



A Hybrid Deep Learning Approach for Sentiment Analysis using CNN and Improved SVM with Multi Objective Swarm Optimization for Domain Independent Datasets

Raviya K¹, Dr. Mary Vennila S²,

¹ Research Scholar, PG and Research Department of Computer Science, Presidency College, Chennai, India, raviya.mca@gmail.com

² Associate Professor and Head,

PG and Research Department of Computer Science, Presidency College, Chennai, India, vennilarhymend@yahoo.co.in

ABSTRACT

In the Big Data world, there is an exponential increase in the volume of numerous data, such like text, image, audio and video, as text is the largest among them. Sentiment analysis is a trendy application in text mining, where text data concerning the feelings or attitude of the consumer is collected using different methods or techniques. Sentiment detection of online product reviews is helpful to figure out emotions and viewpoints of customers. Many researchers have just developed the Deep Learning model for obtaining tremendous performance in NLP. This paper suggested novel deep-learning hybrid architecture that is highly effective for analyzing sentiments on domain independent datasets. We blend deep Convolutional Neural Networks (CNN) and support vector machines (SVM) for a better overall classification. The reason for using CNN is not only to extract local features but also the framework for predicting sentiments and combining CNN output into SVM progress the classification. In addition, we have adopted MSPSO (Multi-swarm Particle Swarm Optimization) system to obtain optimized feature selection and to train the SVM for further improved in sentiment classification. To demonstrate that our proposed method is of a generic nature, we have evaluated on datasets of online product reviews from various domain. Evaluation exposes that the implementation of the proposed approach is reliable among all the datasets and often outshine the state-of-art systems.

Key words: Convolutional Neural Network, Deep learning, Sentiment Analysis, Support Vector Machine.

1. INTRODUCTION

With the exponential advancement of information technology, the Internet has emerged as integral part of human's lives. Mostly people express their views about

various entities including products, places, services on various stages such as blogs, web-forums, social media review pages, etc. Such websites are composed of useful knowledge on a number of topics, as of trading issues to political problems and social awareness. It is hard to handle the analysis of this huge amount of data manually. The techniques of sentiment analysis have been recognized to be very useful in this case. In

particular, the classification of text has gained significant consideration as the text can have a categories of sentiment labels. The study of emotions includes various levels of mood (e.g., happiness, pleasure, excitement, anger), while the study of emotional analysis includes the identification of sentiment polarities namely positive, negative and neutral. Here, we emphasis on the analysis of sentiments which classifies the given text into one of the classes of sentiments. If that sentiment is correctly predicted, this will extend to different industries (e.g., film recommendation, customer support system, customized news feed etc.). Many websites (e.g., Netflix) are providing film recommendation services that effectively predict consumer opinion or rating scores. Recently, machine-learning based methods have gained growing interest. Typically, this technique uses phrases such as bi-grams, tri-gram, part-of-speech, sentiment words and knowledge acquisition as features to reflect the training data. The machine-learning approaches, such like support vector machines (SVM), Naive Bayes (NB) and Maximum Entropy (ME), are operated on the feature vectors of the labelled dataset to train the classifier. The deep learning approach based on artificial neural networks has grown rapidly over the last few years, which offers a new idea for the classification of sentiments. For better overall classification, the deep CNN is combined with SVM to form a hybrid method [18]. It integrates CNN output into an SVM that improves classification. However, the CNN's fully connected layer is simply act as a classifier and it cannot achieve non-linear classification effectively. Here CNN is then applied on the labelled training data to learn vector representations of features and those features are forwarded to SVM for sentiment classification. This hybrid model is supposed to integrate the benefits of the CNN's feature learning with the

effective non-linear classification of the optimized SVM model.

2. RELATED WORKS

Recently The researchers implemented deep learning frameworks to analyses sentiments. Deep learning gained more fascination due to its correctness and automated feature engineering. The following listed the selective sentiment analysis research works on deep learning techniques. Ashima Yadav and Dinesh kumar [1] evaluated several deep-learning architectures like RNN, CNN and ANN for types of sentiment analysis perhaps sentence level, document based and aspect based, multi-domain and multimodal sentiment classification. Shahida Khanaet al. [2] list out and estimate various hybrid approach that make use of deep network models i.e. CNN, RNN and LSTM with different ML algorithms and resolved deep learning is more successful than others. Hannah Kim and Young-SeobJeong[3] proposed a deep CNN and its structure was carefully crafted for an accurate classification of sentiments on data sets of movie reviews and customer reviews and attained 80.96 % and 81.4% of accuracy respectively. Currently Nature-inspired algorithms [4] are emerging research area for proposing novel algorithms for optimizing the result. Abd. Samad Hasan Basaria et al. [5] performed opinion mining using hybrid SVM and particle swam optimization to improve the accuracy level from 71% to 77%. The main purpose of this study is to identify key features for each sentiment class and to improve accuracy. It has been studied from the literature review that the hybrid approach using machine learning with deep learning architectures realized the benefits of both the algorithms in a single framework. In our proposed method the hybrid approach that combined the deep learning CNN with MSPSO based SVM classifier for appropriate features extraction and optimize SVM parameters to improvise the classification precision.

3. DEEP NEURAL NETWORK

Primarily G.E. Hinton proposed Deep Learning in 2006 and is the machine learning part of the deep neural network [17]. Neural network is constructed by multiple neurons that gives an outstanding network and it is activated like a human brain. Convolutional neural networks, Recurrent neural networks, Recursive neural networks, Deep belief networks etc are some of the deep learning techniques. Models of deep neural networks have obtained excellent execution particularly in computer vision and speech recognition. Researchers used this technology in NLP for the benefits in sentence modelling, text generation, word embedding, vector representation, feature extraction and sentence classification. The tasks of sentiment analysis can be performed effectively by applying various deep learning methods, which have recently been expanded. Hannah Kim et al. proposed [3] the successive convolutional layers for better performance on relatively long text. Guolong Liu et al. [16] proposed a hybrid approach that

incorporates generative and discriminative framework that includes recurrent neural networks for multi-language text sentiment analysis. Kalchbrenner et al. [15] recommended dynamic convolutional neural network (DCNN) to extract the features from the key sentences with fluctuating length and it is able to catch short and long-range relations openly. As described above, two dominant forms of deep learning methodology exist for the sentiment analysis includes RNN and CNN. we proposed a hybrid model using CNN, whose configuration was carefully built for effective sentiment feature extraction.

4. CONVOLUTIONAL NEURAL NETWORK (CNN)

Convolutional neural network is a specific form of deep-neural multi-layer network that is a back bone of computer vision, image classification [19,20] and semantic segmentation. however, Recently Convolutional Neural Networks have found popular in tackling issues related to NLP tasks as text classification, sentiment analysis etc, which consists of successive convolutional and pooling layers tailed by fully-connected layers.

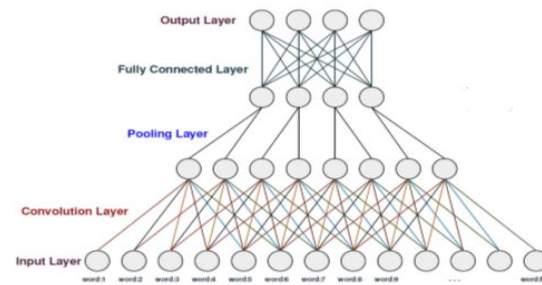


Figure 1: CNN Architecture

Implicitly, these layers derive appropriate feature description from given data and It is transmitted to fully connected layers for classification. The size and weights of the convolution filter decide which features to extract from input data. Max pool layer is then added to pick the most appropriate features from the convolutional layered output. Consequently, this process is iterated in many convolutional and pooling process then fed into fully connected layers for classification. Usually the soft-max is an activation function in fully connected layer. Figure 1 shows the typical CNN architecture and its layers. Here, the layers act like filters. The feature map reflects the filter size while each edge resembles the filter weight.

5. SUPPORT VECTOR MACHINE (SVM)

SVM is a typical machine learning technique based on statistical method. It is highly efficient in text classification due to the transformation of input data into high-dimensional feature vector by non-linear transformation and then to identify the optimal feature in the next space. Usually, the higher dimension will initiate the computation difficulty, but the introduction of kernel

function in SVM algorithm resolve computational complexity and the Curse of dimensionality. The choice of different kernels is used to construct different SVM classifier. There are three common kernel functions:

Linear kernel function:

$$K(y_i, y_j) = (y_i^T \cdot y_j) \quad (1)$$

Polynomial kernel function:

$$K(y_i, y_j) = (\gamma y_i^T y_j + r)^d, \gamma > 0 \quad (2)$$

Radial basis function:

$$K(y_i, y_j) = \exp(-\gamma \|y_i - y_j\|^2), \gamma > 0 \quad (3)$$

Here, kernel parameters are λ , r , and d .

However, SVM attains linear hyper plane by exposing input data in to advanced dimension and generalizes well. The feasibility of SVM is improvised by selecting appropriate SVM parameters [7]. In our proposed work, we consider and integrate this concept by replacing soft-max regression with improved SVM at the CNN output layer.

6. MSPSO BASED FEATURE SELECTION

Multi-swarm particle swarm optimization is constructed from PSO for the optimized feature selection is defined in Figure-2. Here Particle swarm optimization (PSO) is an optimization algorithm that is stimulated by bird flocking behavior of the birds [6]. The process of feature selection can be exhibited as follows: $G_d = \{g_1, g_2, \dots, g_d\}$ are well-ordered feature set with reducing rates of occurrences. All the review are represented as feature vector that consist of feature frequencies i.e. $r_i = \{f_{i1}, f_{i2}, f_{i3}, \dots, f_{id}\} (r_i \in \mathbb{R})$ f_{ij} refers to the occurrence rate of feature g_{ij} in review r_i . Let we assume $R = \{(r_i, s_i) | 1 \leq i \leq L\}$ is a training samples where $s_i \in S = \{-1, 1\}$ positive sentiment is indicated by "1" and negative sentiment is indicated by "-1". The strategic objective of feature selection technique is to develop various distinct feature selection function $f_t(G_d) (t=1, 2, 3, \dots, T)$ by separating sample space. In this evaluation function $f_t(G_d)$ is categorized by a binary particle swam $X^t = (X^t_1, X^t_2, \dots, X^t_M)$ and it is produced from t^{th} sample subspace. In that swarm every single particle $X^t_p = [x^t_{p1}, x^t_{p2}, x^t_{p3}, \dots, x^t_{pd}]^T (x^t_{pq} \in \{0, 1\} | 1 \leq p \leq M, 1 \leq q \leq d)$ contain d elements and related to single dimension of G_d . After finishing a number of progressions of each and every particle swarm, the results of T evaluation are merged to acquire the features that are decreasingly classified corresponding to discriminability. The exact d' features are selected by formula (4).

$$\text{BestF}(g_1, g_2, g_3, \dots, g_d) = \text{sorting}(\sum_{t=1}^T f_t(G_d)) \quad (4)$$

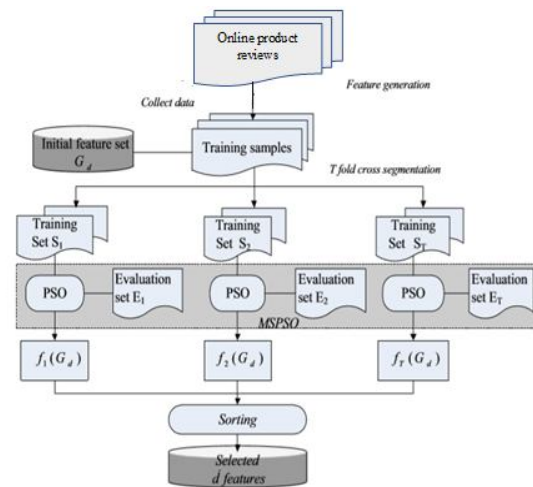


Figure 2: Feature selection process with MSPSO

7. CNN-BASED SVM CLASSIFIER

The three evident features of CNN are local connectivity, spatial sampling and weight sharing. In neural network architecture, the parameter size is drastically reduced by local connection operated among the neurons. We may use the repeated convolution and polling to extract the features for training and classification function. but it will require a lot of computing. The dimension of the obtained feature is reduced by spatial sampling process, this is to reduce the computational difficulty in the upper layers and improves the robustness. Furthermore, CNN's weight sharing network configuration highly simulating the biological neural networks, and it is used to avoiding complicated feature extraction process and data reformation method in machine learning algorithms. Thus, it has been broadly used and proposed by kim [14] in the field of NLP task. The fully interconnected layer as a classifier in CNN-based model and it can't perform non-linear classification effectively. SVM is a kind of supervised machine-learning algorithm for classification and regression analysis and it has always been active and stable compared with other conventional machine learning algorithms, but there is a difficulty for implementing large-scale training data to handle multi-classification task. In this paper, we proposed a deep learning CNN with SVM that handles CNN as the automated feature learner to extract the sentiment features and SVM as the sentiment classifier. But The existing CNN architecture simply append all input features at the input layer and these features are not well replicated in the output because of iterative convolution and pooling process. Soft-max is a weak classifier and has constraint over SVM. In our method, we plan to address all the issues, such hybrid model which uses CNN, SVM and MSPSO to performs sentiment analysis is novel as well as more effective for dataset from different domain. The outputs of CNN, the sentiment features of the input sentences and the optimized sentiment features derived from MSPSO are considered as input features to trained the SVM classifier. When this model is applied to the

classification of sentiments. It is expected that such a hybrid model blends the advantages of CNN, SVM and optimized feature vector using MSPSO. To examine the effect of externally derived optimized features, we created various basic models and compared with each other.

8. METHODOLOGY

The proposed Hybrid CNN-SVM approach works in three stages are well-defined in Figure-3.

1. CNN architecture is used to learn sentiment embedded vector.
2. Multi-swarm particle swarm optimization algorithm (MSPSO) is to construct sentiment augmented optimized features.
3. Training of SVM with non-linear kernel make use of the CNN trained features in step-1 and MSPSO based optimized features of step-2.

In step-1, We then train a CNN using a convolution and pooling layers with stochastic gradient descent back-propagation algorithm. The configuration of CNN was carefully built for effective sentiment feature extraction. Parallel, In Step-2, The MSPSO based optimization algorithm is employed to detect the most related set of features within the context of SVM. The optimal parameter of CNN is found once after its training, in step-3 we combine the output of top most hidden layer and optimized feature vector derived by MSPSO and supply it to SVM.

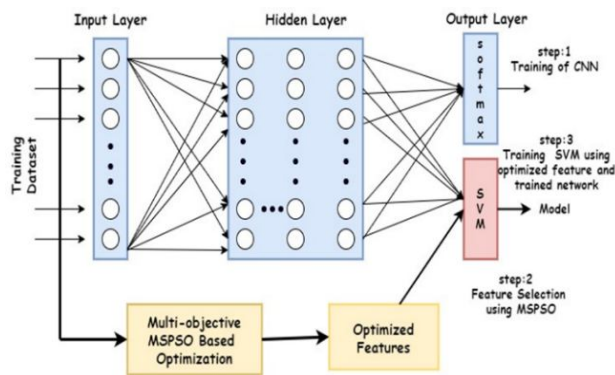


Figure 3: Proposed methodology

CNN is performing well to obtaining the significant lexical and syntactic features on its own structure. Hence The first phase of the proposed method ensures that the relevant features are extracted automatically from the training data. Then in the final phase of the proposed method, the SVM makes use of the features extracted from CNN as well as the MSPSO based optimized features to describe a further improvement in hyperplane compared to those derived by either CNN or SVM with optimized features on its own. The pseudo code of the proposed approach is outlined in Algorithm 1. Code 1-6 described the step-1 i.e. training of the

CNN to acquire sentiment embedded vectors and code 7 is for finding optimized feature vector using MSPSO i.e. step-2. The step-3 is carried out by code 8-14.

```

Algorithm 1 (Predict, Accur) = Hybrid CNN-SVM(W+O) (Train, Div, Test, Test-Final,  $\theta$ )
Require: Train, Div, Test, Test-Final, Datasets;
 $\theta$  - Termination condition
Ensure: Predict  $\rightarrow$  Predicted sentiment output;
Accur  $\rightarrow$  Classification Accuracy attained.

1: Net  $\rightarrow$  Create_Network()
2: Network_initialize(Net)
3: for error  $\geq \theta$  do
4: error Network_Train(Net, Train, Div)
5: end for
6: /* Training completed */
7: Featureopt  $\rightarrow$  MSPSO(Train, Div)
8: HTrain  $\rightarrow$  GetTop_HiddenLayer(Net, Train)
9: Train_combined  $\leftarrow$  HTrain + Featureopt
10: ModelSVM  $\leftarrow$  SVMTrain(Train_combined)
11: HTest  $\leftarrow$  GetTop_HiddenLayer(Net, Test)
12: Test_combined  $\leftarrow$  HTest + Featureopt
13: Predict  $\leftarrow$  SVMTest(ModelSVM, Test_combined)
14: Accur  $\leftarrow$  Evaluation (Test-Final, Predict)
15: return (Predict, Accur)
    
```

9. EXPERIMENTAL SETUPS AND TOOLS

In our work, CNN has been constructed by considering a number of parameters in each layer and it is stated in Table 1. The parameter values were set by analyzing the studies carry out in this field by other researchers [3]

Table 1: Parameter configuration of CNN

Name of parameter	Value of parameter
Number of convolution layer	128
Convolution kernel size	5*64
Pooling layer	1-max pooling (1 * 1)
Hidden layer	128
Character vector dimension	64
Number of epochs	10
Reserved node probability	0.5
Batch size	64
Dropout	0.2
Learning rate	1e-3

To comparing the performance of the proposed method, we described three basic models as follows:

- SVM_{Base} :. This model is based on SVM and integrates all the existing features.
- CNN_w : It is a basic model based on CNN with embedding word as a features to train and test the system .
- CNN_(w+o) : It is a basic model based on CNN, trained and tested using both embedding words as features and optimized features extracted by MSPSO based feature selection techniques.

In addition, in the presence or absence of extra handcrafted elements, we seek to understand the behavior of the proposed method. We describe one more model based on CNN-SVM architecture for comparative analysis.

- Hybrid CNN-SVM_w: This is the proposed method without the optimized feature set. Only embedding words as features for training and evaluated the model. The features extracted from the top most hidden layer is forwarded to SVM for its training.
- Hybrid CNN-SVM_(w+o): This is the proposed model with the usage of word embedding and optimized features derived using MSPSO based feature selection process.

We have used Jupyter Notebook [11] as open-source development environment for implementing all baseline and proposed models. This is a web-based application for generating and distributing of documents comprising of test, code, formulas and visualizations and It is used to perform data cleaning, transformation and statistical and machine learning models etc. In order to produce rich set of features, the input data are pre-processed by eliminating unrelated data such as stop word, punctuation etc, then it is followed by stemming, term frequency identification using python coding. The TFLearn[12] is a python package in TensorFlow[13] for developing and speedup all deep learning methods. First of all, the sentences are converted to list of sequences. This is performed by TFLearn vocabulary processor we need to translate these sequences to numerical values, because a neural network model can only perform operations over numbers.

10. DATASET DESCRIPTION

For the experimental evaluation of multi domain data set. The Amazon own product review data set from Kaggle around 45000 customer reviews are collected. Here the reviews are sub divided in to three domain related datasets such as kindle and fire tablet and fire tv then these reviews are categorized into the probable ways i.e. positive, negative and neutral based on the star rating of the review. Here, a rating of 1 or 2 stars are labelled as negative while 4 or 5 were labelled as

positive and 3 stars rated reviews representing the neutral class were represented in table 2.

Table 2: Sentiment labels based on star rating

Domain	No.of reviews	Positive	Negative	Neutral
Kindle	16040	6864	5478	3698
Fire tablet	15269	5576	5123	4570
Fire TV	12468	5311	4584	2573

11. RESULT AND DISCUSSION

In order to highlight the advantages of our proposed approach. In each domain 70% of the collected reviews are for training the models and the remaining 30% is for testing the performance of the proposed approach. Table 3 shows the performance measures of all the models in each domain. Table 4, Figure 4 shows the accuracy of all the models in each domain. The performance metrics are precision, recall, F-score and accuracy. These four active measures are founded on confusion matrix output as True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN).

$$\text{Precision(P)} = \text{TP}/(\text{TP}+\text{FP})$$

$$\text{Recall(R)} = \text{TP}/(\text{TP}+\text{FN})$$

$$\text{Accuracy(A)} = (\text{TP}+\text{TN})/(\text{TP} + \text{TN} + \text{FP} + \text{FN})$$

$$\text{F-score} = 2. (\text{P.R})/(\text{P}+\text{R})$$

The effectiveness of text categorization is typically calculated using the F-score and accuracy. F-score is a combined measure of effectiveness based on precision and recall.

Table 3: Performance measures in percentage

Model	Kindle			Fire Tablet			Fire Tv		
	P	R	F1	P	R	F1	P	R	F1
SVM _{Base}	72	67	69	70	65	67	73	68	70
CNN _w	79	76	78	77	72	74	78	75	76
CNN _(w+o)	81	76	78	79	76	77	82	79	80
Hybrid CNN-SV M _w	82	84	83	81	82	81	86	84	85
Hybrid CNN-SV M _(w+o)	98	95	96	95	92	93	95	90	92

Table 4: Classification Accuracy

Model	Kindle	Fire Tablet	Fire Tv
SVM _{Base}	70	68	71
CNN _w	77	75	76
CNN _(w+o)	79	77	80
Hybrid CNN-SVM _w	83	81	85
Hybrid CNN-SVM _(w+o)	96	94	93

In comparison of baseline CNN_w, CNN_(w+o) with augmenting external features shows better accuracy. For all other settings we observe similar phenomenon as well. Results state that the fusing of SVM on top most layer of CNN, i.e. CNN-SVM_w is outshine than the typical CNN. Addition of external feature in CNN-SVM_(w+o) achieved an average accuracy of 94.3% as compared to 83% without it.

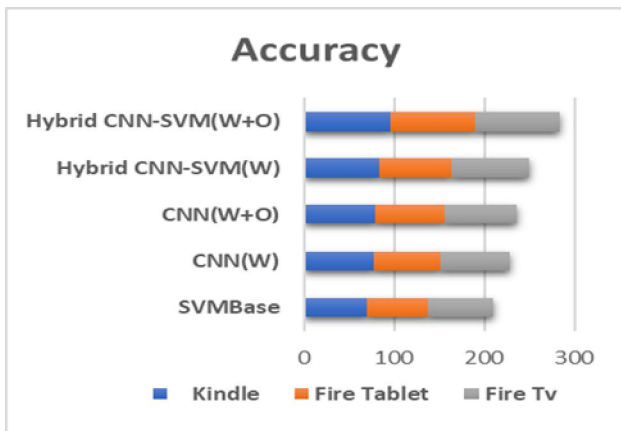


Figure 4: Classification Accuracy

In CNN-SVM_(w+o), Using SVM instead of the conventional soft-max function on CNN with a sentiment embedded vector will generate the decision hyperplane in particular by proposing the CNN features into an advanced dimension. In addition of CNN features, the optimal collection of features includes sentiment enhanced optimized vector supplied by MSPSO based feature selection technique assists SVM with accurate prediction

To further check the performance of the proposed algorithm, the training time of all baseline and proposed algorithms were calculated on datasets of different size and it is shown in Table -5 and Figure-5. We must emphasize that, like all other models of neural networks, these deep architectures are extremely parallel, with a drastic reduction in training time. Meanwhile, The SVM_{Base} classification model requires enormous number of weights and is really less parallel, so it takes lot of time for training, All CNN based models are relatively less time-consuming due to less training time. On the basis of a comparison of performance metrics and training time, CNN-SVM_(w+o) algorithm is best for domain independent data sets.

Table 5: Training time in seconds

Dataset size	SVM Base	CNN w	CNN (w+o)	Hybrid CNN-SVM w	Hybrid CNN-SVM (w+o)
	Training time in seconds				
5000	1258	194	150	102	64
10000	1587	243	200	120	87
15000	1703	268	225	127	91
20000	2458	348	306	160	120
25000	4987	639	523	380	231
Average value	2398	338	281	178	119

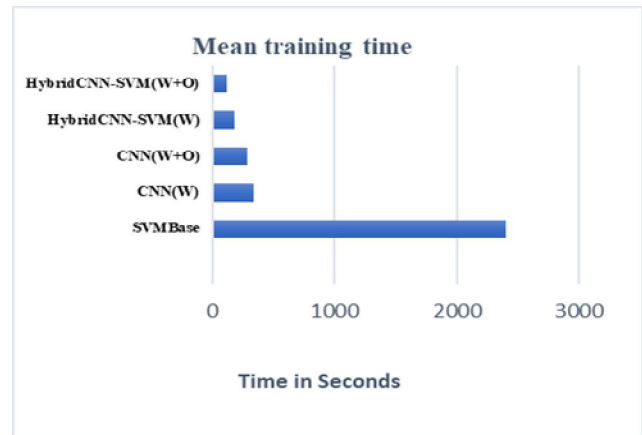


Figure 5: Mean Training time

12. CONCLUSION

Traditionally, an analysis of sentiments has concentrated on text analysis using NLP and feature-based machine learning techniques. The exponential growth of Big Data and Deep Learning technologies has influenced and facilitated the development of this field. In this research, we proposed a hybrid CNN with improved SVM for sentiment analysis of domain independent datasets. The CNN architecture is used to learn sentiment embedded vector and obtain sentiment classification by replacing soft-max function with a strong classifier, i.e. SVM as CNN output layer. This approach handles CNN as a feature learner that constantly learns the representations of features from the given data. In addition, the hybrid model obtains the benefits of classification efficiency with SVM classifier to improve the CNN’s classification capability. SVM training is further facilitated by the optimized feature set derived from MSPSO-based feature selection technique. We constructed number of baseline models and test our proposed method on the domain independent product review data sets. We found that using CNN-SVM with optimized features in the proposed algorithm assist to get promising performance across the domains in comparison with advanced methods. Future work involves research on RNN deep learning methods as well as the addition of the data set from public domains such as social

media and Political information to generally construct the sentiment analysis system. so that it can be used for the benefit of people in other applications too.

REFERENCES

1. Ashima Yadav, Dinesh Kumar Vishwakarma, **Sentiment analysis using deep learning architectures: a review**, *Artificial Intelligence Review*, Springer Nature B.V.2019, <https://doi.org/10.1007/s10462-019-09794-5>
2. Shahida Khan, Kamlesh Chopra, Vijay Malviya, **Sentiment Analysis based on Hybrid Approach: A Survey**. *Proceedings of International conference on Recent Advances in Interdisciplinary Trends in Engineering & Applications (RAITEA)* 2019.
3. Hannah Kim and Young-SeobJeong, **Sentiment Classification Using Convolutional Neural Networks**, *Appl.Sci.* 2019, <https://doi.org/10.3390/a pp9112347>
4. Ashima Yadav, Dinesh Kumar Vishwakarma, **A comparative study on bio-inspired algorithms for sentiment analysis**, *Cluster Computing, part of Springer Nature* 2020. <https://doi.org/10.1007/s10586-020-03062-w>
5. Abd. Samad Hasan Basaria, Burairah Hussina, I.Gede Pramudya Anantaa, Junta Zeniarjab **Opinion Mining of Movie Review using Hybrid Method of Support Vector Machine and Particle Swarm Optimization**, *Proceedings of the Malaysian Technical Universities Conference on Engineering & Technology*, pp.453–462, 2013, doi:10.1016/j.proeng.2013.02.059.
6. Ming-Yuan Cho and Thi Thom Hoang, **Feature Selection and Parameters Optimization of SVM Using Particle Swarm Optimization for Fault Classification in Power Distribution Systems**, *Hindawi Computational Intelligence and Neuroscience-2017*. <https://doi.org/10.1155/2017/4135465>
7. Tatjana Eitricha, Bruno Langb, **Efficient optimization of support vector machine learning parameters for unbalanced datasets**, *Journal of Computational and Applied Mathematics* 196, 2006 pp. 425–436, doi:10.1016/j.cam.2005.09.009
8. Ilkayyelman, Metinzontul, Oguzkaynar, Ferdi sonmez , **A Novel Hybrid Approach for Sentiment Classification of Turkish Tweets for GSM Operators** , *International journal of circuits, systems and signal processing*, ISSN:1998-4464, 2018
9. D. V. Nagarjuna Devi, Thatiparti Venkata Rajini Kanth, Kakollu Mounika and Nambhatla Sowjanya Swathi, **Assay: Hybrid Approach for Sentiment Analysis**, *Smart Innovation, Systems and Technologies* 106, Springer Nature 2019, https://doi.org/10.1007/978-981-13-1742-2_30
10. Nurulhuda Zainuddin, Ali Selamat, **Sentiment analysis using Support Vector Machine** 2014 *International Conference on Computer, Communications, and Control Technology* 2014, ISBN: 978-1-4799-4555-9, doi: 10.1109/I4CT.2014.6914200
11. **Jupyter**[Online]. <http://jupyter.org/>. Accessed on 5 March 2020
12. **TFlearn**: Deep learning library featuring a higher-level API for TensorFlow [Online]. <http://tflearn.org/>. Accessed on 3 March 2020
13. **Tensorflow**[Online]. <https://www.tensorflow.org/>. Accessed on 3 March 2020
14. Kim, Y.: **Convolutional neural networks for sentence classification**. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1746–1751, October 2014 <https://doi.org/10.3115/v1/D14-1181>
15. Kalchbrenner, N., Grefenstette, E., Blunsom, P. **A convolutional neural network for modelling sentences**. In: *Proceedings of ACL*, pp. 655–665. Association for Computational Linguistics, Baltimore, June 2014 <https://doi.org/10.3115/v1/P14-1062>
16. Guolong Liu, Xiaofei Xu, Bailong Deng, Siding Chen, Li LA, **Hybrid Method for Bilingual Text Sentiment Classification Based on Deep Learning**, *IEEE-2016*
17. QuratTul Ain, Mubashir Ali, Amna Riaz, **Sentiment Analysis Using Deep Learning Techniques: A Review (IJACSA)** *International Journal of Advanced Computer Science and Applications*, Vol. 8, No. 6, 2017 <https://doi.org/10.14569/IJACSA.2017.080657>
18. Raviya.K, Mary Vennila.S **A Novel Deep Learning Based Sentiment Analysis of Movie using Hybrid CNN_SVM Algorithm** *IJRTE*, ISSN: 2277-3878, Volume-8 Issue-4, November 2019, doi: 10.35940/ijrte.D7115.118419
19. Jennalyn G. Nicolas-Mindoro, Mon Arjay F. Malbog, Joshua S. Gulmatico, Jennifer B. Enriquez **NotiPoy: Early Notification of Detected Fire Using Image Processing Applying Convolutional Neural Network**, *International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE)* Volume 8, No.6, pp. 707-719, 2019, <https://doi.org/10.30534/ijatcse/2019/21862019>.
20. Adil Alharthi, Nouf Alzahrani, Ikram Moalla, **Convolutional Neural Network based on Transfer Learning for Medical Forms Classification** *International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE)* Volume 8, No.6, pp.3405-3411, 2019, <https://doi.org/10.30534/ijatcse/2019/115862019>.