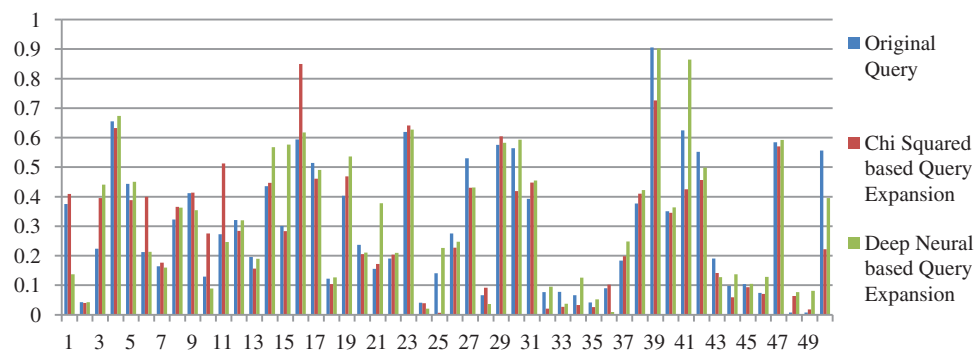


Figure 6: Performance of proposed model vs. Chi Square model vs. original query

5 Discussion

From [Tabs. 1–5](#) it is clear that the proposed model performs well over the other models. The proposed model improves the MAP result 6.61%, 6.93%, and 9.07% concerning original query, query BM25 model, and query expansion with Chi-Square model respectively. The proposed model also improves the result on R precision parameter 8.47%, 7.02%, and 12.13% concerning original query, BM25 model, and query expansion with Chi-Square model respectively. The proposed model improves

recall value by retrieving 84, 25, and 81 additional documents compare to the original query, query expansion with the Chi-Square model, and BM25 model respectively. From Figs. 4, 5, and 7 it is clear that per query analysis reveals that out of 50 queries proposed model performs well in 30, 36, and 30 queries compare to the original query, query expansion with the Chi-Square model, and BM25 model respectively. Per query, analysis shows that more than 60% of queries proposed model performs well compare to other models. The proposed model performs well compare to the original query in query numbers Q128, Q129, Q130, Q131, Q133, Q139, Q140, Q141, Q143, Q144, Q146, Q147, Q148, Q150, Q154, Q155, Q156, Q157, Q159, Q160, Q162, Q163, Q165, Q166, Q169, Q170, Q171, Q172, Q173, and Q174. The proposed model also performs well compare to query expansion with the Chi-Square model in query numbers Q127, Q128, Q129, Q130, Q137, Q138, Q139, Q140, Q142, Q143, Q144, Q145, Q146, Q147, Q150, Q151, Q152, Q155, Q156, Q157, Q158, Q159, Q160, Q162, Q163, Q164, Q165, Q166, Q167, Q169, Q170, Q171, Q172, Q173, Q174, and Q1175. The proposed model also performs well compare to the BM25 model in query numbers Q127, Q128, Q129, Q130, Q132, Q133, Q139, Q140, Q141, Q143, Q144, Q146, Q147, Q148, Q150, Q154, Q155, Q156, Q157, Q159, Q160, Q162, Q163, Q165, Q166, Q169, Q171, Q172, Q173, and Q174.

Table 5: Comparison of relative performance of proposed model with other models

	MAP (%)	R Precision (%)	P@50 (%)
Original Query	6.61	8.43	8.62
BM25 Model	6.93	7.02	8.08
Chi Square Model	9.07	12.13	10.59

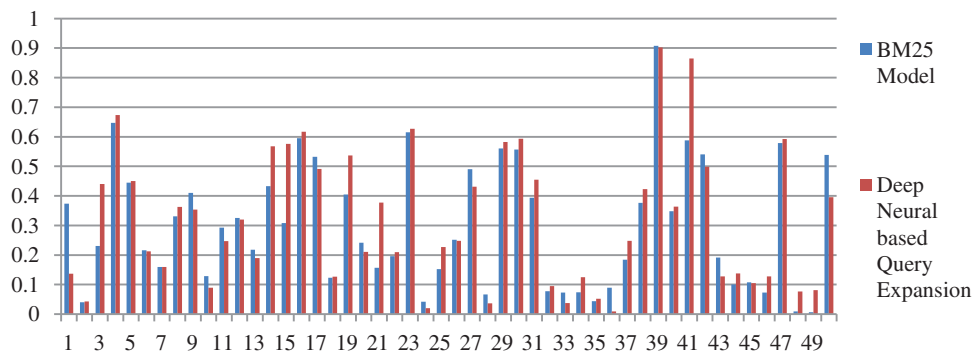


Figure 7: Performance of proposed model vs. BM25 model

Table 6: Sample query and their expansion terms using proposed model

Query no.	Original query	Expansion terms
Q130	Price hike of petroleum products	Increase, rise, petrol, fuel, rs, ministry, prices, oil, duties

(Continued)

Table 6: Continued

Query no.	Original query	Expansion terms
Q133	Torture at the Abu Ghraib prison	Iraqi, Iraqis, abuses, abuse, detainees, jail, guantanamo, us, military, prisoners
Q148	Popularity of social networking sites	Professionals, father, com, clear, networks, games, online, software, web, like, popular, media, facebook, users, using
Q152	Successful missile test in India	Pakistan, range, defence, brahmos, successfully, 26, ballistic, launch, cruise, capable
Q154	A. Raja and the 2G Spectrum scam	Housing, telecom, allocation, resigned, cbi, licences

6 Conclusion

In this paper, the word mismatch problem is minimized by applying combination of pseudo relevance feedback and deep neural network based method. In the proposed method, we have applied the skip-gram-based neural method for selecting the expansion terms. The mean average precision of the proposed method is 0.3176. An improvement of 6.61% and 9.07% is observed on MAP parameter in comparison to the original query and query expansion with Chi-square model respectively. The proposed model also retrieves 84 and 35 more documents in comparison to original query and query expansion with Chi-square model respectively. In near future, we will try to further improve the performance of the proposed method by tuning the parameters.

Acknowledgement: One of the authors is pursuing a full-time Ph.D. from the Department of Mathematics, Bio-informatics, and Computer Applications, Maulana Azad National Institute of Technology (MANIT) Bhopal (MP), India. He expresses sincere thanks to the Institute for providing an opportunity for him to pursue his Ph.D. work. The author also thanks the Forum of Information Retrieval and Evaluation (FIRE) to provide a dataset to perform his experimental work.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] A. Broder, "A taxonomy of web search," *ACM SIGIR Forum*, vol. 36, no. 2, pp. 3–10, 2002.
- [2] J. Bhogal, A. MacFarlane and P. Smith, "A review of ontology based query expansion," *Information Processing & Management*, vol. 43, no. 4, pp. 866–886, 2007.
- [3] T. Mikolov, K. Chen, G. Corrado and J. Dean, "Efficient estimation of word representations in vector space," *ArXiv Preprint ArXiv: 1301.3781*, 2013.
- [4] J. Pennington, R. Socher and C. D. Manning, "Glove: Global vectors for word representation," in *Proc. of the 2014 Conf. on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, pp. 1532–1543, 2014.

- [5] D. N. Milne, I. H. Witten and D. M. Nichols, "A knowledge-based search engine powered by Wikipedia," in *Proc. of the Sixteenth ACM Conf. on Information and Knowledge Management*, Lisboa, Portugal, pp. 445–454, 2007.
- [6] Y. Qiu and H. P. Frei, "Concept based query expansion," in *Proc. of the 16th Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, Pittsburgh, PA, USA, pp. 160–169, 1993.
- [7] Y. Kakde, "A survey of query expansion until June 2012," Bombay: Indian Institute of Technology, 2012.
- [8] V. Dang and B. W. Croft, "Query reformulation using anchor text," in *Proc. of the Third ACM Int. Conf. on Web Search and Data Mining*, New York, USA, pp. 41–50, 2010.
- [9] R. Kraft and J. Zien, "Mining anchor text for query refinement," in *Proc. of the 13th Int. Conf. on World Wide Web*, New York, USA, pp. 666–674, 2004.
- [10] H. Cui, J. R. Wen, J. Y. Nie and W. Y. Ma, "Query expansion by mining user logs," *IEEE Transactions on Knowledge and Data Engineering*, vol. 15, no. 4, pp. 829–839, 2003.
- [11] J. R. Wen, J. Y. Nie and H. J. Zhang, "Query clustering using user logs," *ACM Transactions on Information Systems*, vol. 20, no. 1, pp. 59–81, 2002.
- [12] Z. Yin, M. Shokouhi and N. Craswell, "Query expansion using external evidence," in *European Conf. on Information Retrieval*, Springer, Berlin, Heidelberg, pp. 362–374, 2009.
- [13] S. Bhatia, D. Majumdar and P. Mitra, "Query suggestions in the absence of query logs," in *Proc. of the 34th Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, Beijing, China, pp. 795–804, 2011.
- [14] S. Clinchant and F. Perronnin, "Aggregating continuous word embeddings for information retrieval," in *Proc. of the Workshop on Continuous Vector Space Models and Their Compositionality*, Sofia, Bulgaria, pp. 100–109, 2013.
- [15] G. Zheng and J. Callan, "Learning to reweight terms with distributed representations," in *Proc. of the 38th Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, Santiago, Chile, pp. 575–584, 2015.
- [16] D. Ganguly, D. Roy, M. Mitra and G. J. Jones, "Word embedding based generalized language model for information retrieval," in *Proc. of the 38th Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, Santiago, Chile, pp. 795–798, 2015.
- [17] T. Mikolov, I. Sutskever, K. Chen, G. Corrado and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proceedings of the 26th International Conference on Neural Information Processing Systems*, Lake Tahoe, Nevada, USA, pp. 3111–3119, 2013.
- [18] D. Roy, D. Paul, M. Mitra and U. Garain, "Using word embeddings for automatic query expansion," *ArXiv Preprint ArXiv: 1606.07608*, 2016.
- [19] F. Diaz, B. Mitra and N. Craswell, "Query expansion with locally-trained word embeddings," *ArXiv Preprint ArXiv: 1605.07891*, 2016.
- [20] J. Singh, M. Prasad, O. K. Prasad, E. M. Joo, A. K. Saxena *et al.*, "A novel fuzzy logic model for pseudo-relevance feedback-based query expansion," *International Journal of Fuzzy Systems*, vol. 18, no. 6, pp. 980–989, 2016.
- [21] K. Collins-Thompson and J. Callan, "Query expansion using random walk models," in *Proc. of the 14th ACM Int. Conf. on Information and Knowledge Management*, Bremen, Germany, pp. 704–711, 2005.
- [22] B. He and I. Ounis, "Combining fields for query expansion and adaptive query expansion," *Information Processing & Management*, vol. 43, no. 5, pp. 1294–1307, 2007.
- [23] H. Wu, W. Wu, M. Zhou, E. Chen, L. Duan *et al.*, "Improving search relevance for short queries in community question answering," in *Proc. of the 7th ACM Int. Conf. on Web Search and Data Mining*, New York, USA, pp. 43–52, 2014.
- [24] J. Dalton, L. Dietz and J. Allan, "Entity query feature expansion using knowledge base links," in *Proc. of the 37th Int. ACM SIGIR Conf. on Research & Development in Information Retrieval*, Queensland, Australia, pp. 365–374, 2014.
- [25] P. Kumar and R. S. Thakur, "An approach using fuzzy sets and boosting techniques to predict liver disease," *Computers Materials & Continua*, vol. 68, no. 3, pp. 3513–3529, 2021.

- [26] V. Kumar, “Evaluation of breast computationally intelligent techniques for cancer diagnosis,” *Neural Computing and Applications*, vol. 33, no. 8, pp. 3195–3208, 2021.
- [27] T. Mikolov, K. Chen, G. Corrado and J. Dean, “Efficient estimation of word representations in vector space,” *ArXiv Preprint ArXiv: 1301.3781*, 2013.
- [28] B. Aklouche, I. Bounhas and Y. Slimani, “Query expansion based on NLP and word embeddings,” in *Text Retrieval Conference, TREC 2018*, Gaithersburg, Maryland, USA, 2018.
- [29] A. Imani, A. Vakili, A. Montazer and A. Shakery, “Deep neural networks for query expansion using word embeddings,” in *European Conf. on Information Retrieval*, Springer, Cham, pp. 203–210, 2019.
- [30] S. Palchowdhury, P. Majumder, D. Pal, A. Bandyopadhyay and M. Mitra, “Overview of FIRE 2011,” in *Multilingual Information Access in South Asian Languages*, Springer, Berlin, Heidelberg, pp. 1–12, 2013.
- [31] C. Macdonald, R. McCreadie, R. L. Santos and I. Ounis, “From puppy to maturity: Experiences in developing terrier,” in *OSIR@ SIGIR*, Portland, Oregon, USA, pp. 60–63, 2012.