Intelligent Automation & Soft Computing
DOI: 10.32604/iasc.2023.027429
Article

Tech Science Press

SF-CNN: Deep Text Classification and Retrieval for Text Documents

R. Sarasu[1,*], K. K. Thyagharajan[2] and N. R. Shanker[3]

[1]Computer Science and Engineering, Dhanalaksmi College of Engineering, Anna University, Chennai, India
[2]R. M. D Engineering College, Anna University, Chennai, India
[3]Computer Science and Engineering, Aalim Muhammed Salegh College of Engineering, Anna University, Chennai, India
*Corresponding Author: R. Sarasu. Email: sar1234rag@gmail.com
Received: 17 January 2022; Accepted: 13 March 2022

Abstract: Researchers and scientists need rapid access to text documents such as research papers, source code and dissertations. Many research documents are available on the Internet and need more time to retrieve exact documents based on keywords. An efficient classification algorithm for retrieving documents based on keyword words is required. The traditional algorithm performs less because it never considers words' polysemy and the relationship between bag-of-words in keywords. To solve the above problem, Semantic Featured Convolution Neural Networks (SF-CNN) is proposed to obtain the key relationships among the searching keywords and build a structure for matching the words for retrieving correct text documents. The proposed SF-CNN is based on deep semantic-based bag-of-word representation for document retrieval. Traditional deep learning methods such as Convolutional Neural Network and Recurrent Neural Network never use semantic representation for bag-of-words. The experiment is performed with different document datasets for evaluating the performance of the proposed SF-CNN method. SF-CNN classifies the documents with an accuracy of 94% than the traditional algorithms.

Keywords: Semantic; classification; convolution neural networks; semantic enhancement

1 Introduction

On the Internet, enormous text documents are available due to the increase in users. Researchers refer to text documents in research articles, coding and dissertations and need appropriate documents during retrieval. The increase in the number of research articles leads to redundant research papers, and retrieving correct documents based on keywords is challenging. Retrieval of research papers is a big challenge for information retrieval [1]. Customized learning content and storing the content in the databases in a hierarchical structure is disclosed to retrieve documents [2]. Adaptive e-learning services have been adopted to provide text documents to the user based on the learner's requirement. A multi-agent system is proposed to automatically retrieving relevant learning material from the Internet [3]. The hybrid technique based on machine learning retrieves research documents [4]. In recent days, deep learning methods have been used in Natural Language Processing (NLP). Deep learning is applied in the

Retrieval of research papers. Deep Neural Networks is an alternative approach to supervised learning, which automatically learns high-level representations of research papers for retrieval. Deep Neural Networks automatically learns the distributed representation of textual words sentences for text document retrieval. Convolution Neural Networks (CNN) and deep nets classify documents.

Scholarly search engine for research paper extraction and classification never shows appropriate documents related to keywords and shows overlapping research area articles. Moreover, results for the same keyword search on the Internet vary in retrieving documents. The researcher spends the maximum time obtaining a research paper. Scholarly search engine retrieval never shows structured materials such as magazines, thesis and journal papers. Moreover, duplication of a document appears. In Search Engine, lack of quality control and clear indexing guidelines are missing. As the database of research articles increases, the search engine fails to meet "one size fit all". The duplication is proportional to the size of the increased database. The tie search per query base quality is challenging for retrieving research papers. The proposed Semantic Featured Convolution Neural Networks (SF-CNN) contribution includes understanding keywords according to meaning and context. SF-CNN has a logical structure of words to identify the most relevant elements in the text and understand the topic. The proposed SF-CNN algorithm is applied for research paper retrieval from the different datasets of research articles. SF-CNN algorithm has semantic property for identifying the polysemy of each keyword through the linguistic unit for extracting the exact meaning of the word. The building structure of the SF-CNN algorithm provides concepts from each keyword with proper linking and exacts meaning according to the context.

2 Related Work

Research paper classification for retrieval of research article from the Internet is reviewed in this section. Text classification and information retrieval are applied for retrieving the research article. Traditional method of text classification algorithms are logistic regression, support vector machines. The above methods retrieve text document using voting, bagging and boosting based feature selection and improve retrieval performance. CNN algorithm is proposed and classify large textual document [5]. The existing deep learning optimization and max pooling. Traditional CNN never deal with the discontinuity of word. Large Scope Based CNN (LSS-CNN) based text retrieval according to input keywords have discontinuity of word. Base pooling convolution kernel is combined and perform pooling operation. The operation is based on the text n-gram feature. Three type of CNN are created and designed in pooling method [6]. The research article is classified based on summaries of research article. Web of science dataset is used for classification [7]. Fast text algorithm is used for the classification of text in Research article [8]. The classification is performed with Recurrent Neural Network (RNN) [9]. Deep learning model such as CNN and RNN, solve the problem of longer dependence of text [10]. Deep learning methods are used in NLP [11]. Minimized number of words are used for classification. RNN based controller use less number of word for text classification. ArXiv dataset consists of 1000 word and 11 class for text classification [12]. CNN–Glove and Domain Phrase -Attention Based Hierarchical Recurrent Neural Network (DPA-HNN) are used for information retrieval from radiology report [13]. Attention-Based Gating Mechanism (ABLG-CNN) use salient features of research documents to classify the text. The gating mechanism assign weight to Bidirectional-Long Short-Term Memory (BI-LSTM) and CNN with text fusion features and classify text [14]. Temporal feature is obtained from text data, and gating mechanism replace the max-pooling method for text feature classification [15]. Text classification is performed based on Deep Averaging Network (DAN) [16]. DAN is performed based on the sentence in the text document. Dependency Sensitive Convolutional Neural Networks (CNN) classify the sentence and document through Long Short-Term Memory network. Subsequently, features are extracted with convolution operator. CNN never depend on phrase labelling, and sentence level task, whereas other CNN model consider sentence, by sliding window of word in Research article [17]. CNN system capture both the dependency information and relationship across sentence for text classification. Dynamic

Convolutional Neural Network (DCNN) consists of Dynamic k-Max pooling in the architecture [18]. The Bag of words (BOW) and CNN classify text in Research article [19]. The semi-supervised structure with Convolution Neural network (CNNs) for text categorization is developed [20]. Instead of using word embedding in neural network, embed small text region from unlabeled data for integration into a supervised CNN is developed. Semi-Supervised Convolution Neural Network (SWNN) utilize the one-dimensional structure for word order of text data [21]. Fast clustering is applied for text classification [22]. Multi-Group Norm Constraint CNN (MGNC-CNN) is developed based on multiple word embedding using sentence classification [23]. Convolution neural network is used for relation classification of word, which help in search engine and reduce the error caused by annotator and linguistic expert [24]. The lightweight tool, Cerno" a framework, is proposed for semi-automatic semantic annotation of textual document. This framework was developed based on annotation schema and used in the classic vector space model [25]. The exploration technique is used for identifying the concept present in the documents. Text is segmented and finds the linguistic pattern in the text. The contextual exploration technique is used for determining clue in the text and document [26]. Multi-Level Semantic Representation Enhancement Network (MSRLEN) method enhance the semantic representation of word, phrase and context level of sentence [27]. The machine learning algorithm lacks background knowledge of word, and knowledge graph is used in machine learning algorithm. Machine learning enhance background knowledge required for training session in text classification [28]. Backpropagation Neural Network used for classification and analyzed membership function of the fuzzy set for text classification [29]. The Chinese text, which is never evenly distributed and hence Long Short Memory, Convolution Neural Network and attention algorithm model with weight for key features are used for classification of Chinese word [30]. Legal documents are classified based on Multi-label text classification [31]. The cloud dataset is classified based on image patches available on the cloud dataset [32].

3 Proposed Work

Many classification algorithms are developed using CNN, such as sentiment classification, Emotion classification, short sentence classification, document classification. Still, Researchers need efficient classification and retrieval of research articles. Till and now then, researchers have retrieved journal articles based on keywords. Semantically word-based retrieval of research articles is performed. Moreover, retrieval of journal articles is performed based on missing words in keywords. The linguistic patterns of keywords are used for journal article retrieval.

Furthermore, machine learning algorithms perform less in research-paper retrieval and ML is added with graph-based knowledge to understand the search word links for retrieving the research article. In this paper, the research articles are classified using SF-CNN. In SF-CNN, Semantic Featured Enhanced Layer is merged with maximum pooling layer. The structure of SF-CNN contains words and context when words and context have semantically enhanced the documents for classification and retrieval. In SF-CNN, two levels of enhancement, initially at word level through the BERT (Bidirectional Encoder Representations from Transformers) model. Finally, Semantic representation from word features is semantically enhanced by calculating the weighting between the words. At the context level, context is semantically enhanced by applying the context to BI-LSTM, and the context features are semantically enhanced.

The vectors are generated from both levels and input to the semantic fusion layer. After obtaining the representation from the fusion layer, the vectors are applied to the classification layer. When the user requests the paper based on keywords, documents are classified and retrieved using SF-CNN. The overall architecture of the proposed work is shown in Fig. 1. This paper uses word embeddings in SF-CNN and classify the text. In SF-CNN, Vector conversion initializes a matrix of word embeddings for training and classification. Word embeddings are vector representations of words, where words are mapped to vectors instead of a one-dimension space. Semantically, close words should have a similar vectors representation instead of a distinct representation. Vectors fed into the embedded layer; vectors converted by the tool

should be semantically enhanced. In SF-CNN, semantic enhancement is performed through text *corpus* word embedding. Extraction of Semantic features from text data through SF-CNN classification is obtained and improves the research paper retrieval. Before converting the word to vector, the topic or concept in the text document should be extracted. Aylien text analysis API is used for extracting the topic from the given text document. The ontology has 685 class hierarchies and 2,795 properties for text classification and document retrieval. In this paper, Word2Vec and GloVe convert words to a vector representation.

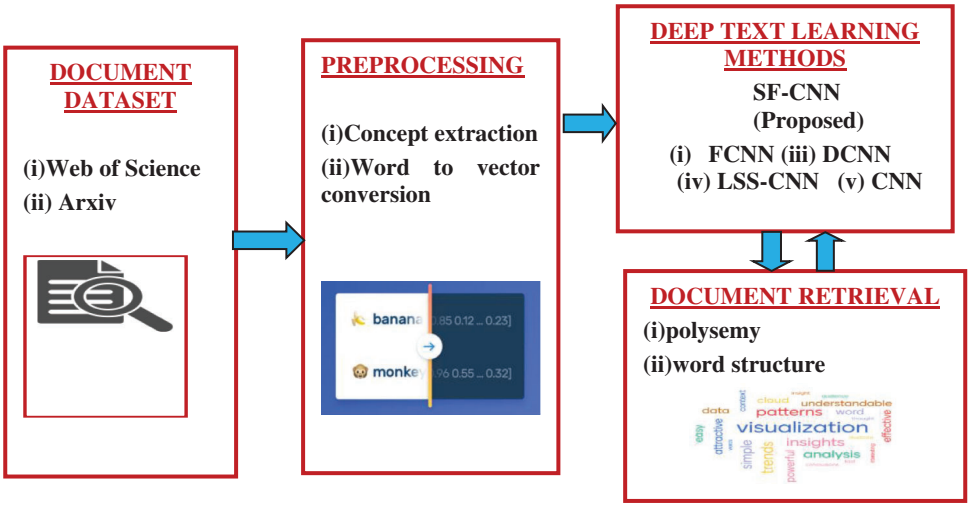


Figure 1: Proposed SF-CNN architecture diagram for retrieval of research papers

TextRazor API is used for higher accuracy converts words to vectors. TextRazor uses a set of machine learning algorithms and prolog to generate rules, and these rules are customized for the Research paper retrieval. TextRazor has a statistical topic classifier. Entity extractor provides solutions for searching, monitoring and creating content for text mining applications. The topic extractor is used with ontology for classification work. Textrazor is used for vector initialization. The proposed work is implemented in Python 3.7 programming language. This work is implemented in Windows 8.1 Pro (64-bit Operating system with Intel® Pentium Processor and 6.00 GB RAM with processor speed 1.3 GHz.

3.1 Document Retrieval Using SF-CNN Model

Traditional models such as Vector Space Model, Boolean Model, and Probabilistic Model retrieve the information quickly and inaccurately. In addition to traditional models, deep learning paved the alternate way in classifying information. The Convolution Neural Network is used for text classification. CNN is a type of feed-forward neural network consisting of three layers. The three layers include embedded, convolution, and pooling based on convolution filters. The convolution filter is used for extracting the features directly from the documents for research article classification based on the training set. Traditional CNN suffers from semantic representation during document retrieval applications. The application includes text classification and relationship extraction. In this paper, Semantic Featured CNN (SF-CNN) classifies the documents based on semantic representation. SF-CNN has BERT (Bidirectional Encoder Representations from Transformers) and BI- LSTM(Bidirectional-Long Short-Term Memory). The document has both words and context. When both words and context are semantically enhanced, the documents retrieval for text document accuracy is improved. The semantic enhancement performs through analyzing the polysemy words.

Polysemy is a word or phrase with different and related meanings. Polysemy is the concept of relatedness. For example, 'Bank' means a steep slope rising ground, bordering water, or slant an aeroplane tangentially. A financial institution offers services to deposit money or have an account in a bank. SF-CNN has two levels of enhancement. Before applying the words and context to BERT and BI-LSTM, the words and context are semantically connected with probase knowledge base. Probase is an ontological knowledge base used to find the relatedness of words and context. At the word level, the BERT model performs the enhancement, and At the context level, context is further semantically enhanced by applying the context using the BI-LSTM method. The vectors are generated for both levels and input to the semantic fusion layer. After obtaining the representation from the fusion layer, vectors are applied to the classification layer. Feature vectors obtained from the embedded layer are fused and fed to the classification layer. In this paper, concatenated fusion merges the output vectors obtained from each channel. The vectors are generated for word building structure block and combined with concatenation fusion in Semantic Fusion Layer. Concatenation fusion is defined as in Eq. (1)

$$\text{Zconcat} = Ve1 \cup Ve2 \cup Ve3..Ve4 \tag{1}$$

3.2 Classification Layer of SF-CNN

The semantic representation infusion layer vectors the applied in the classification layer and calculated the classification results for text document retrieval. The classification layer consists of two layers: the Convolution layer and the max-pooling layer. The first layer is the convolution layer and consists of a convolution filter. The convolution is a binary operation over the text, and the convolution filter is the real matrix. The output of this binary operation is a single number. The convolution filter has the same dimension as the text segment matrix. The convolution filter is applied to the text segment of the research article using a sliding window and providing a similar real number. The continuous real numbers are called feature maps. The max-pooling layer performs with input features map, and the research papers are classified based on the probability estimates. The max-pooling layer produces probability distributions as the output. The base element in the model is a word vector $X \in R^{d}$, where 'd' is the dimension of word vectors. A document is represented as a matrix $D \in R^{n*d}$, where 'n' is the number of words and each row in the matrix represents the word vector. The convolution filter is $F \in R^{h*d}$, where 'h' is the number of words. Convolution is a commutative process flipped over the kernel. The 2-D convolution operation is defined as in Eq. (2)

$$G = CF * D_{j:j+h-1} \tag{2}$$

Mapping 'h' word window to the real number is as in Eq. (3)

$$c_j = f\left(CF * D_{j:j+h-1} + b\right) \tag{3}$$

where 'b' is the bias term and 'f' is rectified linear function. After convolution, operation the feature map for the document is represented as Eq. (4)

$$c(CF) = c_1, c_2, c_3, c_4, \ldots, C_{n-h+1} \tag{4}$$

Max pooling operation perform as in the equation Eq. (5)

$$\left(C_{cf}\right) = maxc(CF) \tag{5}$$

The traditional CNN model is used for the classification of sentences. CNN classify the documents based on differences in the layer. In this paper, SF-CNN classifies the documents with three convolution layers and three pooling layers. All three convolution layers are input to the pooling layer and output from pooling layers and merged to get a single feature. Singe feature is called a label, which is used for classification. The SF-CNN algorithm is as given as below,

Algorithm 1: Classification and Retrieval Algorithm for SF-CNN

Input: *Documents taken from dataset |D|, Testing set* X_{test}

Parameters: *V-size-vocabulary size (1876) filter size- number of words in convolution filter(56), no_filter-The number of filters(3)-filter list [3,4,5], Seq-length –Sentence Length [56], Embedding-size-dimensions of Embedding(300).*

//Embedding layer – maps the vocabulary of words into vector representation

For each data point in the dataset, do

Embed(Semantic Featured Enhancement algorithm)

W=embed(V-size, Embedding-size)

End for

//Convolution Layer-Tensors obtained from the Embedding layer is given as input to convolution and also produces tensors of various shapes of filters. Then results are merged to get big feature vectors by applying the nonlinearity function.

For each filter_size in filter_sizes do

fil-shape = [filter_size, embedding_size, 1, no_filters]

conv = conv2d(W)

// Apply nonlinearity

h = relu(add(conv, b)

End for

Semantic Featured Enhancement algorithm

Input: *Words taken from training set* X_{train},

For each wordin the Training Set | Xtrain| do

Constructs Semantic featured CNN

Words[Xtrain] =Enhance Semantic Feature;

Vectors[Vi] =Embed (Word[Xtrain)]

Semantic _Fusion _Layer [ ]=Vectors[Vi]

End For

For each context in the Training Set | Xtrain| do

Constructs Semantic featured CNN

context[Xtrain] =Enhance Semantic Feature;

Vectors[Vi] =Embed (context [Xtrain)]

Semantic _Fusion _Layer [ ]=Vectors[Vi]

Output:Vectors Semantically Enhanced Vector[$Ve_{1,\ldots Ve4}$*]*

//Max Pooling layer-Pooling reduces the output dimensionality.

Pooled_output = max_pool(h, ksize= [1, seq_length - filter_size + 1, 1, 1]

// Combine all the pooled features

*num_filters_total = num_filters * len(filter_sizes)*

pooled = concat(3, pooled_outputs)

Output: *The class label Ctest the class label as output.*

4 Experimental Setup and Results

CNN performance is analyzed for various datasets for research paper retrieval after classification. The process of evaluations is suitable to research articles dataset, the semantic enhancement and classification model. The research articles are gathered from the ArXiv and Web of Science datasets to assess the SF-CNN model's performance. These research articles are used for classification and retrieval. ArXiv has a collection of research papers in various domains, including Physics, Mathematics, Biology and subdisciplines of Computer Science. The dataset has 11 classes and 33388 papers. The research article is in PDF format, and these articles are converted to the text document. Text documents are used for classification in the SF-CNN method, detail of papers and the average number of words used for classification as shown in Tab. 1. The dataset assigns several tags to one research paper. For example, a paper related to Neural networks assigned to areas like Artificial Intelligence or Computer Vision.

Table 1: Data statistics taken from ArXiv Data set

Class Name	No. of Documents	Average Words
cs.AI(Artificial Intelligence)	2995	6212
cs.CE(Computational Engineering)	2505	5777
cs.CV(Computer Vision)	2525	5630
cs.DS (Data Structures)	4136	7439
cs.IT (Information Theory)	3233	5938
cs.NE(Neural Evolutionary)	3012	5856
cs.PL(Programming Languages)	2901	7012

Web of Science database published in the year 2017 and provides comprehensive dataset from different academic disciplines. This dataset contains research articles and is used to classify and retrieve the research paper; the science database contains research articles from various major domains such as Computer Science, Electrical Engineering, Psychology, Mechanical Engineering, Medical Science, and biochemistry. This database consists of 35,238 research articles.

4.1 Performance of SF-CNN for Document Retrieval

To evaluate the SF-CNN performance for the web of science dataset is shown in Tab. 2. During the training phase in SF-CNN, a set of words are used for training. For example, the number of iterations is fixed to 30 rounds. Tab. 2. depicts the number of sample words used in training the network. For example, examining SF-CNN with 1000 words, the model's accuracy is achieved up to 80%. Tab. 2. demonstrates the training and testing time batch-wise for classification and retrieval of a research article. The traditional CNN suffers, due to training time, whereas SF-CNN requires less training time for retrieval. Traditional Text-CNN algorithm has less accuracy as word size increases. When word size has increased, the accuracy of the SF-CNN model is increased and shown in Tab. 2.

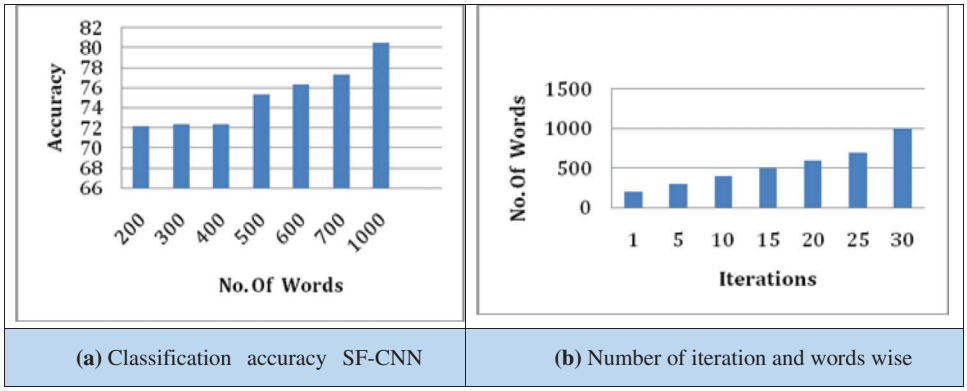
The training time is influenced by word size; when the word size increases, the training time increases. As shown in Tab. 2, the word size is 1000 the training time per batch is increased to 0.79 min s. Thus, time cost increases linearly with data size during the training process, and the accuracy of the validation set is decreased within 3 epochs.

Table 2: Computational efficiency of SF-CNN (Word-Wise) for text classification of documents

No. of Words	Parameter	Training Time Per min (s)	Testing Time Per min (s)	Accuracy
200 Words	2.10×10^5	0.13	0.10	72.14
500 Words	2.10×10^5	0.17	0.13	75.3
750 Words	3.10×10^5	0.20	0.13	77.3
1000 Words	8.7×10^5	0.79	0.63	80.47

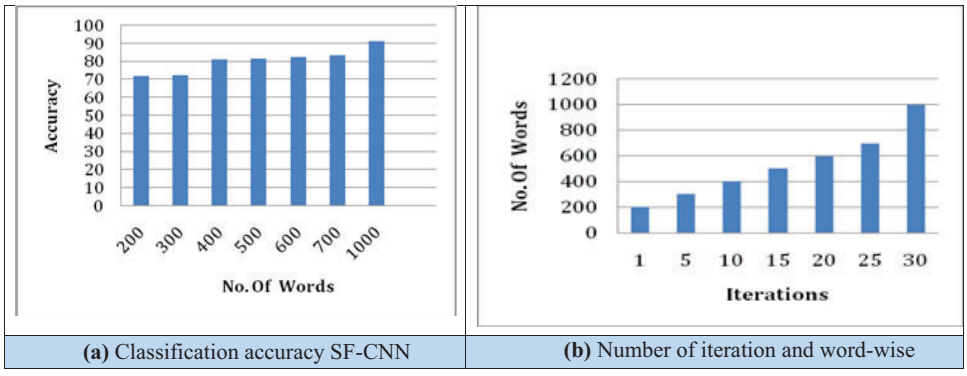
4.2 Computational Efficiency of SF-CNN (Wordwise) for Web of Science Dataset

Figs. 2 and 3 show SF-CNN's accuracy for the number of words used from the web of science and Arxiv datasets. The number of iterations is fixed at 30 for the Web of Science and Arxiv dataset. The accuracy is increased as the word size is increased. From the results, classification and retrieval of document accuracy were never affected by the number of keywords. When kernel size or filter size is fixed to 5, the classification accuracy of SF-CNN has reached about 91%.



(a) Classification accuracy SF-CNN **(b)** Number of iteration and words wise

Figure 2: (a) Classification accuracy SF-CNN, (b) Number of iteration and words wise



(a) Classification accuracy SF-CNN **(b)** Number of iteration and word-wise

Figure 3: (a) Classification accuracy SF-CNN, (b) Number of iteration and word-wise

4.3 Analysis of SFCNN- Parameters used for Classification of Documents

SF-CNN accuracy in classification document retrieval is affected by kernel size. Kernel size is essential for text classification. The Kernel reduces the computational costs for classification of documents and

retrieving the research article. The vector of weight and bias is called as filter and represent particular input feature. SF- CNN model use more than three kernel. Different convolution kernel for different filter is used in SF-CNN

Kernel, slide over the feature map of text vector. The filter slide over the input vector and should be consistent. The relationship between kernel size and the proposed SF-CNN model is consistent. Kernel play an important role in text classification. Kernel size should be appropriate for deleting the redundant word feature. As shown in Tab. 3, kernel show the classification accuracy for document retrieval. From the simulation result of SF-CNN, the relation between kernel size and accuracy are shown in Tabs. 3 and 4. Tab. 3 depict word size, kernel size, and classification accuracy for ArXiv and Web of Science dataset. The word size referred as vocabulary size and used for both testing and training. The kernel size should be appropriate and consistent for classifying documents and retrieval. For the ArXiv dataset, the kernel size is 3, and classification accuracy is 93.8%.

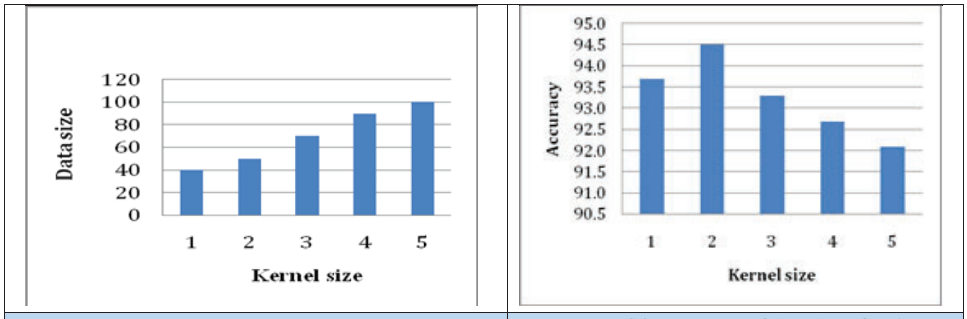
Table 3: Performance-based on: Kernel size for ArXiv

ArXiv Dataset		
Kernel size	Data size	Accuracy
1	40	93.5
2	50	94.8
3	70	93.8
4	90	92.3
5	100	93.5

Table 4: Performance-based on: Kernel size for web of science dataset

Web of Science Dataset		
Kernel size	Data Size	Accuracy
1	40	93.7
2	50	94.5
3	70	93.3
4	90	92.7
5	100	92.1

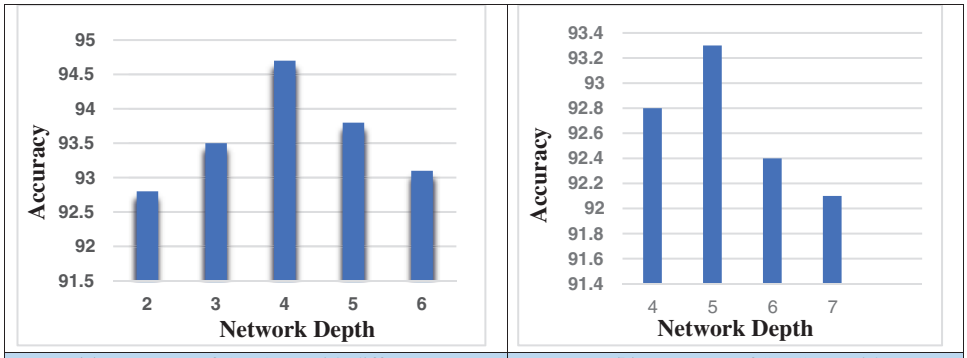
When the kernel size increased to 4, classification accuracy is about 92.3%. Kernel size will affect the feature. Figs. 4a and 4b shows the kernel size, data size and accuracy of proposed SF-CNN. In the web of science dataset, kernel size is 3, the accuracy of the SF-CNN classification is 92.7%. When the Kernel size is increased to 4, the accuracy of SF-CNN classification and documents retrieval declined to 92.1%. Figs. 5a and 5b shows the SF-CNN depth accuracy for the dataset. When the network depth is small, the kernel never capture the long dependence of text. When the depth of the network is increased, convolution layer lead to dilation.



(a) Kernel size – Data

(b) Accuracy of SF-CNN for the given kernel size

Figure 4: (a) Kernel size –Data, (b) Accuracy of SF-CNN for the given kernel size



(a) Accuracy of SF-CNN with different network depth- ArXiv dataset

(b) Accuracy of SF-CNN with different network depth- Web of Science dataset

Figure 5: (a) Accuracy of SF-CNN with different network depth- ArXiv datas, (b) Accuracy of SF-CNN with different network depth- Web of Science dataset

SF-CNN eliminate the dilation problem due to vectors. Irrespective of the dataset, the network depth should be consistent. When the network is increased in the proposed SF-CNN, increases the accuracy in documents classification and retrieval. The network depth is 5, and accuracy is increased to 93.8%. Thus network depth influences the accuracy of classification and document retrieval. The classification accuracy is based on a number of document classified. SF-CNN model is analyzed with traditional algorithm through F1 measure, which is defined using precision and recall, where 'tp' mean True Positive, 'fn' represent False Negative for SF-CNN accuracy measurement.

4.4 Parameter Setting of Network Depth in SF-CNN

The network's depth is considered the output parameter that controls the number of neurons in the layer, which connects the input. Increasing the depth in Convolution Neural Networks will never provide the expected results. In a deep Neural Network, when the depth model increases, the accuracy level increases and finally drops. Figs. 5a and 5b show the SF-CNN depth accuracy for the dataset. When the Network's depth is small, the kernel will never capture the long dependence of text. When the depth of the network is increased, the convolution layer will suffer from dilation. Whether it is the Web of Science dataset or ArXiv dataset, SF-CNN eliminates the dilation problem. Irrespective of the dataset, the network depth should be consistent. When the network is increased, SF-CNN increases the accuracy. The network depth

is 5 the accuracy is increased to 93.8%. Network depth influences the accuracy of classification. The SF-CNN model is analyzed with the traditional algorithm through precision and recall as in Eqs. (7) and (8).

$$Precision = \frac{tp}{tp + fp} \tag{6}$$

$$Recall = \frac{tp}{tp + fn} \tag{7}$$

$$f1\ measure = \frac{Precision.Recall}{Precision + Recall} \tag{8}$$

The proposed SF-CNN is compared with Attention Based BILSTM fused CNN (ABLG-CNN), Domain Phrase Attention Based Hierarchical Recurrent Neural Network (DPA-HNN), Recurrent Neural Network -Convolutional Neural Network (RNN-CNN), Large Scope Based CNN (LSS-CNN) and Deep Convolutional Network (DCNN) and shown in Fig. 6. Attention Based BILSTM fused CNN, ABLG-CNN with gating mechanism derive keyword information by calculating context vector. The gating mechanism is applied on BILSTM and CNN. The gating mechanism assigns weights to output generated from BILSTM and CNN and obtains the fusion vectors required for classification and document retrieval. Domain Phrase Attention Based Hierarchical Recurrent Neural Network (DPA-HNN) calculates synthesizing information of text reports. This model encodes hierarchical structures composed of sentence, word, and document levels. FCNN (Fuzzy Convolution Neural Network) uses fuzzy logic membership degree to refine text classification and document retrieval output. RNN-CNN is used in long documents classification by considering important pages using recurrent attention learning. It focuses on significant words, not all words present in the document [33]. Large Scope Based CNN(LSS-CNN) captures complicated local features based on convolution, aggregation optimization and max-pooling operation [34].

As depicted in Fig. 6, the proposed SF-CNN method outperforms the traditional methods. FCNN training is very difficult, requiring a large amount of training time. SF-CNN handles a large number of complex datasets. While comparing the SF-CNN with the traditional CNN, the performance difference is about 10%

as shown in Tab. 5. SF-CNN focus on word sequences and ignores the global word co-occurrence information. The proposed SF-CNN captures the semantic information present in the research paper. Attention-based BI-LSTM fused CNN with a gating mechanism (ABLG- CNN)weighting scheme for important features present in the document to extract local features and salient features.

4.5 Comparison of Classification Accuracy of SFCNN and FCNN for Web of Science Dataset

Fig. 7 shows the comparison of SFCNN and FCNN for the web of science dataset. The results show that the average accuracies of the SFCNN and FCNN for the research paper classification system are 94% and 75%, respectively. FCNN has the same layer as CNN and replaces the normal convolution layer with a fuzzy Convolution Layer. In contrast, the SFCNN replaces the CNN with a semantic enhancement layer based on polysemy words.

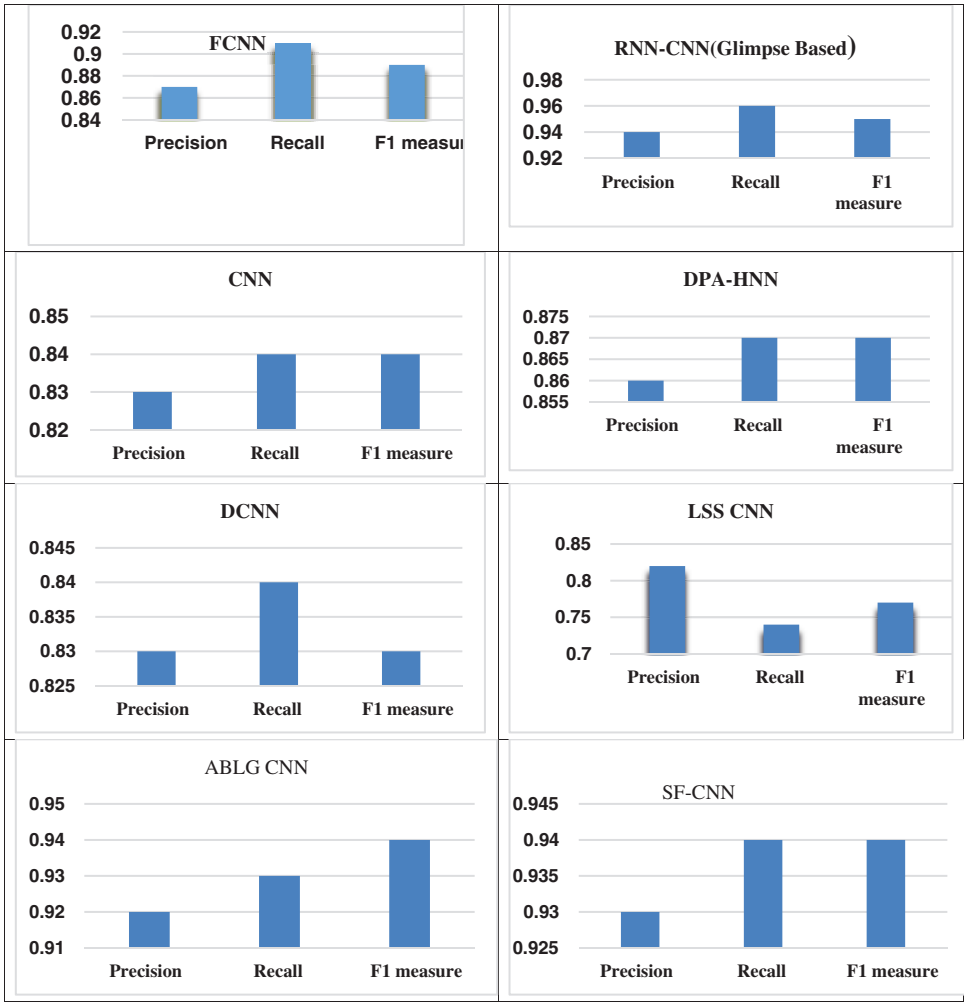


Figure 6: Comparison of SF-CNN with traditional methods

Table 5: Accuracy of SFCNN and FCNN

No. Of Words	CNN	CNN
200	72.14	67.4
300	72.4	69.4
400	81.4	73.3
500	81.7	71.7
600	82.4	72.4
700	83.3	83.3
1000	91.4	88.1

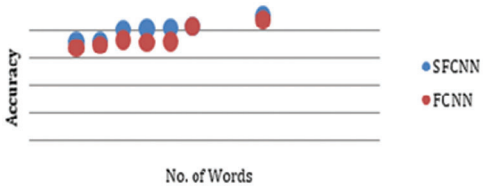


Figure 7: Accuracy of SFCNN and FCNN

5 Conclusion

In this paper, we proposed a deep learning-based SF-CNN method for solving the problem of semantic classification documents and retrieval problems. The effectiveness of the proposed SF-CNN is demonstrated by conducting experiments on two standard datasets like Arxiv and Web of Science. SF-CNN model can classify research documents based on the semantic feature of word and context. The semantic features of both word and context levels are identified using a semantic featured enhancement layer added to the embedded layer in SF-CNN. The feature vectors are generated for word context levels and combined for the fusion layer. The classification layer classifies the documents based on the vectors generated from the fusion layer. The proposed SF-CNN method enhances the semantic features for classifying and retrieving research documents better than traditional methods. Results obtained are compared with traditional methods like LSS-CNN, ABLG-CNN, CNN and DPA-HNN. The F1 measure result shows that the SF-CNN classifies the articles more accurately and retrieves the document with reduced time. The proposed SF-CNN method achieves the best performance of about 94%. Furthermore, SF-CNN can be improved with semi graph theory-based vectors.

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] J. Hsin, "Automatic generation of semantically enriched web pages by a text mining approach," *Expert System with Web Applications*, vol. 36, no. 6, pp. 9709–9718, 2009.

[2] K. K. Thyagharajan and R. Nayak, "Adaptive content creation for personalized e-learning using web services," *Journal Applied Sciences Research*, vol. 3, no. 9, pp. 828–886, 2007.

[3] A. Thukral, S. Datta and H. B. P. Bedi, "Informal e-learning using multi-agent systems," in *Proc. Int. Conf. on Intelligent System Design and Applications*, Kochi, India, vol. 1, pp. 34–67, 2012.

[4] A. A. Kardan and O. R. B. Speily, "Smart lifelong learning system based on Q-Learning," in *Proc. Int. Conf. on Information Technology: New Generations*, Las Vegas, Nevada, USA, vol. 1, pp. 108–1091, 2010.

[5] J. Wang, Y. Li, J. Shan, J. Bao, C. Zongand *et al.,* "Large-Scale text classification using scope-based convolutional neural network: A deep learning approach," Special scale on deep learning for big data, *IEEE Access*, vol. 7, no. 5, pp. 171548– 171558, 2019.

[6] S. Zeng, Y. Ma, X. Zhang and X. Du, "Term-based pooling in convolutional neural networks for text classification," *Journal of China Communications*, vol. 17, no. 8, pp. 109–124, 2020.

[7] M. Chouyyekhensias, H. Omar and M. Lazar, "Scientific paper classification using convolutional neural networks," in *Proc. Int. Conf. on Big Data and Internet of Things*, New York, vol. 2, pp. 1–6, 2019.

[8] T. Yao, Z. Zhai and B. Gao, "Text classification model based on fast text," in *Proc. Int. Conf. on Artificial Intelligent and Information System*, Dalina, China, vol. 1, pp. 45–54, 2020.

[9] K. Radhika, K. R. Bindu and L. Parameswaran, "A text classification model using convolution neural network and recurrent neural network," *International Journal of Pure and Applied Mathematics*, vol. 119, no. 7, pp. 1549–1554, 2018.

[10] J. Cai, J. Li, W. Li and J. Wang, "Deep learning model used in text classification," in *Proc. Int. Computer Conf. on Wavelet Active Media Technology and Information Processing*, Chengdu, China, vol. 5, pp. 78–89, 2019.

[11] T. Young, D. Hazarika, S. Poria and E. Cambria, "Recent trends in deep learning-based natural language processing," *IEEE Computational Intelligence*, vol. 13, pp. 55–75, 2018.

[12] J. He, Liu L.Wang, J. Feng and H. Wu, "Long document classification from local word glimpses via recurrent attention learning," *IEEE Access*, vol. 17, no. 9, pp. 40707–40728, 2019.

[13] I. Banerjee, Y. Ling and M. Chen, "Comparative effectiveness of convolution neural network (CNN) and recurrent neural network (RNN) architectures for radiology text report classification," *Journal of Artificial Intelligence in Medicine*, vol. 9, no. 7, pp. 79–88, 2019.

[14] J. Denga, L. Chenga and Z. Wang, "Attention-based bi-LSTM fused CNN with gating mechanism model for Chinese long text classification," *Journal of Computer Speech & Language*, vol. 68, no. 3, pp. 101182, 2020.

[15] J. Ren, W. Wu, G. Liu, Z. Chen and R. Wan, "Bidirectional gated temporal convolution with attention for text classification," *Journal of Neuro Computing*, vol. 455, no. 1, pp. 265–273, 2021.

[16] M. Iyyer, V. Manjunatha, J. B. Graber and H. Daum, "Deep unordered composition rivals syntactic methods for text classification," in *Proc. Int. Conf. on Natural Language Processing*, Beijing, China, vol. 1, pp. 15–1162, 2015.

[17] R. Zhang, H. Le and D. Radev, "Dependency sensitive convolutional neural networks for modelling sentences and documents," in *Proc. Int. Conf. on North American Chapter of the Association of Computational Linguistics*, San Diego, China, vol. 2, pp. 1512–1521, 2016.

[18] N. Kalchbrenner, E. Grefenstette and P. Blunsom, "A convolutional neural network for modelling sentences," in *Proc. Annual Meeting of Association for Computational Linguistics*, Baltimore, Maryland, vol. 1, pp. 23–27, 2014.

[19] C. N.Doss and S. M. Gatti, "Deep convolutional neural networks for sentiment analysis of short texts," in *Proc. Int. Conf. on Computational Linguistics*, Dublin, Ireland, vol. 1, pp. 69–78, 2014.

[20] R. Johnson and T. Zhang, "Effective use of word order for text categorization with convolution neural networks," *Journal of Computational and Language*, vol. 1, no. 12, pp. 103–112, 2015.

[21] R. Johnson and T. Zhang, "Semi-supervised convolutional neural networks for text categorization via region embedding," *Journal of Advanced Neural Inf. Process System*, vol. 1, no. 11, pp. 919–927, 2015.

[22] P. Wang, B. Xu, J. Xu, G. Tian, C. L. Liu *et al.,* "Semantic expansion using a word embedding clustering and convolutional neural network for improving short text classification," *Journal of Neuro Computing*, vol. 174, no. PB, pp. 806–814, 2016.

[23] Y. Zhang, S. Roller and C. B. Wallace, "MGNC-CNN: A simple approach to exploiting multiple word embeddings for sentence classification," in *Proc. Int. Conf. on North American Chapter of the Association of Computational Linguistics: Human Language Technologies*, San Diego, California, vol. 8, pp. 1522–1527, 2016.

[24] P. Qin, W. Xu and J. Guo, "An empirical convolutional neural network approach, semantic relation classification," *Journal of Neuro Computing*, vol. 190, no. C, pp. 806–814, 2016.

[25] N. Kiyavitskaya, N. Zeni, J. Cordy, L. Michand and J. Mylopoulos, "Cerno: Lightweight tool support for semantic annotation of textual documents," *Data & Knowledge Engineering*, vol. 68, no. 12, pp. 1470–1492, 2009.

[26] B. Smine, R. Faiz and J. P. Desclés, "Extracting relevant learning objects using a semantic annotation method," *International Journal of Metadata semantic and Ontologies*, vol. 8, no. 1, pp. 13–27, 2013.

[27] J. Liu, Y. Yang and H. He, "Multi-level semantic representation enhancement network for relationship extraction," *Journal of Neuro Computing*, vol. 403, no. 11, pp. 282–293, 2020.

[28] S. Bhatt, A. V. Shalin and J. Zhao, "Knowledge graph semantic enhancement of input data for improving AI," *Journal of IEEE Internet Computing*, vol. 24, no. 2, pp. 66–72, 2020.

[29] Y. Shiand and J. Zhao, "The semantic classification approach base on neural networks," *IEEE Access*, vol. 8, no. 12, pp. 1473, 2020.

[30] J. Xie, Y. Hou, Y. Wang, Q. Wan, B. Li *et al.,* "Chinese text classification based on attention mechanism and feature-enhanced fusion neural network," *Springer Computing*, vol. 102, no. 12, pp. 683–700, 2020.

[31] D. Song, A. Vold, K. Madan and F. Schilder, "Multi-label legal document classification: A deep learning-based approach with label-attention and domain-specific pre-training," *Journal of Information System*, vol. 1, no. 22, pp. 101718–101723, 2021.

[32] V. H. Phung and E. J. Rhee, "A high-accuracy model average ensemble of convolutional neural networks for classification of cloud image patches on small datasets," *Journal of Applied Sciences*, vol. 9, no. 21, pp. 21, 2019.

[33] J. He, L. Wang, L. Liu, J. Feng and H. Wu, "Long document classification from local word glimpses via Recurrent Attention Learning," *IEEE Access*, vol. 7, no. 23, pp. 40707–40718, 2019.

[34] J. Wang, Y. Li, J. Shan, J. Bao, C. Zongand *et al.,* "Large-scale text classification using scope-based convolutional neural network: A deep learning approach," *IEEE Access*, vol. 7, no. 29, pp. 171548–171558, 2019.