Convolutional Neural Network with Word Embedding Based Approach for Resume Classification

Shabna Nasser
Calpine Labs
UVJ Technologies, Kochi
Govt Engg. College,Palakkad India
shabnanasser@gmail.com

Sreejith C
Calpine Labs
UVJ Technologies, Kochi
Govt Engg. College,Palakkad India
sreejith.cherikkallil@calpinetech.com

Irshad M
Calpine Labs
UVJ Technologies, Kochi
Govt Engg. College,Palakkad,India
idofirshad@gmail.com

***Abstract*—Document Classification is a very prominent area, it isapplicableforadiversityofnovelapplications.Inthisstudy,we focussed on classifying resumes to different classes. The proposed approach is for classifying resumes using Convolutional Neural Network with Glove-Word Embedding. We have segmented resumes into various levels and a hierarchy of classification levels is created. For each level, the CNN with word embedding model is used for classification. The output of each classifier is later combined to define the overall hierarchy of the resume category. Results are evaluated using the performance measures such as precision, recall, and f-score. The results obtained are promising and the proposed system is helpful in the recruitment and selection process of candidates.**

Keywords—CNN, Deep Learning, Glove, Neural Network, Word Embedding.

I. Introduction

Document classification is a relevant area in recognition of patterns, web search, library science and information science. It has a fundamental part in computer science as the electronic contents in the form of documents are rapidly increasing day by day. It is applicable in various natural language processing applications and other computational applications, which includes language identification, spam filtering, sentiment analysis, text simplification, information retrieval, ranking etc. Classifying documents into categories is nothing but allowing the context of documents or text to one or more particular categories. There are a variety of approaches for performing document classification. The task of document categorization is quite challenging when using high dimensional unstructured data like resumes.

Majority of the traditional methodologies of classifying thetext represents the classification of the document, used certainlexical features like bag of words, kernel, linear models, n-gram models and the supervised machine learning classifierslike Logistic Regression, Support Vector Machine, and RegularExpression. Also, linear classifiers are pretended as strongbaseline models for such text categorization tasks [1] [2] [3].The process of document classification does have a vitalrole for feature extraction. There are several approaches that have used the features such as, TF-IDF (Term Frequency-Inverse Document Frequency), Centroid method and WMD (Word Mover's Distance) [4] based on word2vec, and Latent Semantic Indexing. This paper, make use of the convolutional neural network with words embedded using pretrained Glove embedding. Here we evaluate the efficiencies in the performance of each of the classification model using certain effective measures such as precision, recall & f-measure. The most recent approaches used for classifying text documents are with deep learning methods like long short-term memory (LSTM) models based on recurrent neural network (RNN) and convolutional neural network (CNN) [5] for learning representation of text. This paper identifies how far the effectivenesss of CNN-Glove word embedding model helps in the classification task. The goal is to classify the resume files in various job classes particularly in the technical and nontechnical domain.

Proposed approach center around document classification for resumes furthermore, however, utilizes resume dataset and job description dataset for training. Also, this work is useful for reducing the human workload and save the time of manually classifying the resumes during recruitment and selection of candidates to various jobs. Most of the big organizations deal with thousands of resumes during their recruitment process. Hence, some effective methodologyis needed to classify this huge amount of resumes into various categories.There sumes are in different file formats like.pdf, .docx, .doc, .odt, etc. and this again may be a crucial issue. The conversion of these format types into text format is also a complextask.

The paper is arranged as follows in the upcoming sections: In Section II, we discuss few works related to the classification of text documents and algorithms used. Then Section III, illustrates the system design and working, and describes resumes and job descriptions datasets. Experimental setup and results are given in section IV. In section V, we provide the conclusion and future research directions.

978-1-5386-5744-7/18-$31.00©2018IEEE

II. RelatedWorks

Classification of text documents into certain categories or classes have been recommended by various researchers using several classification algorithms, more generally in text categorization and sentence classification tasks. But there is not much work available on resume domain. Now, let us check out a few of the works since many efficient approaches were proposed in the field oftext document classification.

As a superior to Naive Bayes and Neural Network approaches, Conditional Random Field (CRF) based model isput forth by [6]. In order to mark the various segments of the documents as signature noise/ handwriting or machine print, the ideology used by [6] was a new one using CRFs. They gave a probabilistic model of CRF and the parameters were estimated by conjugate gradient descent. Ref. [6] solve the problem by making segments of the document into several patches and each segment is labeled as either handwriting/ noise or machine print. Also used a six-step process to label documents that are scanned. State and transition features were extracted and evaluated using accuracy, recall, and precision. This work obtained about 95.75% of overall accuracy. Evaluated that the CRF based model which use neighboring patches with the spatial interdependencies gives greater performance. For the evaluation of the model, the Tobacco industrial litigation archives wereused.

The distributions of word vectors or document classes have been proposed in [7], this paper uses the naive approach keeping in mind the end goal to bring up the representation of documents with an average pooling. Demonstration of documents in the semantic space is additionally investigated, and analyses that, instead of the pooled centroid method for a class or a document, the word vectors distribution is more suitable. According to this approach by [7], it proposes two models, first, the word vectors of a document class is representedusing ClassSpecificGaussianMixtureModel(CSGMM).Second,in the word vector space, the document is denoted as a posterior probability above the components of global GMM, which is the Semantic Space Allocation (SSA) model. Word vector is moreover called as the continuous and dense representation of the word. Though the words and their relationship between them are embedded as word vectors, which paves an easy way to combine the meanings for documents and paragraphs. A Gaussian mixture distribution is used by the document class word vectors, which is designed with CSGMM. This methodology used sohu text database by sohu research center. In order to produce the word vectors they used, word2vec tool and, for training SVM and kNN models uses scikit-learn toolkit and weka toolkit respectively. Ref. [7] investigate that SSA have higher performance than w2v pooling and,CSGMM is less efficient than LDA and w2vmodel.

Adistancefunctiontocalculatethedistanceorthesimilarity measure between the text documents is offered by [4], named as Word Mover's Distance (WMD). It gives the dissimilarity measure in between words, in two similar documents as the largeramountofdistance.ThatisWordMover'sDistanceis a measure that gives higher similarity score for the texts in similar documents and with low distance. They are actually inversely proportional to each other.If the text of two documents has larger dissimilarities then do have larger distance. WMDis a strategy that enables us to evaluate the distance between two documents genuinely, even though there are no words in like manner but by coordinating the significant words. It can precisely quantify the similarity or dissimilarity between the two sentences. The deployed approach by [4] is compared against seven baselines of document representation including, bag- of-words, Latent Dirichlet Allocation, Componential Count- ing Grid, Latent Semantic Indexing, BM25 Okapi, TF-IDF, Marginalized Stacked Denoising Autoencoder (mSDA).WMD is free from hyper parameter, that provides higher accuracy in retrieval and also combines the understandings encoded in word2vec. It is efficiently interpretable as well as outperforms all alternate SOA (state-of-art) documentdistances.

Ref. [8] proposes a TF with stemmer based feature extraction algorithm for document categorization. This paper uses J48: DT algorithm and stemming algorithm for the accuracy of feature extraction on Reuters Dataset that produces an accuracy of 98.5 percent. Ref. [9] put forward Character level convolutional networks for text classification. This suggested method does have the convolutional modules with temporal features, that helps to compute the 1-D convolution. Deeper models are trained using a major 1-D version of the max-pooling module, named as, the temporal max-pooling. It helps to train the deeper convolutional networks more than 6 layers and all other networks other than this fails. The approaches discussed in [9] is implemented using the torch and uses stochastic gradient descent algorithm. Also, a thresholding function is used that makes the convolutional network similar thatofrectifiedlinearunits(ReLUs).

A combination of Convolutional Neural Network (CNN) and Long Short-Term Memory models called C-LSTM was presented by [10]. It was a novel method for modeling sentences. LSTM takes the output of a one layer CNN. Each sentence window features are transformed than that of directly designing the LSTM from the sentences. As a prior to feed the neural network, the sequences based inputs are selected, than choosing the parse trees. The effectiveness of this approach is evaluated using 6-way classification and sentiment classification tasks. This approach helps in learning long range-dependencies by LSTM. As [10] take the merits of both LSTM and CNN, they uses CNN for extracting higher-level dependency features, while LSTM is used for grabbing long-term dependencies of word vectors. This paper shows that TREC, 6 way question type classification produces an accuracy of 94.6 percent while the others such as binary classificationgiveanaccuracyof87.8percentandfine-grained is about 49.2percent.

Hierarchical attention networks for document classification is proposed by [11]. This model performs significantly better than previous methods and effective in picking out important words and sentences. This method has two peculiar factor of importance. One, the model deploys a hierarchical structure

TABLE I
SUMMARY OF APPROACHES AND DATASETS USED.

Model	Approach	Dataset	Performance
Kusner et al. [4]	WMD	BBCSPORT, TWITTER, RECIPE, OHSUMED,CLASSIC, REUTERS,AMAZON & NEWS	-
Sayfullina et al. [5]	CNN	Job Description, Resume Summaries & Children's Job Description	74.88%, 40.15%, 51.02% respectively.
Shetty et al. [6]	CRF	Tobacco industrial litigation archives	95.75% of overall accuracy
Xing et al. [7]	CSGMM, SSA	Sohu text database	-
Vidhya et al. [8]	J48: DT algorithm	Reuters Dataset	98.5%
Zhang et al. [9]	Char. CNN	AGs News, Sogou News, DBPedia, Yelp Review Polarity, Yelp Review Full, Yahoo! Answers, Amazon Review Full & Amazon Review Polarity	-
Zhou et al. [10]	C-LSTM	SST-Movie reviews	-
Yang et al. [11]	HAN	Yelp reviews, IMDB reviews, Yahoo answers & Amazon reviews	Outperforms baselines.
Joulin et al. [12]	fastText	AG, Sogou, DBP, Yelp P., Yelp F., Yah. A., Amz. F. & Amz. P.	Sogou goes up to 97.1%
Johnson & Zhang [13]	LSTM	IMDB, Elec, RCV1 & 20-newsgroups	-
Conneau et al. [14]	VDCNN	AG, Sogou, DBP, Yelp P., Yelp F., Yah. A., Amz. F. & Amz. P.	-
Ranjan et al. [15]	LSTM	20 newsgroups dataset	-

for anticipating documents' hierarchical structure. Two, uses attention mechanisms of two levels to apply it on the sentence and word level. Also, extraction of features plays a key role in document classification. It is used to document categorization for the betterment of efficiency, scalability, and accuracy. Feature extraction acquires an important subset of features from a dataset for improving the classification task. The objective of extracting features is to avoid the unimportant ones and lessen the dimensionality and so that, the performance and efficiency of classification algorithms are enhanced. Ref. [11] used six large scale datasets (Yelp reviews- Yelp dataset challenge in 2013, 2014 & 2015, IMDB reviews, Yahoo answers, Amazon reviews) for training the models and for other evaluation purposes.

Ref. [12] investigates a simple and efficient baseline fortext classification called as Fast Text classifier. It shows that fastText works well and faster during training and testing purposes as well as best in accuracy measures. Two different processes are used to evaluate fastText: First, collate it with sentimental analysis problem by using the text classifiers proposed previously. Secondly, make use of a tag prediction dataset in order to evaluate its efficacy to extend to large output space. Another concept in text categorization for region embeddings was put forth by [13] using LSTM. This paper explores that it produces good results when region embeddings are combined as convolutional layers and LSTM, on any unlabeled data. For training a linear model and feature generator they made a framework that is more general. It is created on top of 'region embedding + pooling', a general framework and, gives out an enlightened region embedding through LSTM. Either any of the supervised LSTM or semi-supervised LSTM, simplifying the model as much as possible is the major strategy. Considers region embeddings through one-hot LSTM and investigated that it outperforms SOA one-hot CNN. Also, according to [13], by working directly with one-hot vectors on unlabeled data, the region embeddings can belearned.

A Convolutional neural network model is deployed in [5] to classify resumes. In this, words present in a sentence are embedded using Word2Vec, then these concatenated word vectorsformamatrix,thisisfedtoCNN.Itshowsthat this model outperforms fastText classifier introduced in [12], which is also used for classifying text data more efficiently. This model is based on learning word embeddings, averages them and the resulting vector is fed to a linear classifier. The fastText approach is very faster in training and testing without accessing GPU's. While, when moving from source totargetdomaintheproposedmethodologyusingCNNby [5] drops in the accuracy from 74.88 percent to 40.15 percent, also underperform LSTM-GRNN models (by 1 percent). This approach uses resume dataset for testing, children's dream job for validation, and job descriptions datasets for training the models.

Ref. [14] proposes a novel approach for text processing system that makes use of small convolutions and pooling operations, and works directly at the character level. Also, this approach uses 29 convolution layers and though is the Very Deep Convolutional Neural Network (VDCNN). As the depth increases the performance of the architecture does also increase. Ref. [14] is the first to bring out VDCNN for the processing of texts. It uses Stochastic Gradient Descent for training the model. This model is based on two principles of design: (1) operates on characters, (2) to learn the hierarchical high-level representation of the sentence, uses max pooling andconvolutions.ThisisimplementedonasingleNVidiaK40 GPU system using Torch 7. Freely available eight large scale datasets are used for evaluation purposes. Also, outperform state-of-the-art ConvNets, and compared to other pooling methods max-pooling gives betterperformance.

According to [15], LSTM is good for learning the long-term dependencies and it is helpful in document classification. It is one of the recurrent neural network variation. An LSTM neural network has 3 gates: forget gate, input and output gates, each neuron possess a memory cell and these 3 gates. These gates are used for passing or blocking information to and from the memory cell. Found that this approach of using LSTM for document classification is successful while comparing withthe context. LSTM is also helpful in handling complexities as it can store information for a long period of time. In certain cases, the layers may vanish or blow up when the errors are backpropagated through time, in such situations LSTM is made useful. TABLEI demonstrate the summary of various

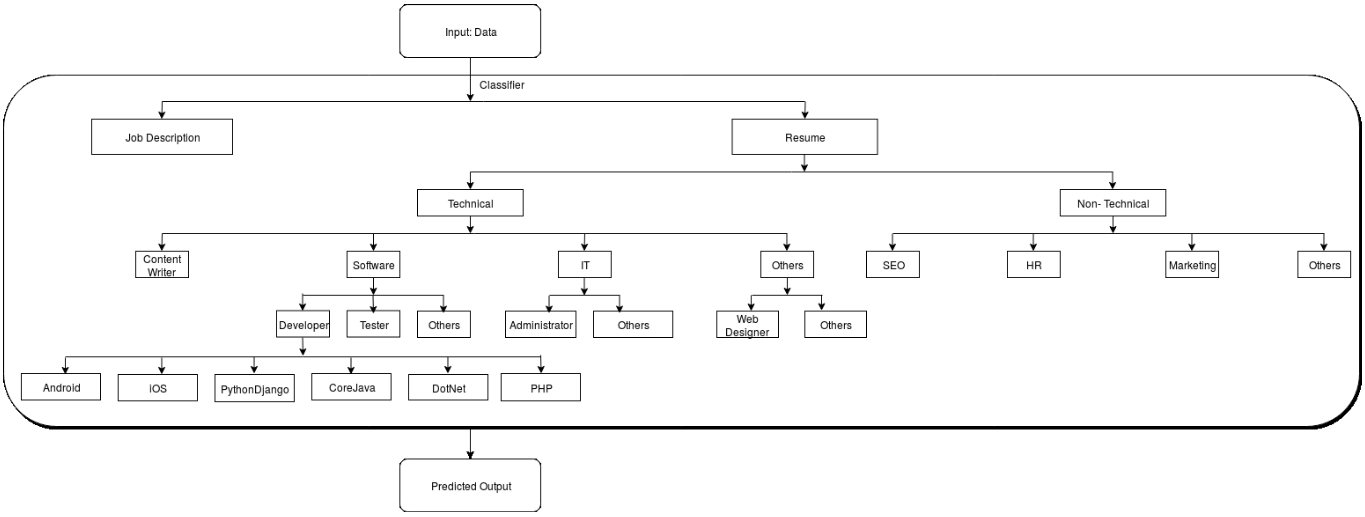


Fig. 1. Classification Hierarchy.

approaches and datasets used for text document classification.

III. System Design

The proposed system have four major modules such as:

1) *Extracting Plain Text*: This is the first step that extracts the contents from the resumes and job description file samples which are in .doc, .docx, .odt, .pdf or .rtf file formats,byusingApacheTikaserver[16].
2) *Preprocessing*: This step includes cleaning and tokenizing of the input files. Cleaning the documentby removing multiple whitespaces, stopwords, punctuations usingstringreplacementandregularexpressions.
3) *Feature representation*: After filtering the sample files through preprocessing, represents the word feature vectors using the pretrained Glove-100 dimensional word vectors[17].
4) *Classification*: Feeding the embedded words as vectors to the deep learning network CNN for the classification purpose. It then classifies and predicts the desired categoriestowhichtheinputfilebelongsto.

The following subsections describe the datasets and the detailed architecture of the proposed system. The system design in Fig. 2. shows the different modules.

A. Datasets

We have two datasets, resumes dataset and job descriptions dataset. There are not much publically available datasets for resumes. We have utilized some of the resumes files from Calpine Lab's resume collection. It contains above 2000 resumes in different file types like .doc, .pdf, .docx, .rtf, etc. Where job description dataset contains description regarding various jobs. We have created a job description sample set [18] of different job categories, and some of the samples of resumes were collected from [19] for training our models as demonstrated in TABLE II.

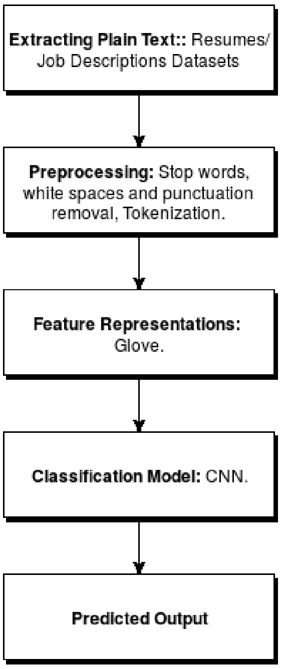

Extracting Plain Text:: Resumes/ Job Descriptions Datasets
Preprocessing: Stop words, white spaces and punctuation removal, Tokenization.
Feature Representations: Glove.
Classification Model: CNN.
Predicted Output

Fig. 2. Design of the Proposed System.

B. SystemArchitecture

This work proposes a document classification approach for classifying resumes and job description files belonging to different categories in the technical and nontechnical domain. Our proposed method classify the resume in these specified domains into more specific categories. The technical and nontechnical areas are again further classified. In the technical class we had, Software, IT, Content Writer and Others, in the nontechnical we had various categories like SEO,HR, Marketing, and Others, which will be again classifiedinto furthermorespecificcategoriesasdepictedinClassification

TABLE II
DISTRIBUTION OF DATASETS AT EACH LEVEL
OF THE HIERARCHY

Level	Category	Dataset
Level 1	Job description & Resumes.	500 each
Level 2	Tech & Nontech.	500 each
Level 3:		
3.1	Tech: Software, Testing, CW & Others.	250each
3.2	Nontech: SEO, HR, Marketing & Others.	250each
Level 4:		
4.1	Software: Developer, Tester & Others.	250each
4.2	IT: Administration & Others.	250each
4.3	Others: Web designer & Others.	250each
Level 5	Developer: Android, iOS, PythonDjango, CoreJava, DotNet, PHP.	250 each

Hierarchy illustrated in Fig. 1.

The Classification Hierarchy shown in Fig. 1 describes 5 levels of classification. The system processes the input data and then feed them to the classifiers so that the input data is first classified as either resume or job description. If the input data is a resume, then classifying it to either technical or nontechnical, and if technical then put it into a certaincategory and then if it belongs to any other specific class is alsochecked andthencategorizingittoamorespecificandprecisecategory. Similarly, if the data is nontechnical then it is classifiedfurther into a specific category. Total of eight classifiers is used. That is, the input data is fed to the required classifiers and get back the correctly predicted output, predicting to which categories the input file belongsto.

IV. EXPERIMENTSANDRESULTS

In this section, we describe the methods used to build the convolutional neural network model for the classification in each level of the hierarchy and the comparison of results procured.

We evaluate the word embedding based CNN model on job description files and resumes. First two levels in the hierarchy have 1000 files, distributed as 500 each for Job description,Resume,Tech,andNontech.Inthethirdleveleach category is constituted with 250 samples i.e; a total of 2000 samples. Similarly, in the fourth level, 1750 total sample is also distributed among the different categories with 250 each. While in level five, the total of 1500 samples distributed over six categories with 250 each. The distribution of datasets for each category at each level is shown in TABLE II. There are eight classification models in total for the system architecture. The third level is build using two CNN classification models and the fourth level uses three classification models, while all other levels required exactly one CNN classification model. Total of four binary classifiers and four multi-class classifiers areused.

The CNN model contains 8 layers with an input layer, an embedding layer, two 1D-Convolutional layers, one max-pooling layer, a global max-pooling layer, a dropout layer and adenselayer.Thefirstlayeristheembeddinglayer,which embeds the words using the pretrained Glove 100-dimensional wordvectors,whichcontainsabout400000wordsvectors.The output of this layer is given to the first 1D-Convolutional layer with window size 5 and the number of feature maps is 100 for each layer. The features from the first 1D-Convolutional layer are passed to the corresponding 1D-Max-pooling layer, which has the pool size of 5. Likewise, the second 1D-Convolutional layer features are fed to the 1D-GlobalMax-pooling layer. The downsampled features from this layer are given to a dropout layer with a dropout rate of 0.2 for regularization, hence, during training this layer dropouts 20 percent of the randomly selected neurons in order to prevent from overfitting. Thenthis final feature vectors from this layer are passed to the fully connected dense layer with size equal to the number of output predictions needed for each classifier. Fig. 3.demonstrates the 8 layered Convolutional Neural Network model for binary classifications.Thenumberoflayersmayexceeddepending on the type of classification, that is, for multi-class classifications the CNN model is built with 9 and 11 layers. The number of convolutional layers and corresponding pooling layers, and the number of dense layers are increased while moving to the multi-class classifier. The 11 layered CNN model is used for the classification on Level 5 while all other multi-class classifiersarewith9layeredCNN.

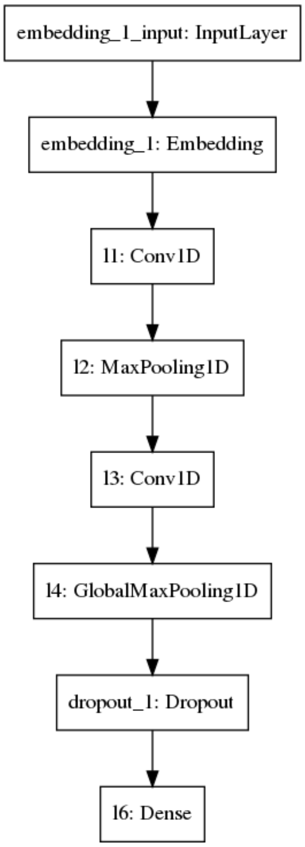


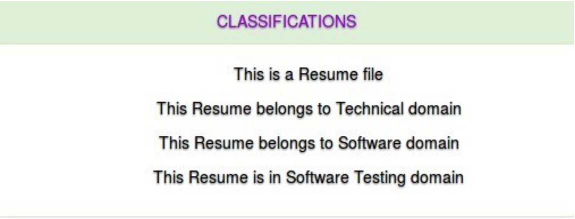
Fig. 3. CNN Architecture for Binary Classification.

Precision, recall, and f-score are considered as the measures

TABLE III
TRAIN AND TEST ACCURACIES OF EACH CLASSIFIER.

Level	Train(%)	Test(%)
Level 1	99	94
Level 2	93.7	88.5
Level 3:		
3.1	99	94.6
3.2	98	93
Level 4:		
4.1	97.9	90
4.2	90	87.9
4.3	99	96
Level 5	98.7	92.9

for evaluating the effectiveness of our system. We have taken 80 percent of the total sample for training and 20percent wereconsideredfortestingoneachclassifier.Theclassifiersat eachlevelanditscorrespondingtrainingandtestingaccuracies are demonstrated on TABLE III. The sample output of the proposedsystemisasshownbelow:



CLASSIFICATIONS

This is a Resume file

This Resume belongs to Technical domain

This Resume belongs to Software domain

This Resume is in Software Testing domain

Fig. 4. Output of the System for an Input Resume File.

V. CONCLUSION

As of now, resume categorization task is a very sparsearea in the document classification domain.Hence, classifyingresumes into various categories for easiness intherecruitmentandselectionprocessisarelevantareaofresearch. Wehavecorroboratedthattheconvolutionalneuralnetworkwithpre-trained Glove embedding based modelsreturnspromisingresultsinthistask. In this paper, we train the hierarchyofjobcategoriesusingbinaryandmulti-classclassificationCNN models with pretrained Glove embedding. Totalofeightclassificationmodelsof CNNwerebuiltandtheperformanceof each classifier is evaluated with certaineffectivemeasures.Forfutureresearch,wecanstretchthesizeofsampledatasets,addmoredataclassesandmakeuseofcustomwordembedding models with more deep neuralnetworks. Also,further improvements in the efficiency can be checked out using multi-label classification algorithms.

ACKNOWLEDGMENT

This work was fully funded by Calpine Labs, UVJ Technologies, Kochi, India. Also, this research was remarkably supported by Mr. Sreejith C, Data Scientist, Calpine Labs. We are also immensely grateful to Mr. Bijeesh Devassy, Project Manager,CalpineLabs,andMr.IrshadM,AssistantProfessor-CSE Department, Government Engineering College, Palakkad for sharing their pearls of wisdom with us during the course of this research. We also thank all of the colleagues from Calpine Labs, UVJ technologies, Kochi, India, who provided insight andexpertisethatgreatlyassistedtheresearch.

REFERENCES

[1] A. McCallum, K. Nigam, *et al.*, "A comparison of event models for naive bayes text classification," in *AAAI-98 workshop on learning for textcategorization*,vol.752,pp.41–48,Citeseer,1998.

[2] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in *European conference on machine learning*, pp. 137–142, Springer,1998.

[3] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "Liblinear: A library for large linear classification," *Journal of machine learningresearch*,vol.9,no.Aug,pp.1871–1874,2008.

[4] M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger, "From word embeddings to document distances," in *International Conference on Machine Learning*, pp. 957–966,2015.

[5] L. Sayfullina, E. Malmi, Y. Liao, and A. Jung, "Domain adaptation for resume classification using convolutional neural networks," in *International Conference on Analysis of Images, Social Networks and Texts*,pp. 82–93, Springer,2017.

[6] S. Shetty, H. Srinivasan, M. Beal, and S. Srihari, "Segmentation and labeling of documents using conditional random fields," in *Document Recognition and Retrieval XIV*, vol. 6500, p. 65000U, International SocietyforOpticsandPhotonics,2007.

[7] C. Xing, D. Wang, X. Zhang, and C. Liu, "Document classification with distributions of word vectors," in *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2014 Asia-Pacific*, pp. 1–5, IEEE,2014.

[8] S. Vidhya, D. A. A. G. Singh, and E. J. Leavline, "Feature extraction for documentclassification,"

[9] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," in *Advances in neural information processing systems*, pp. 649–657,2015.

[10] C. Zhou, C. Sun, Z. Liu, and F. Lau, "A c-lstm neural network for text classification," *arXiv preprint arXiv:1511.08630*,2015.

[11] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1480–1489,2016.

[12] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of tricks for efficient text classification," *arXiv preprint arXiv:1607.01759*,2016.

[13] R. Johnson and T. Zhang, "Supervised and semi-supervised text categorization using lstm for region embeddings," *arXiv preprint arXiv:1602.02373*,2016.

[14] A. Conneau, H. Schwenk, L. Barrault, and Y. Lecun, "Very deep convolutional networks for text classification," in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, vol. 1, pp. 1107–1116, 2017.

[15] M. N. M. Ranjan, Y. R. Ghorpade, G. R. Kanthale, A. R. Ghorpade, and A. S. Dubey, "Document classification using lstm neural network," *JournalofDataMiningandManagement*,vol.2,no.2,2017.

[16] https://wiki.apache.org/tika/TikaJAXRS,2018.

[17] J. Pennington, "Glove: Global vectors for word representation." https://nlp.stanford.edu/projects/glove/,2018.

[18] http://www.infopark.in/job-search.php,2018.

[19] https://in.linkedin.com/,2018.