INLI@FIRE-2018: A Native Language Identification System using Convolutional Neural Networks

Ajees A P[1][0000−0002−4960−1865] and Sumam Mary Idicula[2]

[1] Research Scholar, CUSAT, Cochin 682022, INDIA
ajeesap87@gmail.com
[2] Professor, CUSAT, Cochin 682022, INDIA
sumam@cusat.ac.in

Abstract. Native Language Identification is the problem of identifying the first language of speakers based on his/her writings in another language. The proposed approach is a deep learning based methodology using convolutional neural networks. Convolutional neural networks are a class of neural networks that have proven very effective in areas such as pattern recognition and classification. They are able to capture the local texture within the text and can be used to find the representative patterns in a text document. The proposed system consists of a language identification model, which is trained by a corpus of 1233 documents. The experiments were conducted using the dataset provided for INLI@FIRE2018. The results indicate that the system is capable of giving performance comparable to the methods employing more sophisticated approaches.

Keywords: Convolutional Neural Networks · Native Language Identification · Natural Language Processing

1 Introduction

Native Language Identification is the process of distinguishing the native language of a writer from his/her writings in the second language(English) [2]. It is a well-known task that finds important applications in fields like forensic, educational settings, etc. Native language is always used as an essential feature for authorship profiling and identification. Nowadays, due to the enormous usage of social media sites and online interactions, getting an intense threat is a common issue faced by commuters. If a comment or post induces any type of threat, then recognizing the native language of the commenter(the one who commented/posted it) will be one of the crucial measures in finding the source. Speakers of various languages may make different types of errors when learning a new language [10]. Hence Native Language Identification finds its applications in educational environments to supply targeted feedback to language students about their errors.

Hindi is by far the most widely spoken language in India. Even though roughly 40% of the population speak Hindi, people use English as their major second language. English is spoken natively by around 375 million people across the globe. It is the second official language of India and is used for business, teaching, learning, and trade on a day to day basis. Around 10% of Indias population speak English and use it in their day to day activities. But it is only a first language for 0.019% people in the Country and becoming a second language for around 125 million people all over the world [1]. This 10% of the population is from different parts of the country and have various native languages. Identification of the native language of such speakers is a challenging task that finds important applications in this social media world.

The structure of this paper is as follows. Section 2 briefly reviews the similar works in this area. Section 3 discusses the task description and details about the dataset. Section 4 explains the methodology and Section 5 demonstrates the results and evaluation metrics. Section 6 concludes the article along with some routes for the future works.

2 Related works

Native Language Identification has a lot of importance in different areas of Natural Language Processing [3]. Most of the works in NLI is reported by taking English as a second language. They treated NLI as a supervised classification task and used statistical models to train data from various languages. The first work in the field is reported by Koppel et al. [9] who explored a multitude of features for NLI. These features include average sentence length, average word length, word n-grams, character n-grams, POS n-grams, content words, function words, spelling errors, grammatical errors, etc. SVM was used to train these features on ICLE corpus(International Corpus of Learner English (ICLEv2)[7]). Unigrams and Bigrams are the most explored n-grams in the previous works.

Syntactic features of the text are also focussed on the recent works. Wong and Dras [14] used production rules from different parsers as features to Language identification system. Similarly, Swanson and Charniak [12] investigated the benefit of Tree Substitution Grammars for NLI. Tetreault [13] experimented the use of Tree Substitution Grammars along with dependency features extracted from the Stanford parser. Tree fragments returned from Tree Substitution Grammar were proved to be beneficial for distinguishing the native and non-native English writers by acquiring the syntactic structures. Similarly, augmenting CFG rules with the grandparent nodes and the augmented rules are found to be outperforming the simple CFG rules in authorship attribution tasks [5].

It has been found that the semantic features are the least experimented one for NLI. Gamon extracted semantic features from semantic dependency graphs[6]. These features include binary semantic features and semantic modification relations which are used as a feature set for classification purpose. Semantic features contain number and gender information of nouns and pronouns as well as tense and aspectual features of verbs. Similarly, semantic modification relations extract the semantic relations between a node and all its descendants within a semantic graph. Experiments showed that the semantic features in combination with the syntactic features resulted in improved accuracy for Authorship Classification tasks [6]. Throughout the literature, we have found that none of the existing works utilizes deep learning based methodologies for language identification tasks. Hence we decided to go for an approach which uses CNN for the above-mentioned problem.

3 Task Description and Dataset Details

The task is focused on identifying the first language of an author from the given Text/XML file which includes a set of Facebook comments in the English language. Six Indian languages are considered for this study. They are Tamil, Hindi, Kannada, Malayalam, Bengali, and Telugu. Spoken forms of English shows significant variations across the different states of India and it is relatively easy to recognize the native language of the speaker using his English accent. But finding the first language of a writer based on his comments or posts in English is a difficult task in the present scenario.

The shared dataset contains data from six different Indian languages. The training data is a set of files in XML format. Each language has around 200 files of facebook comments. Each file contains around 150 words as the comment. Sentence segmentation is carried out using the regular expression. Statistics of the training data is shown in Table 1. The testing data contains two folders say test1 and test2. Test1 consists of 783 files and test2 contains 1185 files from the above-mentioned languages.

Table 1. Training data statistics

Language	# xml files	# sentences	# words
Hindi	211	1688	28983
Tamil	207	1656	34606
Malayalam	200	1600	47167
Telugu	210	1680	49176
Kannada	203	1624	45738
Bengali	202	1616	37623

4 Proposed Method

The proposed system is a CNN-based language identification model which predicts the native language of a writer from his scripts. CNN's are responsible for the important breakthroughs in Image Classification problems and are the core of the most Computer Vision systems today. But they are not common in text analytics. CNN's have been proved to be successful in various text classification problems in recent years [8]. They have an important property of preserving the 2D spatial orientation in computer vision problems. But when it comes to texts these orientations have a one-dimensional structure. A generalized overview of Convolutional Neural Networks is shown in Fig. 1.

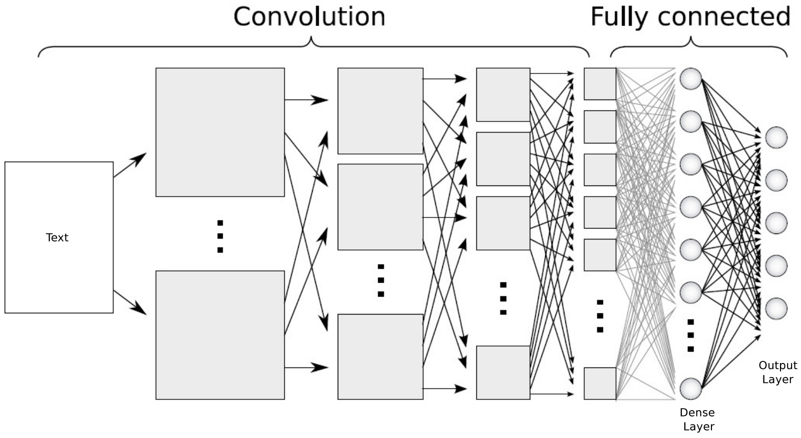

Fig. 1. CNN: A generalized overview

The problem is shaped as a text classification task with language names as labels(classes). The number of classes is the same as the number of languages considered for the study. Text from each language in the training data is sent to a sentence segmentation module where the raw text is converted to a set of sentences using regular expressions. Providing sequences of raw human-alike words will make no sense to computers. For that reason, the raw words are converted into numeric values using dictionaries. For that, we create a vocabulary of words, an array which stores all the words in the training data, but each word appears only once. Two dictionaries which map from word to its corresponding index value and reverse are also created. Two special words-'ZERO' and 'UNKNOWN' are added to the dictionary. 'ZERO' is used to make all the sequences of unique length and 'UNKNOWN' is used for out of vocabulary words. Then the sequences of strings are converted into sequences of numbers using the aforementioned dictionaries. The sentences may have a different length. But CNN training requires sequences of uniform length. So we padded the sentences with less number of words with 'ZEROS' to make them of unique length(ZERO padding). That is why we added the word ZERO to the dictionary. Each sentence in the training data is labeled with a corresponding language label. Hence our final training data contains a lot of sentences and their corresponding labels. Identifying the patterns within the sentences is our ultimate goal.

Sequential model of keras is used for implementation [4]. The network is designed with four convolutional layers, two max-pooling layers, two dense layers, and an embedding layer. The first layer is an embedding layer which performs the word embeddings. The embedding size is fixed at 100. The second one uses the convolutional layer for its ability to capture the local context. The following layers are alternate max-pooling and convolutional layers for acquiring the patterns within the sentence. We have used 'Relu' as the activation function to bring the nonlinearity. The number of filters used in all the convolutional layers is 256. And the kernel size is fixed at 7 for the first two convolutional layers and 3 for the remaining layers. The final dense layer is associated with softmax activation units. During the training phase, filters slide over full rows of the matrix(words). CNN automatically learns the values of its filters based on the task assigned to it. The architecture of the proposed network is shown in Table 2.

Table 2. Configuration of the CNN architecture

Layers	Output shape	Configuration
conv1d_1	40×256	$7 \times 1, strides1$
max_pooling1d_1	13×256	$3 \times 1, strides1$
conv1d_2	13×256	$7 \times 1, strides1$
max_pooling1d_2	4×256	$3 \times 1, strides1$
conv1d_3	4×256	$3 \times 1, strides1$
conv1d_4	4×256	$3 \times 1, strides1$
flatten_1	1024	–
dropout_1	1024	0.5
dense_1	128	Relu activation
dropout_2	128	0.5
dense_2	6	Softmax activation

Different configurations of the network are attempted. Experiments are conducted using deep and shallow convolutional neural networks. The performance of different CNN architectures on the test data is given in Table 3. The best results are given by the above-described architecture. In our experiments, we selected the first 90% of the data as training data and the remaining data as testing data. The batch size is fixed at 64. 'Categorical cross entropy' is used as the loss function. We used 'Adam', the efficient gradient descent algorithm as the optimizer because it is an efficient one for optimization. Dropout is used to prevent overfitting [11]. Model is compiled using Tensorflow in the backend. The network is trained for 10 epochs and the model file is saved for the testing purpose.

5 Results

Experiments are also conducted to measure the effect of training data size on the system performance. It is observed that the performance of the system increases

Table 3. Comparison of different CNN architectures

Name	Network Configuration	Accuracy
CNN1	1 Conv,1 Maxpool,1 Dense,1 Dropout	17.5%
CNN2	2 Conv,2 Maxpool,2 Dense,1 Dropout	21.2%
CNN3	3 Conv,2 Maxpool,2 Dense,2 Dropout	22.2%
CNN4	4 Conv,3 Maxpool,2 Dense,2 Dropout	25.7%
CNN5	4 Conv,4 Maxpool,2 Dense,2 Dropout	25.3%
CNN6	5 Conv,5 Maxpool,2 Dense,2 Dropout	24.7%

with the increase in training data size. Figure 2 shows the effect of training data size on our best performed CNN architecture. Hence it is better to have a larger sized training corpus when dealing with deep learning based classification methodologies. We used accuracy to quantify the performance of our model. Accuracy computes the degree to which the result of a prediction conforms to the true value. The proposed system was tested with the test data sets provided by the task organizers. Our system predicts the tag for each sentence in the post(comment). But our goal is to predict the tag for each XML file(post). So we labeled each post according to the maximum number of predictions for that particular post. Table 4 demonstrates the results of our experimentation on both the datasets. It is clear from the table that test2 results are far better than test1 results. Different runs correspond to different architecture of the proposed network.

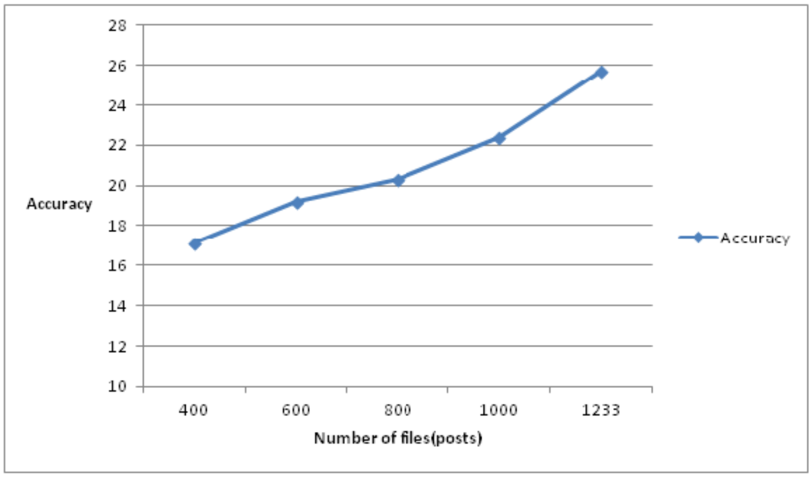

Fig. 2. Effect of training data size on system performance

Table 4. Results

Run File submission	Accuracy
Ajees_TestSet1_run1.txt	14.0%
Ajees_TestSet1_run2.txt	15.2%
Ajees_TestSet1_run3.txt	10.7%
Ajees_TestSet2_run1.txt	24.1%
Ajees_TestSet2_run2.txt	20.9%
Ajees_TestSet2_run3.txt	21.8%

6 Conclusion

In this article, we have discussed a deep learning based native language identification system. The exclusive feature of our approach is the use of the Convolutional neural network for this task. The main reason we preferred a CNN rather than traditional feature-based methods is its ability to capture local texture in a sequence. It has been found that the accuracy of the system increases with the increase in training data size. Hence it is better to have a larger sized training corpus to get improved performance. The accuracy of the system can also be improved by using trained word embeddings. Due to insufficient system requirements, we could not perform this activity. Apart from NLI, Convolutional Neural Networks can be applied efficiently for various language processing applications. We hope to apply CNN based methods to different language processing applications such as text classification, sentiment analysis, etc.

References

1. The top 10 most spoken languages in india. https://www.listenandlearnusa.com/blog/the-top-10-most-spoken-languages-in-india, accessed: 2018-08-03
2. Anand Kumar M, B.G.H., P, S.K.: Overview of the inli@fire-2018 track on indian native language identification. In: workshop proceedings of FIRE 2018, FIRE-2018, Gandhinagar, India, December 6-9, CEUR Workshop Proceedings (2018)
3. Anand Kumar M, Barathi Ganesh HB, S.K.P., Rosso, P.: Overview of the inli pan at fire-2017 track on indian native language identification. In: Notebook Papers of FIRE 2017, FIRE-2017, Bangalore, India, December 8-10, CEUR Workshop Proceedings (2017)
4. Chollet, F., et al.: Keras. https://github.com/fchollet/keras (2015)
5. Feng, S., Banerjee, R., Choi, Y.: Syntactic stylometry for deception detection. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2. pp. 171–175. Association for Computational Linguistics (2012)
6. Gamon, M.: Linguistic correlates of style: authorship classification with deep linguistic analysis features. In: Proceedings of the 20th international conference on Computational Linguistics. p. 611. Association for Computational Linguistics (2004)
7. Granger, S., Dagneaux, E., Meunier, F., Paquot, M.: International corpus of learner english (2009)

8. Kim, Y.: Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882 (2014)
9. Koppel, M., Schler, J., Zigdon, K.: Determining an author's native language by mining a text for errors. In: Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining. pp. 624–628. ACM (2005)
10. Smith, B.: Learner English: A teacher's guide to interference and other problems. Ernst Klett Sprachen (2001)
11. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research **15**(1), 1929–1958 (2014)
12. Swanson, B., Charniak, E.: Extracting the native language signal for second language acquisition. In: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 85–94 (2013)
13. Tetreault, J., Blanchard, D., Cahill, A., Chodorow, M.: Native tongues, lost and found: Resources and empirical evaluations in native language identification. Proceedings of COLING 2012 pp. 2585–2602 (2012)
14. Wong, S.M.J., Dras, M.: Exploiting parse structures for native language identification. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. pp. 1600–1610. Association for Computational Linguistics (2011)