International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 10 Issue: 12
DOI: https://doi.org/10.17762/ijritcc.v10i12.5866
***Article Received**: 08 October 2022 **Revised**: 18 November 2022 **Accepted**: 04 December 2022*

On Optimality of Long Document Classification using Deep Learning

Ms. Ayesha Mariyam[1], SK. Althaf Hussain Basha[2], S. Viswanadha Raju[3]
[1]Research Scholar, CSE,
Jawaharlal Nehru Technological University,
Hyderabad, India.
ayesha.mariyam84@gmail.com
[2]Professor and Head
Computer Science and Engineering
Krishna Chaitanya Institute of Technology and Sciences, Markapur
althafbashacse@gmail.com
[3]Professor
Computer Science and Engineering
JNTUH College of Engineering, Jagtial
svraju.jntu@gmail.com

Abstract— Document classification is effective with elegant models of word numerical distributions. The word embeddings are one of the categories of numerical distributions of words from the WordNet. The modern machine learning algorithms yearn on classifying documents based on the categorical data. The context of interest on the categorical data is posed with weights and the sense and quality of the sentences is estimated for sensible classification of documents. The focus of the current work is on legal and criminal documents extracted from the popular news channels, particularly on classification of long length legal and criminal documents. Optimization is the essential instrument to bring the quality inputs to the document classification model. The existing models are studied and a feasible model for the efficient document classification is proposed. The experiments are carried out with meticulous filtering and extraction of legal and criminal records from the popular news web sites and preprocessed with WordNet and Text Processing contingencies for efficient inward for the learning framework.

Keywords- WordNet, Word2Vec, Vectorization, Recurrent Neural Networks , Convolution Neural Networks and PolicyNet

I. INTRODUCTION

Classification in text data has been becoming predominant in the recent years of research. Increasing attention of deep learning algorithms for classification of document has been in vogue since the emergence of deep learning in artificial intelligence and machine learning. Classification of text on web pages, emails, article publications discussion forums have top notch importance for the business systems to develop competitive intelligence and analyze the opinions of products, systems and people. The core of the research in document classification is related with the problems areas like detection of spam in emails, categorization of news articles of various interests. A seamlessly retrievable document is a well-organized and well classified document that can be accessed easily. A document is often variable, which increasingly adds up the content, where the retrievability becomes laborious, therefore documents become lengthy where, even designating labels and categorization of documents needs etymological expertise and personnel with technical knowledge. Experts with limited knowledge and cognitive capacities could not classify the documents accurately determining the labels and categorize horizontally or vertically. The intervention of artificial intelligence methods proves reduced time and cost and ensures accuracy in classification of long documents. Since a decade and above, artificial intelligence and machine learning algorithms are pioneered as guaranteed approaches in automatic classification of document for knowledge management. Although, traditional methods compete to the new generation methods, the deep learning methods rate good score of appreciation amongst all the algorithms for document classification.

Machine learning algorithms are said to be ingenious for the tasks of automatic document classification for knowledge management. Several methods of traditional importance prevail, such as K-nearest neighbors, support vector machine, which are not suitable of classifying long documents and are insufficiently reliable for real world applications. The deep learning methods ushered in the era of cognitive learning has demonstrated the novel capabilities in principles of classification. The method had focused on

Article Received*: 08 October 2022* ***Revised****: 18 November 2022* ***Accepted****: 04 December 2022*

technical documents of length with the support of NLP techniques to develop an automated classifier. Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) networks and several other specific pre-trained models are the key constituents in the frameworks developed for automatic classifiers.

II. TEXT CLASSIFICATION

Text classification is to classify the text of a document. The domain knowledge plays important role in order classify the text, various topics of the domain are considered as the range of classifiers. Mere text of a document is classified in the text classification in order to classify the entire document. Various kinds of domains like health and medical care systems, legal and crime documents, property documents and their chain of responsibilities, research documents and their bibsonomical connections, reviews on products, reviews on movies etc,. furthermore, text classification is also helpful in document classification in the areas of detecting the spam, opinion analysis, detection of hostility, labeling topics and meta information of search engines.[1][11]

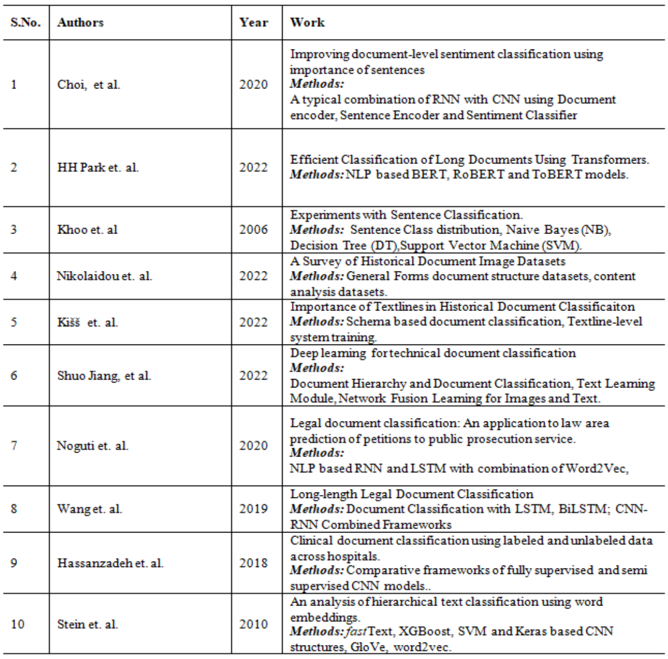
In this work, long document classification is emphasized other than short text classification activities. A long document contains multiple paragraphs and multiple sentences. More number of words are lexically connected to throw the impact of the meaning of the context, where the classifiers are built on such aspects of the document. Maintenance of the length of the document with respect to maintenance of the context relevancy amongst more number of paragraphs, sentences and words is a challenging task for document preservation data structures. In other fold, the task of document classification with respect to text classification become more precise and simpler, as there is large scope to derive the conceptual meaning of the words. Various types of algorithms for classification need to be studied in order to understand their performance on various types of long documents[11].

The recent developments of deep learning for text classification has revolutionary innovations, where the text is categorized in many levels such as word level, sentence level and context level. Texts containing the key words which are necessary for the built up of classifier for word level classification are found as fixed-size vectors called word-embedding. The embedding generally encodes semantic and contextual information about the word or the sentence of the document and then is processed to mold into a model in order to classify the text and provide specific meaning of the text. Deep learning techniques are innovative in building and assigning the classifiers that can understand the complex relationships among the words-topics-sentences with possible sets of embedding to evolve into a best category.

Most of the traditional models are vanilla models such as Naïve Bayes, SVM etc,. Methods of text classification have been advanced to complex BERT models [3]. The recent works on long document classification focus on transformer models. Many shallow neural network architectures are trained from scratch and allow inferences to deep learning approaches intimidating with the probabilistic and statistical models. The BERT and Transformer models have variants of modifications in order to adapt to the domains of long document classification [2].

Bidirectional Encoder Representation from Transformer is a model developed at Google for Natural Language Processing tasks which are believed to be state-of-the-art pre-training architectures for various kinds of NLP tasks, however the limitation for this architecture is a long input [2][4].

TABLE 1: Comparative Studies of Text Classification using Deep Learning approaches.



S.No.	Authors	Year	Work
1	Choi, et al.	2020	Improving document-level sentiment classification using importance of sentences ***Methods:*** A typical combination of RNN with CNN using Document encoder, Sentence Encoder and Sentiment Classifier
2	HH Park et. al.	2022	Efficient Classification of Long Documents Using Transformers. ***Methods:*** NLP based BERT, RoBERT and ToBERT models.
3	Khoo et. al	2006	Experiments with Sentence Classification. ***Methods:*** Sentence Class distribution, Naive Bayes (NB), Decision Tree (DT),Support Vector Machine (SVM).
4	Nikolaidou et. al.	2022	A Survey of Historical Document Image Datasets ***Methods:*** General Forms document structure datasets, content analysis datasets.
5	Kišš et. al.	2022	Importance of Textlines in Historical Document Classificaiton ***Methods:*** Schema based document classification, Textline-level system training.
6	Shuo Jiang, et al.	2022	Deep learning for technical document classification ***Methods:*** Document Hierarchy and Document Classification, Text Learning Module, Network Fusion Learning for Images and Text.
7	Noguti et. al.	2020	Legal document classification: An application to law area prediction of petitions to public prosecution service. ***Methods:*** NLP based RNN and LSTM with combination of Word2Vec,
8	Wang et. al.	2019	Long-length Legal Document Classification ***Methods:*** Document Classification with LSTM, BiLSTM; CNN-RNN Combined Frameworks
9	Hassanzadeh et. al.	2018	Clinical document classification using labeled and unlabeled data across hospitals. ***Methods:*** Comparative frameworks of fully supervised and semi supervised CNN models..
10	Stein et. al.	2010	An analysis of hierarchical text classification using word embeddings. ***Methods:*** *fast*Text, XGBoost, SVM and Keras based CNN structures, GloVe, word2vec.

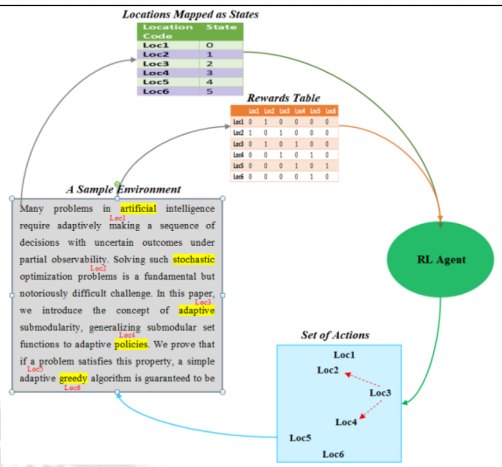
III. POLICY NETWORK (PNET)

As it is a tedious challenge to achieve the long document classification by understanding all the documents' inferences in terms of sentences, particular selection of sentences based on the context are moved towards classification of the document. In a document, all the sentences may not concentrate on the context of the subject; sentences which represent relevant vocabulary related to the context of the subject are selected. The weighted approach for the word embeddings is one of the ways which can be agreed

***Article Received**: 08 October 2022 **Revised**: 18 November 2022 **Accepted**: 04 December 2022*

upon for classification of the document based on the sentences, which represent weights of the word-embeddings. The weights assigned to the words in the sentences are just quantification of the words drawn from the count or the average occurrence score of the word in the document. Apart from the frequency or repetitions of the word in the document, the word shall be given weight based on the importance. The said task can be achieved by employing an agent, which collects all the relevant words of the context. An agent is deployed specifically for supporting the learning operations on the words of a specific context. Building of such agents is very crucial, where in some applications bots are used to build their context agents. An agent therefore is aware of various states, the actions at each state is fixed or a variable. In the case of fixed action at each state, the network is trained for the fixed actions at each state. The variable actions determine the decision making at each state, where training and testing of actions at each state are based on the probability or the experience. The reinforcement learning mechanism trains the network for a specific set of tasks using agents.

Policy is the fundamental element of reinforcement learning with agents. The agents are guided with right choices making the agent adaptable to the environment. Theoretically various types of policies come under the purview of an agent, such as – deterministic policy, stochastic policy, categorical policy and Gaussian policies. In a typical construction of the policy in learning, a policy tells the agent about the decision of the action at a particular state in the environment. A good optimal policy can help agent to reach goal. At the default state, the agent does not hold any good policies to achieve goal optimally. But, the agent will learn iteratively selecting the actions at each state, finds the best acts among them, which is policy-based reinforcement learning. The network that adopts policy in learning iteratively and supports the decision making of deep learning network is a policy network. A policy network for all applications works with stochastic actions where a collection of actions are obtained by experience and models that are employed during the decision making of a classifier. A representative model is built with an action sampling, which is performed during sentence wise document classification stating the categories of the sentences as an agent and the actions into policy of decision making. All the collection of decided actions is considered into a structured representation of a sentence classifier.



Locations Mapped as States

Location Code	State
Loc1	0
Loc2	1
Loc3	2
Loc4	3
Loc5	4
Loc6	5

Rewards Table

	Loc1	Loc2	Loc3	Loc4	Loc5	Loc6
Loc1	0	1	0	0	0	0
Loc2	1	0	1	0	0	0
Loc3	0	1	0	1	0	0
Loc4	0	0	1	0	1	0
Loc5	0	0	0	1	0	1
Loc6	0	0	0	0	1	0

A Sample Environment

Many problems in artificial intelligence require adaptively making a sequence of decisions with uncertain outcomes under partial observability. Solving such stochastic optimization problems is a fundamental but notoriously difficult challenge. In this paper, we introduce the concept of adaptive submodularity, generalizing submodular set functions to adaptive policies. We prove that if a problem satisfies this property, a simple adaptive greedy algorithm is guaranteed to be

Figure 1: A scenario of policies that are applied to select actions based on the rewards in the environment.

A. *Optimal Policy*

If the yield of the policy entails good quality and proposes good directions for the agent to achieve the said tasks of classification with best possible accuracy, then the policy is said to be optimal. A deterministic policy defines one possible action for a given state, when the agent is set to single given state and restricted to single particular action. A stochastic policy defines action space as probability distributions of actions for a given state, where the agent has scope of large choices of actions. Comparative to the deterministic policy the stochastic policy has possibilities of developing optimal policies.

B. *Methods*

The novelty in the long document classification that is chosen in this application is to define the categories based on the sentence-sense classification than word based classification which were used in the erstwhile research. The document classification undoubtedly requires NLP and Vectorization of the document into the required size of chunks, the process of sentence-sense based classification has defined a sentence vectors which defines weights for the sense of the sentences. The sense of the sentence in the document is a weight that is measured with the domain knowledge and the measure of importance with respect to the vocabulary.

1. *Sentence Vector*

Several word embeddings are believed to be true in classifying the document [4]. As words cannot be relevant in all contexts, except they are perceived with dictionary meaning and related transitives, a word in a sentence will have the exact meaning in the context. Therefore, a sentence vectorization is needed, which is composed of word embeddings which are more relevant to the context of document [5]. Thus preparation of sentence-embeddings are very purposeful while document classification using sentence-sense.

IV. Methodology

A. *Proposed Framework*

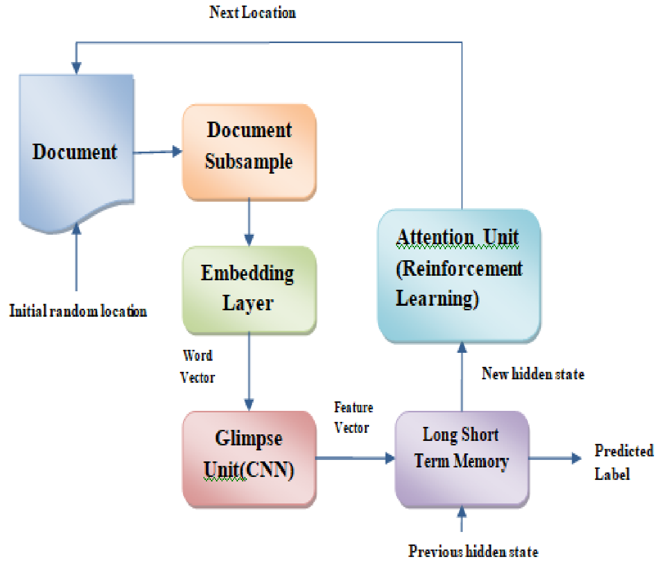


Figure 2: The framework for the proposed system.

Glimpse Unit:

At each time step, the glimpse unit observes a set of words by the given location loct. It encodes the words into a feature vector and gives this as input to Long Short Term Memory (LSTM) unit.

LSTM unit:

At each time step, the LSTM unit takes feature vector generated by glimpse unit as input, then combines it with the previous hidden state to produce the next state.

Attention unit:

At each time step, the attention unit predicts which part of the document should be looked at next which employs Policy Network.

B. *Algorithm*

The Algorithm for the proposed system is given below:

Input: Document D

Output: The probability of predicting category for document D

```
1. Initialize the module parameters [$\theta_{gm}, \theta_{rm}, \theta_{am}$] [$\theta_{gm}$ for glimpse module, $\theta_{rm}$ for recurrent module, $\theta_{am}$ for attention module], no._of_ glimpse G and maximum iterations Max.
2. for i=0,1,2,…Max do
3. for j=0,1,2,…G do
     a. Derive words around the location $loc_t$ and use word embedding to obtain the word vectors.
     b. Use the glimpse module $f_{gm}(\theta_{gm})$ which is basically CNN, to extract the feature vector $o_g$.
     c. Input $o_g$ to the recurrent module $f_{rm}(\theta_{rm})$ to obtain a new hidden state $h_t$.
     d. Predict the next location $loc_{t+1}$ using attention module with reinforcement learning using $h_t$.
4. End for.
5. Obtain the representation $S_i$ of glimpse with the input $o_g$ in the recurrent module.
6. End for.
7. Obtain the representation d of document D with the input $S_i$.
8. Employ the softmax classifier on document representation d.
9. If predicted label is correct get reward=1 otherwise get none reward.
```

V. Experiment

The collections of long documents are considered from legal and crime records. The records pertaining to legal and crime contexts are not in one single chunk, rather many number of documents are collected into a vault. Comprehension in the long documents is implemented by considering the word embedding in the first stage. The inter-related word embeddings are collected and concatenated into sentence vectors into all the possible sentence-sense vectors. The mere sentence vectors does not realize the sense of the context, such of those sentence vectors are weighted with the context relevance and are selected as sentence-sense vectors. Each sentence sense vector is assumed as the classifier and the contents of the documents are categorized by the classifiers as they belong to the sentence-sense. The total experiment is carried out in two phases; viz., sentence classification and document classification as shown in figure 3.

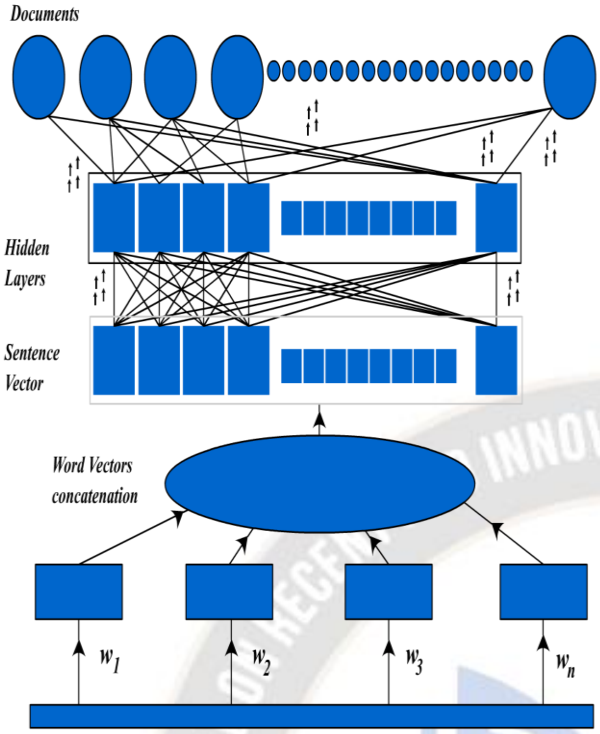


Figure 3: Building sentence-sense vector from word embedding.

A. *Sentence Classification*

Sentence classification is the fundamental job in the long document classification. The word embeddings are chosen for categorization of the fragments / samples of the documents [9]. Such word embeddings which participate in categorizing the fragments of documents are collected into meaningful groups and word-vectors are evolved. The word-vectors are the collection of word embeddings that are found with weighted frequencies in the fragments of the documents [10].

All the vectors with word-embedding are not considered for sentence classification [6]. A group of words from the corpus of the documents is prepared as the training data sets, the vectors that contain the words from the trained data sets are considered for the building of sentences. Thus word-vectors with the context relevance are selected as sentences. In general, the document classification is also achieved with word classification using word vectors.

The granularity of the classifiers is more and indistinguishable, for the cause of better comprehension, document classification is considered for sentence level classification. The objective of the sentence classification is to determine the fragments / samples of the document that evolve into the categories of classifiers, that are built with meaningful words or the words of the word-embeddings as word vector belonging to the context.

B. *Document Classification*

ocument classification is the second phase in the experiment. For classifying the documents that come with meaningful context or the central subject of interest, the sentence with sense, i.e., the sentences which possess meanings and words related to the context are selected [7]. As narrated in the sentence classification, building of sentence-sense vectors, the document is classified into various categories of comprehensions related to meaningful sentence-sense vectors. The total set of fragments / samples of documents are iteratively applied to detect the sentence-sense during classification.

PolicyNets play a very important role in developing the vectors. The word vectors are fundamentally developed from the preprocessed words of the documents and the relevant word-embeddings are stuffed into the copies of the documents, stating that the documents possess the relevant word related to the context. But this is not the complete document classification [8].

A convolutional neural network is employed to summarize the weights of the word-vectors and further into sentence vectors, which are later decided to ne as the important features of the context in the document by the PolicyNet. The PolicyNet is employed as a Recurrent Neural Network, where the collections of the word-vectors are mapped to the fragments / samples of the documents and thus finding the relevance with weights for the word-vectors. The weighter-word vectors thus acts a sentence. A group of sentences that pertain to the context are revealed with the similarities of weights. The agent in the PolicyNet determines the word-vectors with weights to ascertain them into a group as sentence.

Similarly, PolicyNet is applied at the sentence level, considering the groups of word-vectors as a sentence-vector. The sentence-vector appears to be a multidimensional vector of words with weights assigned rows-wise. The relevance of weights given to the sentence-vector is computed with the Sigmoid function in a typical convolutional neural network. The CNN evolves with the weighted sentence-vectors as context relevant sentences called sentence-sense vectors, where the fragments / samples of the documents are classified.

C. *Optimization*

The implementation of the Recurrent Neural Network and the Convolution Neural Network for the PolicyNet, optimizes the selection of words and further into sentence of the context of interest.

The PolicyNet iterates on the word embeddings based on the experience and the current word-embeddings in the preparation of context relevant word-vectors. The next level implementation of PolicyNet optimizes the word embeddings relevant to the context to build the sentence vectors. The weights are assessed for each sentence in the PolicyNet to estimate the relevant sentences of the context and a sentence-sense vector is prepared.

Therefore, during the iterations the optimal score for the word is brought out to be as the member of the sentence-sense vector and enabling the document classified. Figure 4 shows the flow in the framework.

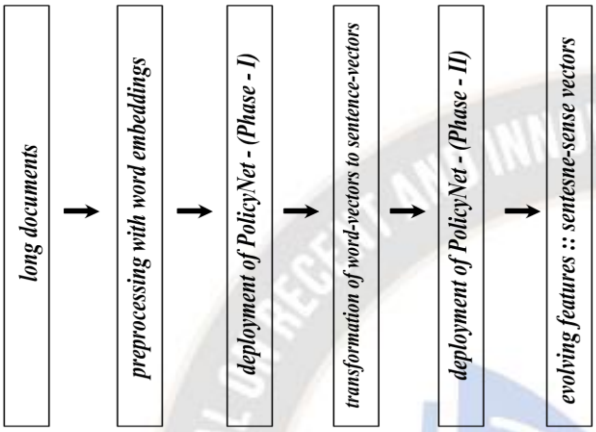

long documents
preprocessing with word embeddings
deployment of PolicyNet - (Phase - I)
transformation of word-vectors to sentence-vectors
deployment of PolicyNet - (Phase - II)
evolving features :: sentesne-sense vectors

Figure 4: Flow in the Framework

VI. Results

A. Experimental Setup

The setup of the experiment used in this article is setup by Python 3.10 on Windows 10. Deep learning framework Keras and TensorFlow are applied to build the relevant CNN. Adam optimizer is the most generally chosen optimizer for the convolution neural networks, which has been used in building the PolicyNet. The effectiveness of the model for classification with other network models has been verified. Comparison on several grounds has been considered to prove the efficiency of the proposed model.

B. Criteria for Scoring

The metrics for evaluation in this experiment for the BBC News Data sets (for selected criminal and legal texts) are accuracy, F1-score and the Macro-averaging. Macro-averaging has been done for several categories of fragments / samples of the documents. The arithmetic mean of the F1 values of all the categories is considered for the Macro-averaging.

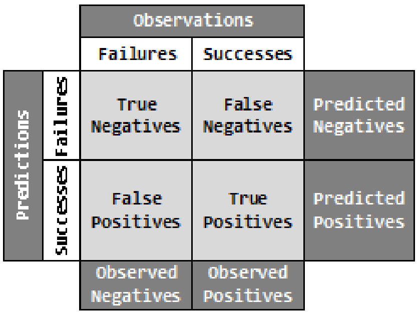


Figure 5: Confusion Matrix for Metrics of Performance Evaluation.

Accuracy is the metric, indicating the performance of the classification task, that the number of correctly classified fragments / samples of the documents as the percentage of the total samples.

$$Accuracy = \frac{TP + FN}{TP + FP + TN + FN}$$

Precision is the metric, indicating the performance of the classification task, that the number of correctly classified positive fragments / samples of the document as a percentage of the number of fragments / samples of the documents judged as they are context relevant (i.e, positive).

$$Precision = \frac{TP}{TP + FP}$$

Recall is the performance metric in the classification task that denotes the number of correctly classified fragments / samples of the document as the percentage of the true number of positive fragments / samples of the document.

$$Recall = \frac{TP}{TP + FN}$$

F1-score is the key measurement for the estimation of the performance of the classification task, where the combined evaluation of the accuracy and recall are computed.

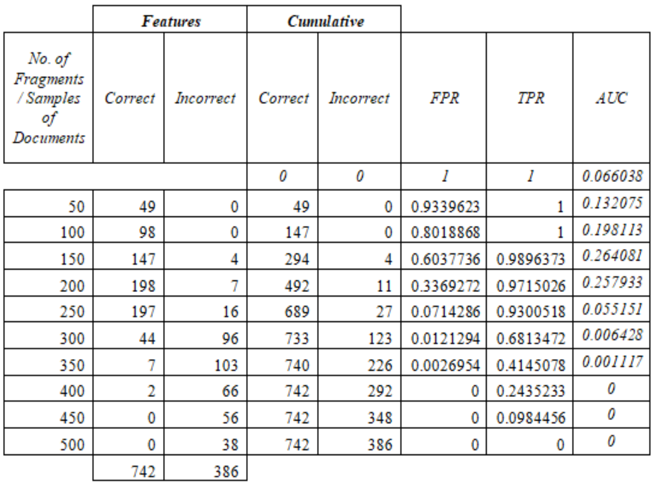
$$F1 = \frac{2 \times Precision \times Recall}{Precision \pm Recall}$$

Macro-averaging in the classification task represents the special metric called indicator, which is an average of F1 scores.

$$MacroAverage(F1) = \sum_{i=1}^{n} F_n$$

TABLE 2: The following shows the number of word-vectors found in the fragments / samples of the documents that are relevant to the context, and thus estimate the performance of the PolicyNet used in the Phase I for conversion of the word-vectors into sentence-vectors.

DOI: https://doi.org/10.17762/ijritcc.v10i12.5866
***Article Received**: 08 October 2022 **Revised**: 18 November 2022 **Accepted**: 04 December 2022*



	Features		Cumulative				
No. of Fragments / Samples of Documents	Correct	Incorrect	Correct	Incorrect	FPR	TPR	AUC
			0	0	1	1	0.066038
50	49	0	49	0	0.9339623	1	0.132075
100	98	0	147	0	0.8018868	1	0.198113
150	147	4	294	4	0.6037736	0.9896373	0.264081
200	198	7	492	11	0.3369272	0.9715026	0.257933
250	197	16	689	27	0.0714286	0.9300518	0.055151
300	44	96	733	123	0.0121294	0.6813472	0.006428
350	7	103	740	226	0.0026954	0.4145078	0.001117
400	2	66	742	292	0	0.2435233	0
450	0	56	742	348	0	0.0984456	0
500	0	38	742	386	0	0	0
	742	386					

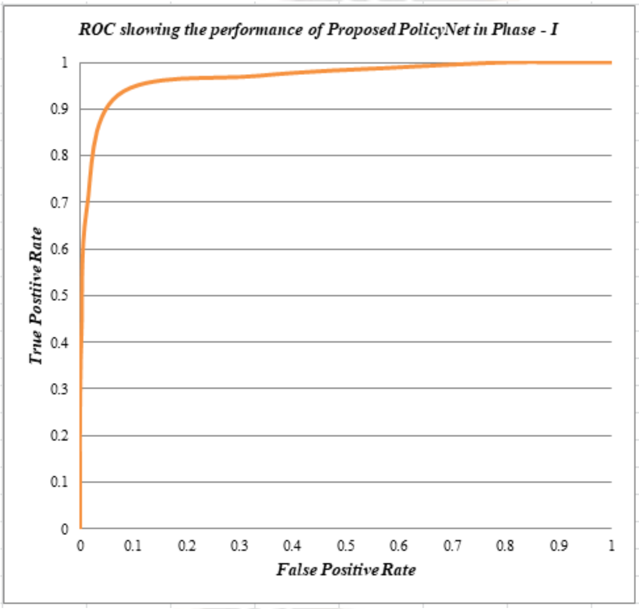
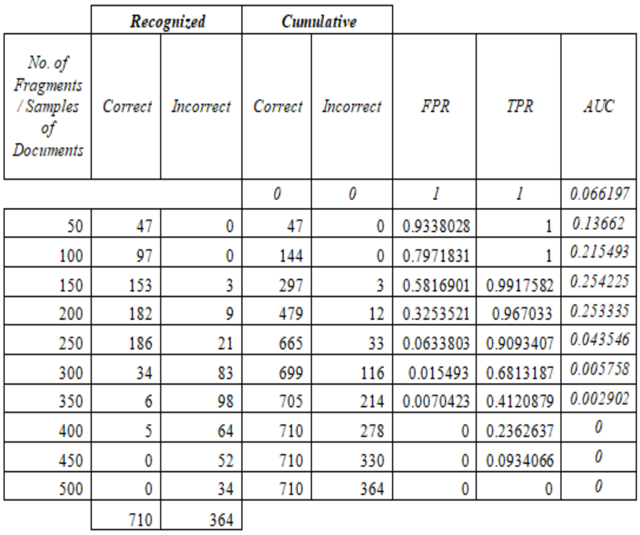


Figure 6: ROC curve showing the AUC as sum of 0.98 and the accuracy of 92.90%.

The repeater-operating characteristic graph is drawn for the measurements of the performance indicators observed from the experiment and it has been proved at AUC sum of 0.98 and accuracy of 92.9 percent, leading to the satisfactory accuracy levels compared to the previous works.

TABLE 3: The following shows the number of sentence-vectors found in the fragments / samples of the documents that are relevant to the context, and thus estimate the performance of the PolicyNet used in the Phase II for feature extraction sentence-vectors into sentence-sense-vectors.



	Recognized		Cumulative				
No. of Fragments / Samples of Documents	Correct	Incorrect	Correct	Incorrect	FPR	TPR	AUC
			0	0	1	1	0.066197
50	47	0	47	0	0.9338028	1	0.13662
100	97	0	144	0	0.7971831	1	0.215493
150	153	3	297	3	0.5816901	0.9917582	0.254225
200	182	9	479	12	0.3253521	0.967033	0.253335
250	186	21	665	33	0.0633803	0.9093407	0.043546
300	34	83	699	116	0.015493	0.6813187	0.005758
350	6	98	705	214	0.0070423	0.4120879	0.002902
400	5	64	710	278	0	0.2362637	0
450	0	52	710	330	0	0.0934066	0
500	0	34	710	364	0	0	0
	710	364					

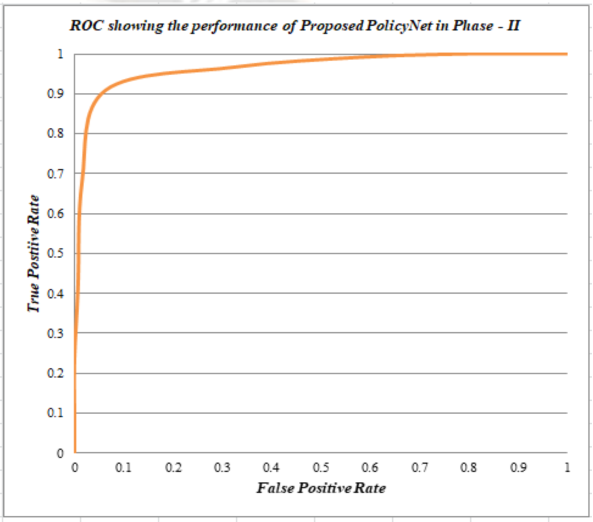


Figure 7: ROC curve showing the AUC as sum of 0.97 and the accuracy of 92.73%.

VII. DATASETS

Datasets used in the context of long document classification are usually belonging to news channels, sports, technology innovations, marketing and sentimental / opinion studies. To interpret the performance characteristics of the proposed framework specific data sets that project the propensity of processes are selected.

TABLE 4: Sources of Datasets

Source	Documents	Language
BBC	2225	English
WOS	46985	English

Documents from BBC News Websites from Kaggle website are available in five different categories technology, business, sports, politics and entertainment. WOS-46985 is the Web of Science Documents collection which contains

***Article Received**: 08 October 2022 **Revised**: 18 November 2022 **Accepted**: 04 December 2022*

46985 documents with 134 categories and 7 parent categories. These two datasets support document categorization and text mining perfectly compared to other data sets where they are limited to information retrieval and word level classification.

Text classification is a foundational experiment for Document classification, where the typical word-embeddings are converted into sentence-sense vectors. The advent of vectorization of the words from word-embeddings has turned the entire document classification sensible to the context of interests. Document classification ascertains the categories of documents classified based on the sentence-sense vectors that are very helpful to the document analysts and experts to categorize label and evolve with opinion analyses. Finally document classification supports quantification and characterization of documents into classes in the context of interest as a spectrum of seriousness of various categories of legal and crime contexts.

VIII. Discussion

For the experiment on documents data sets from BBC New Website (legal and criminal records), maximum number of features is considered as 6000, word embedding categories are considered as 128 from NLTK corpus. Stop words and Lemmatization are applied from WordNet API for the support of preprocessing in the long document data sets. From TensorFlow, the APIs for Keras, Tokenizer and Sequence padding for preprocessing, LSTM, Convolution1D layers are drawn. Keras has an ideal set of initializers, regularizers, constraints, optimizers and ability for defining custom layers. In order to be uniform in the datasets, all the documents are converted to lowercase after filtering stop words, punctuations and then tokenization has been done.

The purpose of this process is to generate smoothly non-ambiguous encodings among the words to avoid misclassification problems. Sentences are certainly of variable length consisting of words which are connected in a non-uniform sequences, which is stated as logarithmic dependencies among the sentences and words.

However during the process of generating candidate input sequences for the network, the uniform length shall be maintained, where padding is used for such vectors which do have limited length. The max length of the sentence size is predefined in the framework. Mean of the lengths for the word vectors is considered and round to nearest integer to determine the max length of the vectors.

IX. Conclusion

The key contributions of the article are developing PolicyNets for the important two phases of document classification. Document classification is never been an plane sweep approach for any methods of machine or non-machine learning approaches. The quality of classification is the ascertain the document classified into the right contexts of interests. The challenging portion of the classification in preprocessing is overcoming the length of the long documents, by means of identifying the word vectors using word embeddings with agent based optimization using PolicyNet iteratively. Converting the word-vectors into reasonable set of sentence vectors and further optimizing using second phase of PolicyNet, into sentence-sense vectors, these achievements have accomplished the assumed objectives of document classification with simplified structures rather directly working on long length documents.

References

[1]. Choi, Gihyeon, Shinhyeok Oh, and Harksoo Kim. “Improving document-level sentiment classification using importance of sentences.” Entropy 22, no. 12 (2020): 1336.

[2]. Park, Hyunji Hayley, Yogarshi Vyas, and Kashif Shah. “Efficient Classification of Long Documents Using Transformers.” arXiv preprint arXiv:2203.11258 (2022).

[3]. Khoo, Anthony, Yuval Marom, and David Albrecht. “Experiments with sentence classification.” In Proceedings of the Australasian Language Technology Workshop 2006, pp. 18-25. 2006.

[4]. Nikolaidou, Konstantina, Mathias Seuret, Hamam Mokayed, and Marcus Liwicki. “*A Survey of Historical Document Image Datasets.”* arXiv preprint arXiv:2203.08504 (2022)

[5]. Kišš, Martin, Jan Kohút, Karel Beneš, and Michal Hradiš. “Importance of Textlines in Historical Document Classification.” In International Workshop on Document Analysis Systems, pp. 158-170. Springer, Cham, 2022.

[6]. Jiang, Shuo, Jie Hu, Christopher L. Magee, and Jianxi Luo. “Deep learning for technical document classification.” IEEE Transactions on Engineering Management (2022).

[7]. Noguti, Mariana Y., Eduardo Vellasques, and Luiz S. Oliveira. “Legal document classification: An application to law area prediction of petitions to public prosecution service.” In 2020 International Joint Conference on Neural Networks (IJCNN), pp. 1-8. IEEE, 2020.

[8]. Wan, Lulu, George Papageorgiou, Michael Seddon, and Mirko Bernardoni. “Long-length legal document classification.” arXiv preprint arXiv:1912.06905 (2019).

[9]. Hassanzadeh, Hamed, Mahnoosh Kholghi, Anthony Nguyen, and Kevin Chu. “Clinical document classification using labeled and unlabeled data across hospitals.” In AMIA annual symposium proceedings, vol. 2018, p. 545. American Medical Informatics Association, 2018.

[10]. Stein, Roger Alan, Patricia A. Jaques, and Joao Francisco Valiati. “An analysis of hierarchical text classification using word embeddings.” Information Sciences 471 (2019): 216-232.

[11]. Wagh, Vedangi, Snehal Khandve, Isha Joshi, Apurva Wani, Geetanjali Kale, and Raviraj Joshi. “Comparative study of long document classification.” In TENCON 2021-2021 IEEE Region 10 Conference (TENCON), pp. 732-737. IEEE, 2021