Performance Comparison for Spam Detection inSocial Media Using Deep Learning Algorithms

Mr. Vikram Bhalerao[1], Prof. Rushali A. Deshmukh[2]

*[1] Dept. of Computer Engineering JSPM’s Rajarshi Shahu College of Engineering
Tathawade.Pune – 411033., India
[2] Dept. of Computer Engineering JSPM’s Rajarshi Shahu College of Engineering
Tathawade.Pune – 411033., India
[1] mailtovikrambhalerao@gmail.com, [2] radesh19@gmail.com*

Abstract

People are connected through Twitter or Facebook or any other social medial tool. However, this resulted in messages with malicious content and malware links. Therefore, it's required to own a powerful spam detection design that might stop these styles of messages. Spam detection in hissing platform like Twitter remains a tangle, thanks to short text and high variability within the language utilized in social media. In this paper, we tend to propose a CNN algorithmic technique and compare results with variants of CNN and with boosting algorithms. The model is supported with the assistance of knowledge-bases such as Word2vec and fastText. The use of these knowledge-bases improves the performance, by providing higher linguistics vector illustration of input testing words. Projected Experimental results with input datasets show the effectiveness of the proposed model in terms of accuracy and F1 Score.

Keywords: *Convolutional Neural Network, Sentiment Analysis, Word2Vec, fastText.*

1. Introduction

Web-based social systems have enabled new community-based opportunities for users to communicate with people. This community value is threatened by spammers, malicious links, and malware [1]. Spam Identification started with manual detection of messages, or by setting filtering rules that would notice a message with some renowned properties. Smart spam detection started with the employment of ancient machine learning ways that do not produce spam detection models. Initially, spam started spreading with email spams. Further, SMS is the price-effective technique used for converting individual messages to the vector form.

Possible purchasers encompass a faster response as compared to junk/spam email. Likewise, Twitter and Facebook also contribute to spam messages. Spam detection could be a complex task without any filter kept in at the receiving end. Earlier classifiers used to be rule based. These classifiers were used to get deployed to a large space of purchasers. Each input message sent through already defined rules, which when produces a higher than threshold score, used to get labeled as spam. Even after the spam is detected, the success of these ways is restricted, and they need to be combined with different machine learning ways so as to give fairly sensible results.

Classifiers such as SVM, Naive Bayes, ANN, and Random Forests are complicated, due to feature extracting options from the text. The Bag-of-words (BoW) is used for the extraction of features. Convolutional Neural network is the deep learning algorithm that addresses the accurate classification of the text messages as spam or ham.

2. Literature Survey

Gauri Jain et al. [1] had proposed a model that was centered on spam recognition short spam content, for example, Twitter. Whereas before this, existing fruitful methodology essentially centered around long email messages anomaly. A deep learning-based methodology has been proposed, which was comprised of CNN and LSTM neural models. Exploratory outcomes showed that the proposed approach beats all other approaches with input two datasets for Twitter text and SMS content each.

Thayakorn Dangkesee et al. [3] has efficiently projected the accommodative knowledge classification model, detecting spams by the victimization of spam word lists using a billboard (URL). Naive Bayes algorithm has been used for classification. This has helped in an efficient performance boost.

Katpatal and Junnarkar [4] has proposed the new training dataset, which was used to train another dataset containing unlabeled tweets. These had resulted in the ending of spam tweets. The author has proposed a scheme that adjusts training data sets. Dropping too old samples after a certain time has helped to eliminate unusual information saving space.

Kamble. et al. [5] has exhibited the plan of ongoing vector space denotation of words. Evaluation of a novel AI-based way to deal with specialization Social spam detection. Their general research goal for consequently shifting and recognizing spammers who point social destinations was to discover methodology ie.SAND, to find compelling devices.

Guanjun Lin. et al. [6] has detected the nine mainstream algorithms that were compared to understand the most suitable algorithm. The stability of each algorithm had been studied thoroughly. It had indicated the variation of the training time according to the CPU core.

Tingmin Wu et al. [7] has proposed a system of twitter spam detection. The paper had addressed the then-existing challenges like low speed and feature extraction difficulties thoroughly. It had experimentally proven the comparisons between achieved results and existing accuracies through the indication of graphs.

3. Proposed Methodology

The system consists of a model that is trained on Convolutional Neural Network rule. On general terms, CNN is most well-liked for image classification. The variation of CNN 1D has been used for spam text information classification. CNN contains various types of layers that are typically improved in terms of accelerating accuracy during and after the implementation. Here, in the projected system, CNN will work as a classifier to investigate whether or not, the text statement is spam. fastText created by Facebook's AI analysis (FAIR) laboratory has been used for word embeddings along with the Word2Vec module. The model permits to form associate unattended learning rule or supervised learning rules to achieve vector representation. Facebook offered pre-trained models' numerous languages. The neural network is a key factor used in fastext for word embedding. The proposed framework comprises of Word2Vec and fastText advancements along with Convolutional Neural Network (CNN) to complete the model. The proposed framework will also be comprised of varieties of CNN models in terms of the number of filters and convolutional layers of the CNN algorithm. These can be used for performance comparisons between different variations of CNN. The research will work around CNN varieties and layers.

A. Architecture

- The proposed system has an advantage of multiple platforms for computer files for model development.
- System has been developed with quite one algorithmic program, so Prediction guarantees are inflated.
- Live updates area unit involved in prediction, so it area unit typically used for live recommendation.
- The proposed system varies in the filter and convolutional layer combination.
- The proposed system will compare the results of the CNN algorithms with that of the boosting algorithms and hence will try to achieve the maximum accuracy with better precision.
- We are also trying to train the module through the vernacular language data sets like Hindi or Marathi.
- Figure 1 shows an overview of the proposed system architecture.
- Out of the complete training data set, 80% of the data will be used to train the model using a supervised learning technique. The remaining 20% of the data set will be used as a test set to generate the accuracy and response time calculation tests.

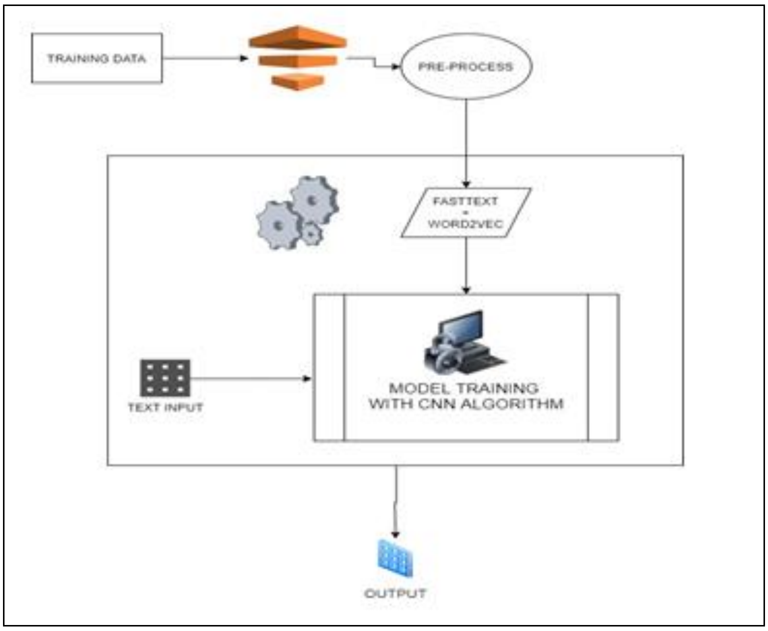


Figure 1. Proposed System Architecture

B. Algorithms

Scope of fastText: FastText as an available library for vector representations and classification of input texts. It is used for processing the number of tasks. It is based on C++ platform. fastText allows the end-user or the developer to train unsupervised or supervised representations of input texts or sentences. These representations, which are also called embeddings, are often used for varied applications, as features to models, for cluster selection. Generally, it trains the models of Skip-gram or continuous bag of words (CBOW). It used SoftMax, negative sampling or hierarchical loss functions. fastText can be crucial in training the word embeddings for a large size within the order of billions of words.

Scope of Word2Vec: Word2Vec may be a shallow, two-layered neural network which is trained to reconstruct linguistic contexts of words. Every word gets assigned with vector representations in the vector space provided the word should be unique. These Word vectors are placed within the vector pace. Words that represent common contexts within the class are identified or placed in close juxtaposition with each other within the already allocated predefined or precalculated vector space. Word2Vec is a very effective predictive model for learning word embeddings from raw text [8]. The continual Bag-of-Words (CBOW) model and Skip-Gram model are the two variants of word2Vec [9].

CNN 1D: CNN-Convolutional Neural Network model, are generally established for image classification, where the model takes a two-dimensional input representing an image's pixels and color channels, this process is called as feature learning. One-dimensional (1D) input text can be applied with the same logic. The model extracts feature from categorizations of data and charts the interior topographies of the sequence. The 1D CNN algorithm is extremely operative for originating features from a fixed-length section of the general dataset, where it is not so significant that the feature is found within the section or not.

A.<u>1D CNN Algorithm:</u>

The Algorithm of a CNN(1D) is formed through the next important steps:

```
Input:
Sentence matrix x(L*d), F filters
Process:
Assign Weightage W and process filter F
  For {                  ...   # first CNN variation 'V1' For {
                         ...   # each epoch N
          1.CNN layers which are hidden and     MLP neurons
          2.Kernel size (In the proposed system the kernal size is 3)
          3.Subsampling factor in each CNN layer. wj = [xj + xj +1 + …+ xj + k-1]
          4.Aactivation functions. (ReLU (wjn +b))
          5. Probabilities of output
           6.Error
           7. Backward propagation.
        }
      }................................. #end for CNN variation 'V1'
Output:
The trained model of the CNN classifier is produced.
```

In each CNN-layer, equation (1) represents the 1D forward propagation (1D-FP) [9]:

$$x_k^l = b_k^l + \sum_{i=1}^{N_{l-1}} conv1D\left(w_{ik}^{l-1}, s_i^{l-1}\right) \qquad (1)$$

B. <u>End to End CNN algorithm [1]</u>

```
**Input**: Sentence Matrix $x(L \times d)$, F filters
**Output**: Most Important Features
    for each filter $f \in \{1, ..., F\}$ do //Get the most important features
        $w_j = [x_j + x_{j+1} + ... + x_{j+k-1}]$
        $c_j = ReLU(w_j n + b)$
        $c = (c_1 \oplus c_2 \oplus ... \oplus c_j)$
    end for
```

Where,

- x is the input term; d is the length of the word vector.
- $x \epsilon R^{L \times d}$ Represents input sentence with length L.
- f stands for the CNN filter
- k denotes the length of the words.
- wj is the window.j is the starting position of the word and (j+k-1) is the last position.
- $(+)$ Denotes the concatenating operation for the vectorized words.
- ReLU is the sigmoid function.

4. Results and Discussion

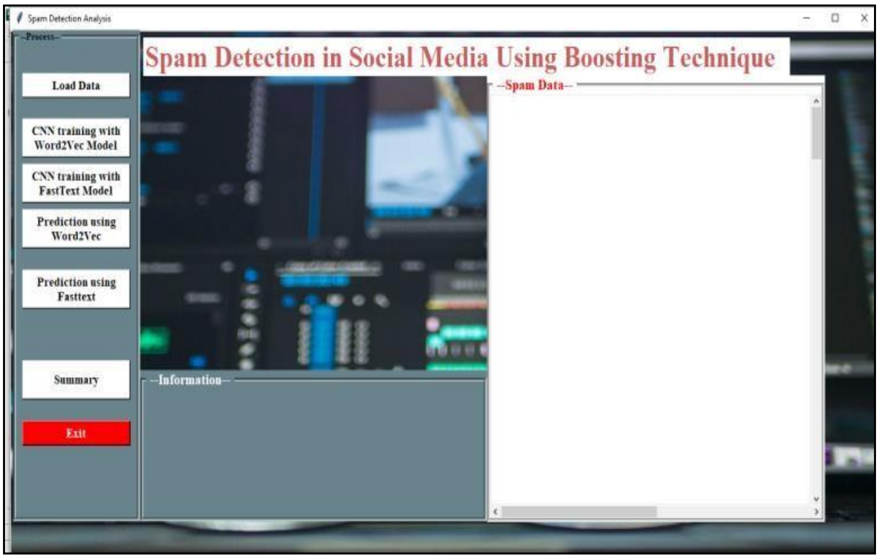
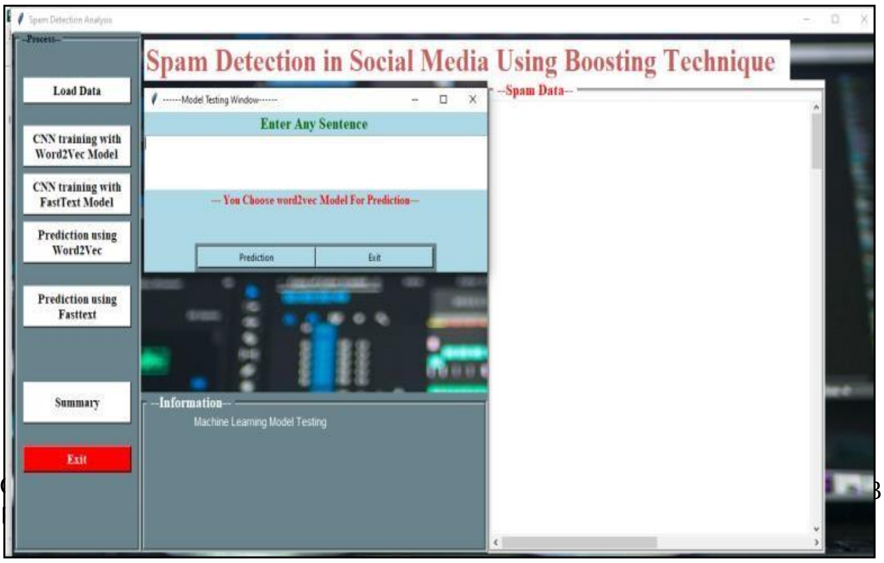

Spam Detection Analysis
--Process--
Load Data
CNN training with Word2Vec Model
CNN training with FastText Model
Prediction using Word2Vec
Prediction using Fasttext
Summary
Exit
Spam Detection in Social Media Using Boosting Technique
--Spam Data--
--Information--

Figure 2. Home Page of proposed system.


Spam Detection Analysis
--Process--
Load Data
CNN training with Word2Vec Model
CNN training with FastText Model
Prediction using Word2Vec
Prediction using Fasttext
Summary
Exit
Spam Detection in Social Media Using Boosting Technique
--Spam Data--
------Model Testing Window------
Prediction
Exit
--Information--

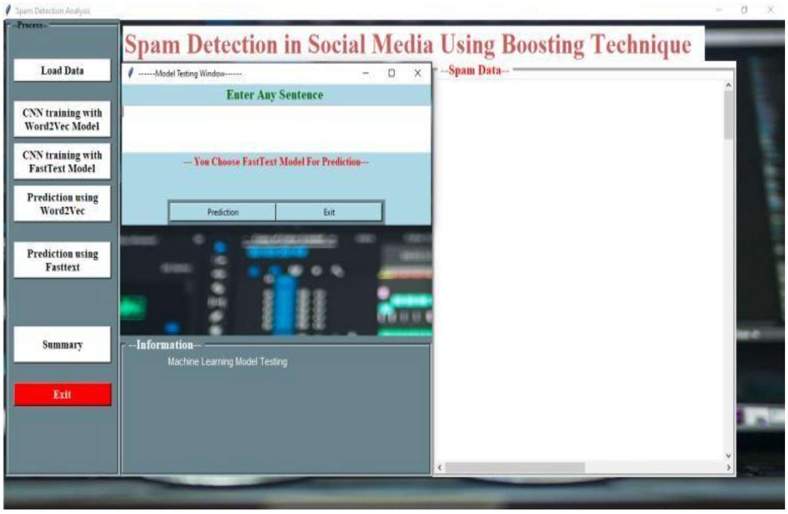
Enter Any Sentence

--- You Choose word2vec Model For Prediction---

Machine Learning Model Testing

Figure 3. The Prediction window from Word2Vec Model

1. Figure 2 indicates the overview of the CNN based structure model. This window contains the main modules of the final model, which in turn runs the CNN algorithms at the back end.
2. After this output window will appear with the prediction window using CNN and word2vec. Here, the input words are converted to vector forms and then are fed to the CNN algorithm for the classification using filters and convolutional layers, as represented in the Figure 3
3. Figure 4 shows authenticated user needs to give input in the form of text, and then the model will give output. In this case, the model will process the input text into vector form and then will feed as the input to the CNN algorithm, and based on the labeled learning, CNN will classify the sentence into spam or ham.
4. The graphical representation of the response time of the CNN classification using word2vec can be seen when we click on the prediction tab. Here the epoch value will decide the number of iterations. The number of points mentioned in the graph indicates the number of iterations. Refer to Figure 5.
5. The Accuracy of the prediction, based on the 80% learning data set can be shown through the graphical representation of the Model. Refer to Fig. 1.5. This accuracy has been reached to 95% when we have used the word2vec module and CNN and LSTM algorithms as endpoints.



Spam Detection in Social Media Using Boosting Technique

Enter Any Sentence

— You Choose FastText Model For Prediction—

--Information--

Machine Learning Model Testing

Spam Detection Analysis
--Process--
Load Data
CNN training with Word2Vec Model
CNN training with FastText Model
Prediction using Word2Vec
Prediction using Fasttext
Summary
Exit
------Model Testing Window------
Prediction
Exit
--Spam Data--

Figure 4. Predictions from fastText Model.

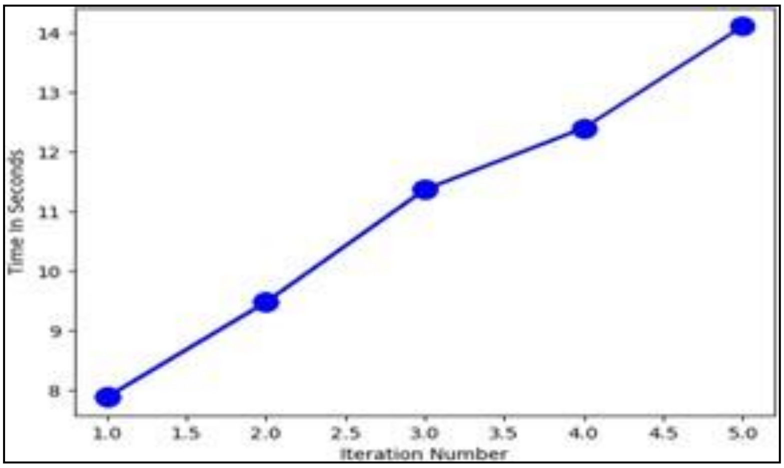


Figure 5. Performance graph of response time from Word2Vec model

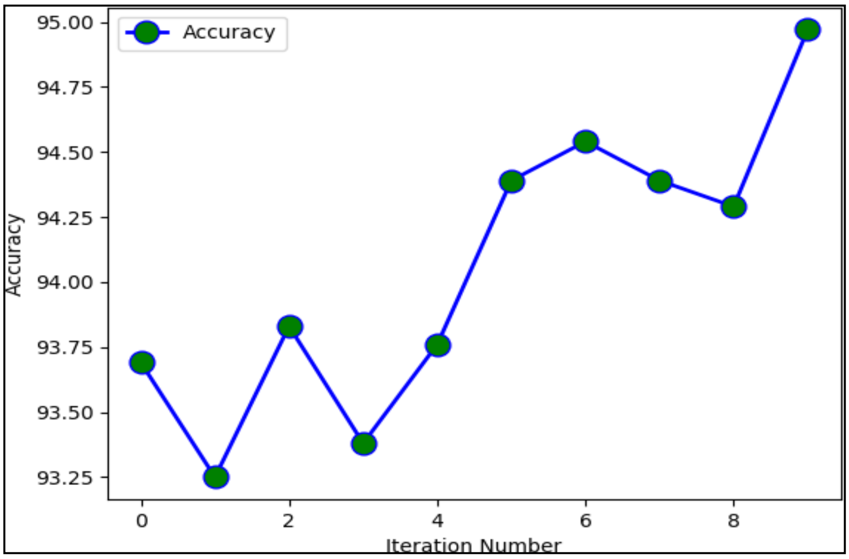


Figure 6. Accuracy Graph of the training model.

5. Conclusion

After research implementation of proposed project and idea, the system reflects the conclusion that Word2Vec and fastText both perform accurately in terms of accuracy performance. In comparison, it has been concluded that the Word2Vec model takes around 8 to 12 seconds to load and respond as shown in the graphical representation in the results and discussion section. However, the fastText model takes much more time to load and respond. fastText takes around 1 to 2 minutes. Both are relying on the earlier stage of model training. 1D Convolutional Neural Network algorithm is used for the training model. The number of iterations of the execution has been based on the epoch.

As a part of the next step of the proposed model, we will be varying CNN configuration in terms of layers, the number of filters and multilingual twitter dataset like Hindi, Marathi. After deriving accuracy, later on, these respective results will be performance compared with the results of the ADA/XG boosting algorithm. Classifiers can be re-trained by the added “changed spam” tweets that are learned from unlabeled models; thus, it can decrease the effect of “Spam Drift” expressively [7].

References

[1] Jain, G., Sharma, M. and Agarwal, B." Spam detection in social media using convolutional and long short-term memory neural network." Annals of Mathematics and Artificial Intelligence, 85(1), **(2019)** pp.21-44.

[2] Lee, K., Caverlee, J. and Webb, S. "Uncovering social spammers: social honeypots+ machine learning." In Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval, ACM, **(2010)** July, pp. 435-442.

[3] Dangkesee, T. and Puntheeranurak, S.,. Adaptive Classification for Spam Detection on Twitter with Specific Data. In 2017 21st International Computer Science and Engineering Conference (ICSEC) IEEE, **(2017)**, November, pp. 1- 4.

[4] Katpatal, R. and Junnarkar, A. "An Efficient Approach of Spam Detection in Twitter." In 2018 International Conference on Inventive Research in Computing Applications (ICIRCA) IEEE, **(2018)**, July, pp. 1240-1243.

[5] Kamble, S. and Sangve, S.M. "Real Time Detection of Drifted Twitter Spam Based on Statistical Features." In 2018 International Conference on Information, Communication, Engineering and Technology (ICICET) IEEE, **(2018)** August, pp. 1-3.

[6] Lin, G., Sun, N., Nepal, S., Zhang, J., Xiang, Y. and Hassan, H. "Statistical twitter spam detection demystified: Performance, stability and scalability." IEEE access, 5, **(2017)**, pp.11142-11154.

[7] Wu, T., Liu, S., Zhang, J. and Xiang, Y. "Twitter spam detection based on deep learning." In Proceedings of the Australasian computer science week multiconference, ACM, (2017) January, p. 3.

[8] Chen, C., Wang, Y., Zhang, J., Xiang, Y., Zhou, W. and Min, G. "Statistical features-based real-time detection of drifted Twitter spam." IEEE Transactions on Information Forensics and Security, 12(4), **(2016)**, pp.914-925.

[9] Kiranyaz, S., Avci, O., Abdeljaber, O., Ince, T., Gabbouj, M. and Inman, D.J. " 1D Convolutional Neural Networks and Applications: A Survey." **(2019)** arXiv preprint arXiv:1905.03554.