A Deep Learning Approach for URL based Health Information Search

R. Rajalakshmi and S. Ramraj

Abstract*: Nowadays, Internet has become the major source of information for many people in diverse requirements such as Education, Entertainment, Sports, Travel etc. The demand for seeking health information from web has increased drastically over the last two decades. As the content of web is dynamic in nature, the retrieval of relevant health information is challenging. In this research work, an URL based approach is presented to help the user to identify the health related web page. Instead of using the hand crafted features, a deep learning approach is suggested in which the feature learning power of Convolutional Neural Network (CNN) has been exploited to categorize the health relevant web pages. Character level embedding has been suggested for extracting the appropriate features using CNN and these extracted URL features have been used for classification. Various experiments have been carried out on the benchmark data set (Open Directory Project) to analyze the performance of this approach. We have achieved an F1 measure of 83% for this deep learning based approach and the comparative analysis shows that, there is a significant improvement over the existing works.

Index Terms*: CNN, Deep learning, Health Information Search, URL classification

I. INTRODUCTION

Introduction:

In this digital era, the usage of search engines and recommendation systems grows exponentially. Many people try to make use of the information technologies for various needs. Seeking health related information from the web has increased dramatically. However, to retrieve the relevant informative web pages for any health related queries, the user has to visit various links and manually look at the web contents. It needs manual intervention and time consuming.

Web page classification is the method of categorizing web pages into various domains such as Health, Sports, Shopping etc. Various machine learning techniques are applied to perform the web page categorization task. The content based classification methods are not effective, as the web page contents are dynamic in nature. To overcome this limitation, URL based approaches are suggested in the literature [15-19] for classifying the web pages based on URL features. The use of token based features and n-gram features are explored in [5]. To identify the health related web pages, an SVM based approach is suggested in [10]. In this approach, the features were derived from URLs alone and an automatic feature learning method is suggested by applying SVM. Various URL features such as lexical, host based and visual related features have been applied for detecting malicious URLs. But, these features are not useful for classifying the web pages based on the topic of the web page. Though the application of machine learning approaches have been studied by various researchers, the advantage of deep learning methods are not explored much. In this paper, an attempt is made to employ Convolution Neural Network (CNN) for URL based web page classification that can assist in Health Information Search. CNN is considered as one of the powerful tools since it is capable of capturing local relations in the temporal and hierarchical structure. Also, this algorithm has been applied successfully in many image classification problems. It was observed that when multiple layers are applied to model sentences, CNN was able to capture high level abstract feature. Use of ConvNet model for understanding text from scratch [4] is established. And, with minor tweaks it can also be used for other tasks such as named entity recognition, parts of speech tagging and even learning from symbolic systems. CNN can be applied in both word level and character level. The word level feature extraction using CNN [14] shows that the number of unknown words is very high which says that these words do not appear in the vocabulary used for embedding. Thus, character level features [6] can be taken in place of word level which will not require the knowledge of “words” for text classification. Character level CNN extracts useful information from raw input signals without any linguistic syntactic and/or semantic structure knowledge. The major advantage of taking-in character level input is that the model doesn’t require understanding of words, phrases, syntax or semantics, and can be used on any language because, number of characters are common in any application. In this work, we explore the character level Convolutional Neural Network for learning the effective URL features for classifying the health related web pages. The character level ConvNet with filters (2, 3, 4, 5) and their combination are applied and the performance was analysed. Various experiments have been conducted on the benchmark data collection ODP (Open Directory Project). By experimental analysis, it was shown that, an F1 score of 83% was achieved with this Character level Convolutional Neural Network. The paper is structured as follows: In Section II, related works in deep learning and text classification is presented; Section II details the proposed methodology; in section III, the experimental details are discussed followed by discussion about the obtained results; and the concluding remarks along with the future directions are presented in Section V.

Manuscript published on 30 March 2019.

***Correspondence Author(s)**
R. Rajalakshmi, School of Computing Science and Engineering, Vellore Institute of Technology, Chennai, India.
S. Ramraj, School of Computing Science and Engineering, Vellore Institute of Technology, Chennai, India.

II. RELATED WORKS

In [1], the performance of few models/architectures of Deep Neural Network (DNN), Recurrent Neural Network (RNN) & Convolution Neural Network (CNN) in natural language processing tasks are discussed. In recent studies, it has been found that CNNs with grating can perform better than LSTMs (Long Short Term Memory) in tasks such as language modeling for which RNNs are used conventionally.Experiments shows that, when the sentiment is determined using the entire sentence and not by few key phrases, Gated Recurrent Unit (GRU), a standard form of recurrent neural network gives better results than CNN. Also, DNN model performs better in text classification tasks, depending on how often the comprehension of global/long-range semantics is required. CNN outperforms RNN in tasks where few key phrases are used in prediction; whereas in tasks where global meaning needs to be considered, RNN or more specifically GRU outperforms any other DNN model. Thus, we found that RNN performs well in broad range of NLP tasks except for the tasks which depends on few key phrases. In [5], an effective first level filter using URL-based classifier and rejection framework was proposed for web page classification. Feature extraction from URL can be done in 3 ways - 1.considering each part as individual tokens; 2.considering n-grams (n=3 to 8) of each part; 3.considering n-gram (n=3 to 8) of the concatenated URL. Irrelevant features are discarded by checking the goodness score using CHIR method. Use of rejection framework avoids excessive misclassification by passing the URLs with confidence scores less than a threshold value onto the next stage for further preprocessing. They used ODP DMOZ dataset with 1.22 million URLs of 13 different categories. Use of statistical feature along with Naïve Bayes gives a better F1 score. Results with n-grams are better than token based or partial matching methods. But the best results are obtained by using n-grams on concatenated URLs The authors of [2][4] have used two Convnets with six convolution layers and three fully connected layers each, to get an encoded character level input, which is then quantized to get an encoded representation having sparse representation. The model outperformed both bag-of-words and bag-of-centroids via word2vec models with ease on multiple datasets and multiple NLP tasks. The above mentioned model, when used to perform text classification, focuses mostly on words and less on character level. ConvNets do not require actual words or even their semantics and/or meanings to work. The input language can be any media such as images or sounds. They perform well with user generated and curated data. At the scale of millions of records, they exponentially outmatch the performance of traditional models. In [4], the author evaluated the performance of ConvNet with DBPedia dataset and Yahoo! Answers. DBPedia dataset consists of 560,000 training dataset with 14 classes. With this dataset, Large Conv Net [6] topped with around 98% test accuracy. Yahoo! Answers classification dataset consists of 1.4 million training samples with 10 main categories. For this dataset, large ConvNet performed with 87% test accuracy. Machine learning algorithms [6] are applied for URL classification. Various features used for training the network should be continuously updated for getting the high true positive value. In [6] malicious URL, File path, and Registry keys are classified using character level ConvNets and the result shown has 0.1 False Positive rate. We have used similar implementation which is used in [6]. The approach is similar to 'n' gram but the weights are approximated in the beginning and updated on semantically similar substring. 19,067,879 unique URLs are labeled using scores given by an antivirus engine like Virus Total. If 5 or more antivirus engine identifies a URL as malicious, it is marked as malicious. The baseline method compared with the character level CNN is 'n' gram (1-5). The features like URL length, number of separators, categorical lexical features like domain name, URL suffix token are manually extracted. The large feature size is hashed to 1024 dimension. Expose character level ConvNet [6] achieves 6% higher detection rate of malicious URL than n-gram. The problem with machine learning is the feature engineering and multiple layer of passes for classification. In [7] combination of convolution neural network and recurrent neural network layers are used for obtaining best features for malware classification**.** Convolution layers uses sequences of n gram to minimize the loss of information. The output of CNN is fed to the recurrent layers called LSTM. This model achieved an 85% precision. In [9] the features of the given image are extracted using the convolution neural network and it is delivered to the SVM classifier. The SVM classifier is introduced into the deep learning neural network architecture by taking hinge loss and using a linear optimizer. But in our work, we extracted the features using the three fully connected layers and trained the SVM separately with the extracted features. In [19], transfer learning approach has been applied to detect malicious domain that are generated by DGA.

III. PROPOSED METHODOLOGY

The task of finding the relevant health related web page is a time consuming task, as the web contains information on various topics for different applications. In particular, classifying the web pages using the URLs alone is a challenging task as the identification of suitable features is highly important. In this research work, the suitable URL features are learnt using Convolutional Neural Network. As the URL is composed of different characters and may not have meaningful words, a character level approach is preferred for this task. The character n-grams that are derived from URLs are found to be useful for URL representation. To obtain the good representative URL features the character combination can be learnt from the training data by using Convolutional Neural Network. Convolutional Neural Networks have been widely used for image classification task. In this paper, an attempt is made to apply CNN for URL classification task. The proposed architecture of CNN is shown in Figure 1, which has three convolution layers and three fully connected layers. In the pre-processing step, the stopping characters such as full-stop, forward slash, comma and semicolon are removed. They are not necessary, as we consider characters as features, and not the words.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

As the average number of characters for any URL in our training set is found to be 100, we have used 100 as maximum length. All the characters in the URL is represented using a 100 dimensional array with one-hot encoding. If the length of URL is more than 100 after pre-processing those characters are not taken into consideration. Similarly, if the number of characters in any URL is less than 100, it is padded to the maximum length. The next step is to represent the URL into a matrix of the form n1xn2, where n1 is the number of characters present in the URL and n2 is the embedding dimension. We have chosen the value of n2 to be 32. Then every URL in the training set is represented in this form. To learn the n-gram features from this URL collection, various filter sizes are chosen. To study the effectiveness of different n-gram features, we have chosen 2, 3, 4 and 5 as the filter size. The combination of these sizes were also been tried. To achieve this, we have applied 4 different convolutions. With these four convolutions , different features are learnt. It was followed by ReLU activation function. To capture the most significant features, max pooling can be applied and to retain all the information, sum pooling can be applied. The purpose of pooling is to reduce the number of parameters and the computation of many parameters. The output from the pooling layer is given as input to the fully connected layers. The pooling operation reduces the size of spatial representation and it results in a M dimensional vector. Then a drop out of 0.5 has been applied. Finally, the combination of these filters is given as input to the fully connected layer followed by a softmax layer. The features learnt using CNN are presented in Figure 2. Now output of the convolution layer will be the input for the dense layer. In our experiment we used three fully connected layers or dense layers. These three layers will try to generalize the features learned from the convolution layer. The output layer is also a dense layer with sigmoid activation function which will give a probability score for the given input. By applying various optimizers, such as Adam and Adadelta, the performance of the CNN models are studied. To combine the advantage of two optimizers and to study the impact of optimizers on the classification performance, we have proposed an ensemble approach. In this ensemble approach, both the models are combined together in the last layer. The experimental details are presented in the next section.

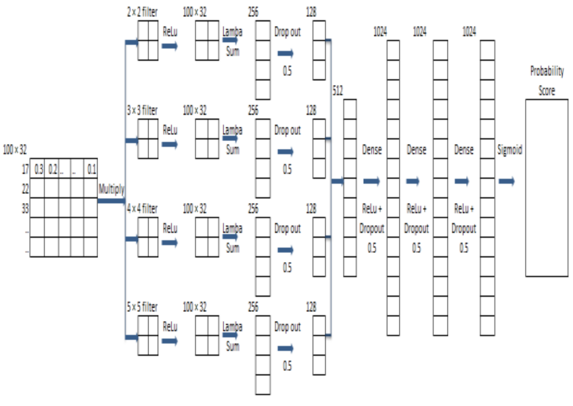
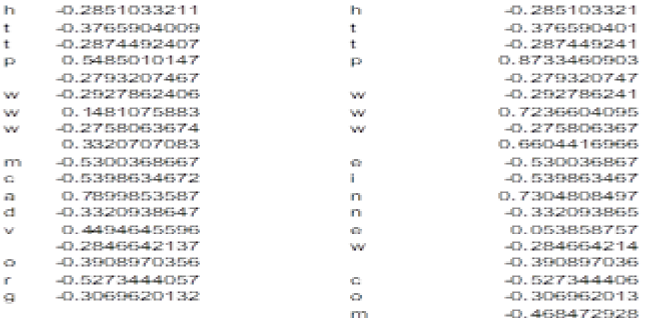


Fig. 1. The CNN architecture of URL Classifier



h	-0.2851033211	h	-0.285103321
t	-0.3765904009	t	-0.376590401
t	-0.2874492407	t	-0.287449241
p	0.5485010147	p	0.8733460903
	-0.2793207467		-0.279320747
w	-0.2927862406	w	-0.292786241
w	0.1481075883	w	0.7236604095
w	-0.2758063674	w	-0.275806367
	0.3320707083		0.6604416966
m	-0.5300368667	e	-0.530036867
c	-0.5398634672	i	-0.539863467
a	0.7899853587	n	0.7304808497
d	-0.3320938647	n	-0.332093865
v	0.4494645596	e	0.053858757
	-0.2846642137	w	-0.284664214
o	-0.3908970356		-0.390897036
r	-0.5273444057	c	-0.527344406
g	-0.3069620132	o	-0.306962013
		m	-0.468472928

Fig 2. Sample Character Level Features Learned

IV. EXPERIMENTS

The experimental details are presented in this section with the result analysis. All the experiments in this study were conducted on the Dell workstation with Intel Xeon Quad Core HT 3.5 GHz processor with 32GB RAM**.** *We have collected the data from ODP dataset and considered only Health category for this work. In this collection, 80% was used for training and the remaining 20% was kept aside for testing.*

We have conducted various experiments using the character level CNN model with the following objectives:

- to find the optimum number of convolutions
- to find the suitable optimizer
- to study the significance of different pooling layer
- to study the effectiveness of ensemble technique

In general, for URL classification, character n-gram based approach was preferred and the combination of n-grams were found to be useful [ ]. The number of characters considered for representing the URL has higher impact on the classification performance. In this CNN based approach, we have tried to apply the filters viz., 2, 3, 4, 5 and their combination. The experimental results are shown in Table 1 and it is observed that an F1 measure of 0.77 was achieved. To find the optimum filter size, various combinations was also tried. From Table 1, it could be observed that, the combination of 3, 4, 5 yielded the highest performance with an F1 measure of 0.82. Though, we tried to apply another combination with 2, 3, 4, 5, the performance of this combination is not better than the 3-4-5 combination. Hence, from the above experiment, it is concluded that, the optimal number of convolutional filters are the combination is 3, 4 and 5. The second set of experiment was carried out to find the suitable optimizer. As the number of convolution is fixed to the combination of 3-4-5, we have tried the Adam optimizer on this setting. Adam (Adaptive Moment Estimation) is found to be better than Stochastic Gradient Descent algorithm, as it is computationally efficient and has less memory requirements. Also, it is well suited for the problems with large data. The URL based approach that could be used to assist in health information search has lot of data and it is growing day by day. In order to address this issue, Adam optimizer was chosen. To reduce spatial size of the representation, that is to reduce the amount of parameters and computation in the network and to reduce overfitting, pooling layer was added in the ConvNet.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

To retain the important information, max pooling was applied and the results were analyzed. We have also conducted the second experiment with sum pooling, in which all values are considered and the average is taken. From the experiments, it is observed that max pooling with Adam optimizer performs better than sum pooling with an F1 measure of 0.82. This model (Adam with max pooling) is referred as Model 1. In order to find the suitable optimizer, the third experiment was conducted with Adadelta and max / sum pooling combination. We could not observe a very significant improvement in the results and an F1 measure of 0.821 was obtained for this Adadelta optimizer with max pooling (referred as Model 2). To combine the advantages of both the optimizers, an ensemble approach has been suggested and the fourth experiment was carried out. In this experiment, the output of model 1 and model 2 are combined together and given as input to form the ensemble layer. The performance of this ensemble CNN approach is found to be better than individual CNN models. For this ensemble approach, we have achieved an F1 measure of 0.83. The results of these experiments are presented in Table 2.

TABLE 1. PERFORMANCE OF CNN USING DIFFERENT FILTERS

Filters	F1
2	0.7752
3	0.8031
4	0.8118
5	0.8024
2,3	0.8052
3,4	0.8102
4,5	0.8202
3,4,5	**0.8216**
2,3,4,5	0.8202

TABLE 2. PERFORMANCE OF CNN WITH DIFFERENT OPTIMIZERS

MODEL	F1
Char_CNN_Model-1 (ADAM)	0.820
Char_CNN_model-2 (AdaDelta)	0.821
Ensemble Model	**0.831**

RESULTS AND DISCUSSION

The results of the various models have been summarized in Table 2. The results clearly indicates that the character level ensemble approach model has overcome the over fitting problem that was present in the individual CNN models. The training accuracy was around 90%, while the test accuracy was around 81%. By merging the outputs of two individual CNN models, this problem has been eliminated. To study the effectiveness of the proposed ensemble approach, 5-fold cross validation was performed. We have obtained an average training accuracy of 85.7% and validation accuracy of 84%. From this best set of ensemble approach, which is a combination of Adam and Adadelta, we are able to obtain 83% test accuracy. From the results presented in the Table 3, we could observe that the performance of the ensemble model is better than the other models and it outperforms all the other techniques.

We compared the performance of the proposed character level CNN with other existing works. For the comparison purpose, the results reported for health related URLs are alone taken into consideration. The summary of the comparison is presented in Table 3.

TABLE 3. COMPARISON WITH EXISTING APPROACHES

	3-grams [16]	*All-grams [15]*	*Proposed method*
F1	78.7	82.0	83.1

In [16], they used only 3-grams for classifying URLs and achieved 78.73 %, whereas in [15] all-grams (4 to 8) were used and they have achieved 82%. But in our ensemble CNN approach, we extracted features learnt from CNN and used it for URL classification. This approach has improved the performance and we have achieved an F1 score of 83.1%.

V. CONCLUSION

In this research work, an URL based design has been suggested to ease the task of health information search. The content based methods are not suitable, as it is time consuming and does not reflect the dynamic changes in the web. So, the performance of Convolutional Neural Network model with character level input for URL classification. We also applied ensemble technique by combining two CNN models with different optimizer and pooling layers and analyzed the performance against those of the individual models. From the experiments, we conclude that the ensemble model gives an improved F1 of 83% compared to the individual models. In future, this approach will be extended by considering the word level features to enhance the performance of the proposed approach.

ACKNOWLEDGMENT

We would like to thank the Department of Science and Technology - Science and Engineering Research Board, Government of India for funding this work (Award Number : ECR/2016/000484). We would also like to thank the management of Vellore Institute of Technology (VIT – Deemed to be University), Chennai for extending their support, where this research work was carried out.

REFERENCES

1. W Yin, K Kann, Mo Yuand and H Schute "Comparative Study of CNN and RNN for Natural Language Processing",in CoRR Feb 2017
2. X Zhang, J Z Y LeCun, "Character-level Convolutional Networks for Text Classification", in CoRR, Feb 2016
3. X Zhang, J Z Y LeCun, "Text Understanding from the scratch", Feb 2015
4. Schmidhuber, "Deep learning in neural networks: An overview" in Science Direct April 2014
5. Rajalakshmi R, Aravindan C, 2018,''A Naive Bayes approach for URL classification with supervised feature selection and rejection framework", Computational Intelligence, 34(1), pp. 363-396.
6. Joshua S, K Berlin "eXpose: Character-Level Convolutional Neural Network with Embeddings For Detecting Malicious URLs, File Paths and Registry Keys" in CoRR Feb 2017
7. B Kolosnjaji, Apostolis Z, G Webster, and C Eckert "Deep Learning for Classification of Malware System Call Sequences" in Australasian Conference on Artificial Intelligence (2016).
8. W. Huang, J. W. Stokes "A multi-task neural network for dynamic malware classification. In Detection of Intrusions and Malware, and Vulnerability Assessment" in Springer International Publishing, 2016.

9. A Fred M. Agarap "An Architecture Combining Convolutional Neural Network (CNN) and Support Vector Machine (SVM) for Image Classification" in arXiv:1712.03541 Dec 2017.

10.R. Rajalakshmi, 2015, "Identifying Health Domain URLs using SVM", Third International Symposium on Women in Computing and Informatics (WCI – 2015), ACM. DOI:http://dx.doi.org/10.1145/2791405.2791441

11.Araque, Oscar and Corcuera-Platas, Ignacio and Snchez-Rada, J. Fernando and Iglesias, Carlos A. "Enhancing Deep Learning Sentiment Analysis with Ensemble Techniques in Social Applications" in Expert Syst. Appl. July 2017

12.Tianqi Chen , Carlos Guestrin "XGBoost: A Scalable Tree Boosting System" in ACM 2016

13.R. Rajalakshmi, C. Aravindan, 2018, An Effective and Discriminative Feature Learning for URL Based Web Page Classification, in Proceedings of IEEE International Conference on Systems, Man, and Cybernetics (SMC), Miyazaki, Japan, 2018, pp.1374-1379.

14.DOI: 10.1109/SMC.2018.00240

15.Hung Le, Q Pham, Doyen Sahoo, Steven C.H. Hoi."URLNet: Learning a URL Representation with Deep Learning for Malicious URL Detection" .In Proceedings of ACM Conference, July 2017

16.Rajalakshmi and C. Aravindan. Web page classification using n-gram based URL features. In Fifth International Conference on Advanced Computing, ICoAC 2013, Dec 18-20, Chennai, India

17.2013. IEEE.

18.Rajalakshmi, R., Xaviar, S., 2017, Experimental Study of Feature Weighting Techniques for URL Based Webpage Classification,Procedia Computer Science, Vol.115, pp. 218-225

19..Kan M-Y, Thi HON. Fast webpage classification using URL features. Technical Report. Singapore: National

20.University of Singapore; 2005.

21.Singh N, Sandhawalia H, Monet N, Poirier H, Coursimault J-M. Large scale URL-based classification using

22.online incremental learning. In: Proceedings of the 2012 11th International Conference on Machine Learning

23.and Applications (ICMLA), Vol. 2; IEEE Computer Society; 2012; Washington, DC

24.Rajalakshmi, R., Ramraj, S., Ramesh Kannan, R., 2019, Transfer learning approach for identification of malicious domain names, Communications in Computer and Information Science, Vol. 969, pp. 656-666. DOI: 10.1007/978-981-13-5826-5_51.