LANGUAGE IDENTIFICATION FROM TEXT

R Akshay[1], T Vinay[2], J Aditi[3]

[1,2,3]SR Engineering College, Warangal, Telangana, India

[a)] Corresponding author: [a)] akshayramagiri55@gmail.com,
[b)] 19k41a0559@sru.edu.in
[c)] 19k41a04A0@sru.edu.in

Abstract. As the world is progressing in terms of trends and technologies, we need to update ourselves with those trending technologies. Artificial Intelligence is one such most popular and advancing technology which is going to rule the world in future. AI is used to make the humans work easy without any hurdles. AI almost entreched into each and every sector such as Health care, Finance, Transportation etc. Being an Indian we may knew only few of the native languages like Telugu, Hindi, English etc. Whenever we visit a new place it is very difficult to understand the language that is spoken by the people and the text written on direction boards etc. So we thought of developing an artificial intelligence application that helps us to identify the language of the text. The application takes the text as input and identifies the languages present in that text. It is also used in case if multilingual documents. If a document consists of different language text by using our application one can easily find the languages present in that document. In order to develop this application we have used a dataset from kaggle and build a model that helps us to identify the language of the text in the document. The model performed well with an accuracy of 99% and 82.5% validation accuracy in detecting the languages from the text.

Keywords: Artificial intelligence, Language detection, Multilingual document.

INTRODUCTION

In the present indigenous world, each and every one are running behind the new advancing technologies in order to make their daily chores simpler. Artificial Intelligence (AI) is one such advancing technology that gained a lot if consideration in the present world. It is a part of computer science that focuses on designing intelligent computer systems that show the traits we re-late with human intelligence like comprehending languages, learning problem-solving, decision making, etc. One of the significant contributions of AI has remained in Natural Language Processing (NLP), which glued together linguistic and computational techniques to assist computers in understanding human languages and facilitating human-computer interaction. Machine Translation, Chat bots or Conversational Agents, Speech Recognition, Sentiment Analysis, Text summarization, etc., fall under the active research areas in the domain of NLP. However, in the past few years, Sentiment analysis has become a demanding realm.

Nowadays, Artificial Intelligence has spread its wings into Thinking Artificial Intelligence and Feeling Artificial Intelligence (Huang and Rust 2021). Thinking AI is de-signed to process information in order to arrive at new conclusions or decisions. The data are usually unstructured. Text mining, speech recognition, and face detection are all examples of how thinking AI can identify patterns and regularities in data. Machine learning and deep learning are some of the recent approaches to how thinking AI processes data. AI has made a big impact on the globe. AI was reintroduced in a significant manner in the twentieth century, and it inspired researchers to perform in-depth studies in domains like NLP, and machine learning. However, the domains of NLP remain ambiguous due to its computational methodologies, which assist computers in understanding and producing human-computer interactions in the form of text and voice.

Language detection is one such area in which we use AI specifically Natural Language Processing (NLP) techniques to make the language identification task much simpler. We have collected a huge dataset consisting of 16 different languages text. Dataset consists of two columns those are text in different languages and the corresponding language name. As most of the people don't know multiple languages. An ordinary person knows at most two to three languages. So whenever such person went to different place where the native language of that place is different from the language the person knew. Then the person may face a difficulty in identify and understanding the text written on road side posters, sign boards etc. So in order to overcome this draw back we came up to develop an application which identifies the language of the text.

LITERATURE REVIEW

Although there are some dedicated survey articles, these tend to be relatively short; there have not been any comprehensive surveys of research in automated LI of text to date. The largest survey so far can be found in the literature review of Marco Lui's PhD thesis [1] which served as an early draft and starting point for the current article. Zampieri provides a historical overview of language identification focusing on the use of ngram language models. Qafmolla gives a brief overview of some of the methods used for LI, and Garg, Gupta, and Jindal [5] provide a review of some of the techniques and applications used previously.

Shashirekha [6] gives a short overview of some of the challenges, algorithms and available tools for LI. Juola [7] provides a brief summary of LI, how it relates to other research areas, and some outstanding challenges, but only does so in general terms and does not go into any detail about existing work in the area. Another brief article about LI is Muthusamy and Spitz [9], which covers LI both of spoken language as well as of written documents, and also discusses LI of documents stored as images rather than digitally-encoded text.

There have been several NLI studies published in the past few years. Due to the availability of suitable language resources for English (e.g. learner corpora), the vast majority of these studies dealt with English [1] (Brooke and Hirst, 2012; [2] Bykh and Meurers, 2014), however, a few NLI studies have been published on other languages. Examples of NLI applied to languages other than English include Arabic [4] (Ionescu, 2015), [5] Chinese (Wang et al., 2016), and Finnish [6] (Malmasi and Dras, 2014).

To the best of our knowledge, the NLI Shared Task 2013 [7] (Tetreault et al., 2013) was the first shared task to provide a benchmark for NLI focusing on written texts by non-native English speakers. A few years later, the 2016 Computational Paralinguistics Challenge (Schuller et al., 2016) included an NLI task on speech data.

The NLI Shared Task 2017 combines these two modalities of non-native language production by including essays and spoken responses of test takers in form of transcriptions and iVectors. The combination of text and speech has been previously used in similar shared tasks such as the dialect identification shared tasks organized at the VarDial workshop series [8].

Cologne-Nijmegen's TF-IDF-based approach (Gebre et al., 2018) competed in the DSL shared task 2015 [8] as team MMS ranking among the top 3 systems. A variation of NRC's SVM approach [9] competed in the DSL 2014 (Goutte et al., 2014) achieving the best results. Bobicev applied Prediction for Partial Matching (PPM) in the NLI shared task [10] with results that did not reach top ten performance. A similar improved approached competed in the DSL 2015 ranking in the top half of the table.

A similar approach to the one by Jarvis [11] that ranked 1st place in the NLI task 2013 competed in the DSL 2017 (Bestgen, 2017), achieving the best performance in the competition. Variations of MQ’s SVM ensemble approach have competed in the DSL 2017 [11] and the ADI 2016, achieving the best performance in both shared tasks.

PROBLEM STATEMENT

Most of the people know only few languages but whenever they visit a new place they face a major problem in identifying the lanugage that is written in the road side boards etc. And in a multilingual document it is very difficult to identify the language of the text.

DATA INSIGHTS

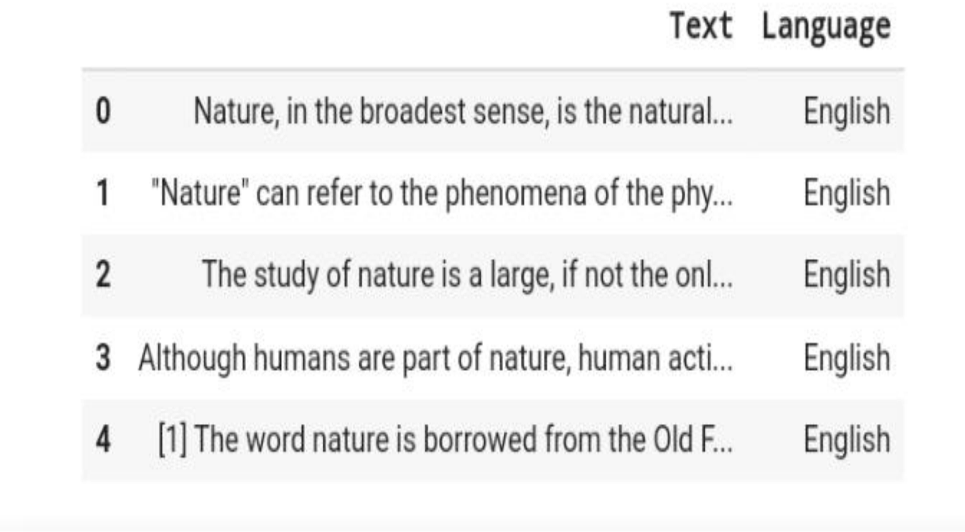
We have used the Language Detection dataset, which contains text details for 16 different languages.

Languages are:

* English
* Portuguese
* French
* Greek
* Dutch
* Spanish
* Japanese
* Russian
* Danish
* Italian
* Turkish
* Swedish
* Arabic
* Malayalam
* Hindi
* Tamil

By using the text in the dataset we developed an application that is able to identify the language from the given text sample. Using the text we created a model which will be able to predict the given language. This is a solution for many artificial intelligence applications and computational linguists. These kinds of prediction systems are widely used in electronic devices such as mobiles, laptops, etc for machine translation, and also on robots. It helps in tracking and identifying multilingual documents too. The domain of NLP is still a lively area of researchers.

Data set was obtained from open source Kaggle website. Data set contains 2columns, Text, and Language. Text column had the text in different languages and the language column has the corresponding language name.

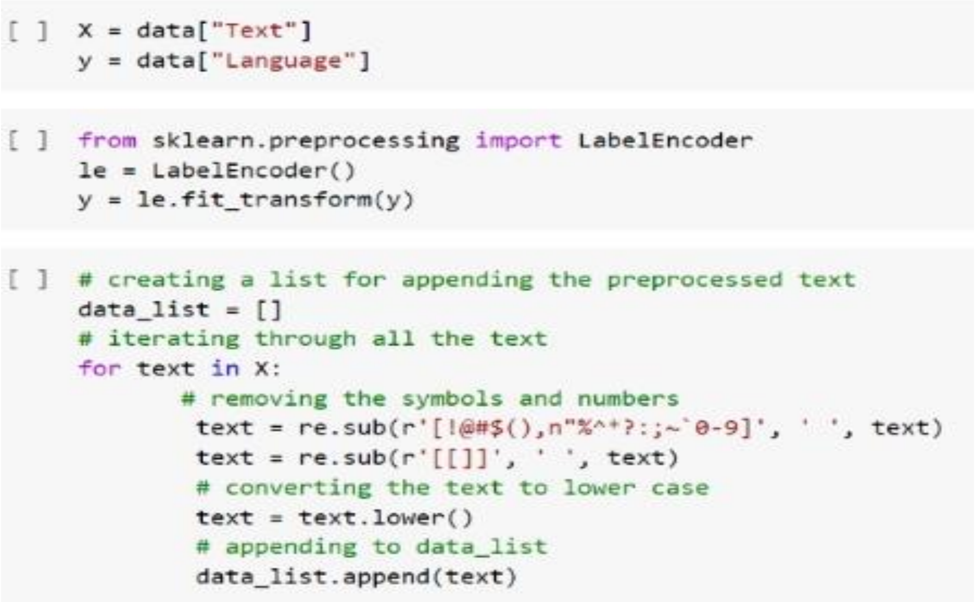


	Text	Language
0	Nature, in the broadest sense, is the natural...	English
1	"Nature" can refer to the phenomena of the phy...	English
2	The study of nature is a large, if not the onl...	English
3	Although humans are part of nature, human acti...	English
4	[1] The word nature is borrowed from the Old F...	English

Figure 1. Dataset insights

Figure 1 gives the information about the dataset. The dataset consists of 2 columns one for the text and other for the language name. The dataset consists of 17 different languages texts.

DATA PRE-PROCESSING



```
[ ]  X = data["Text"]
     y = data["Language"]
```

```
[ ]  from sklearn.preprocessing import LabelEncoder
     le = LabelEncoder()
     y = le.fit_transform(y)
```

```
[ ]  # creating a list for appending the preprocessed text
     data_list = []
     # iterating through all the text
     for text in X:
             # removing the symbols and numbers
              text = re.sub(r'[!@#$(),n"%^*?:;~`0-9]', ' ', text)
              text = re.sub(r'[[]]', ' ', text)
              # converting the text to lower case
              text = text.lower()
              # appending to data_list
              data_list.append(text)
```

Figure 2. Data pre-processing

Figure 2. depicts the data preprocessing steps involved. Firstly the raw data is treated for null values removal after that the special characters and numerical digits are removed from the text and finally the text is converted into lower case letters.

Steps involved in data preprocessing:

- Drop rows with Null (NA) values
- Drop NA may cause inconsistency in index so reset indexes
- Remove special characters in text
- Remove numerical data from the text
- Convert into lower case letters
- Remove stop words

Split dataset:

Firstly split the dataset into features and target variable, then by using the train_test_ split method, split the data into a training set and testing set. The test_size = 0.20 that is 20% of data for testing and remaining 80% for training purpose.

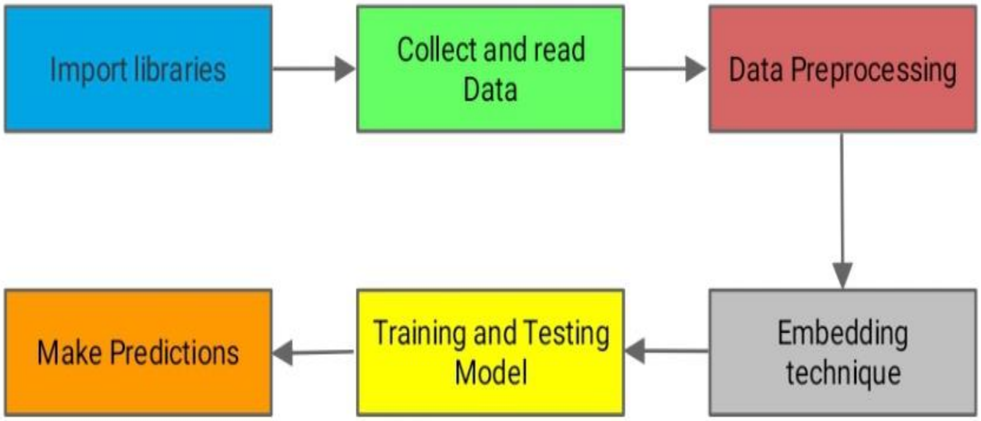


Figure 3. Flow chart

Figure 3 depicts the process work flow of our project. Firstly we need to import all the required libraries. Then we need to collect and tea the dataset into the program. Followed by data preprocessing which includes removal of special characters, removal of numerical data, and followed by removal of stop words. The data after removing stop words is called Bag Of Words. Then those BOWs are converted into vectors by using embedding techniques and then followed by training and testing the model. Finally the model is ready for predictions.

METHODOLOGY

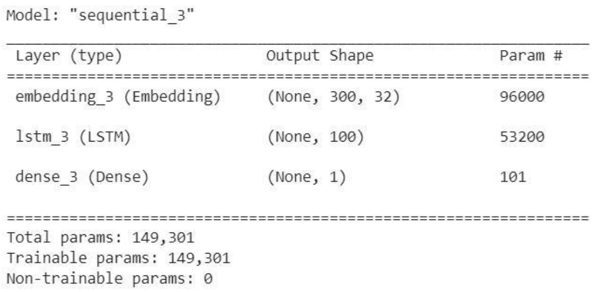
Firstly we have collected the dataset which consists of 16 different languages. Then we observed that their is an imbalance in the data set so we Ade the balanced dataset which consists of 16 different languages. Then we applied data preprocessing techniques to the dataset. Firstly the special characters and numerical data are removed then the dataset is converted into lower case then after the data is sent to embedding process, where the sentences are converted into vectors.

In our Proposed Model we are using GloVe embedding technique and LSTM. GloVe stands for Global Vectors for word representation, is an unsupervised learning algorithm for obtaining vector representations for words. Training is performed on aggregated global word-word co-occurrence statistics from a corpus, and the resulting representations showcase interesting linear substructures of the word vector space. The proposed model consists of data collection and data preprocessing. After preprocessing the data by removing stop words, the corpus is passed to glove embedding layer and converted to one hot representation and finally it is passed to CNN Model which uses LSTM.

We created a NLP model by using GloVe embedding technique and LSTM. Firstly, we have imported the required libraries and read Csv file. We then performed missing value treatment by dropping columns with NA values. Removed special Characters. numerical data and Stop words and then added glove word embedding layer it aims to generate word embedding's by aggregating global word occurrences matrices from corpus. Performed One hot representation for input and initialized our model to sequential () and used a CNN architecture for LSTM. Added embedding layer that can be used for neural networks on text data. It requires that the data to be integer encoded, so that each word is represented by unique value.it is initialized with random weights. Added dropout layer. It is a regulation technique where randomly selected neurons are ignored during training and added LSTM layer. We added dense layer with sigmoid activation function.

Steps:

1. Import the required libraries
2. Read the dataset as .csv file
3. Perform missing value treatment by dropping columns with NA values
4. Remove special Characters from the text
5. Remove numerical values from the text
6. Remove Stop words from the text
7. Add bert or glove word embedding layer it aims to generate word embedding's by aggregating global word occurrences matrices from corpus.
8. Perform One hot encoding representation for input
9. Initializing model to sequential()
10. Adding embedding layer that can be used for neural networks on text data. It requires that the data to be integer encoded, so that each word is represented by unique value.it is initialized with random weights.
11. Add Long Short Term Memory (LSTM) layer
12. Adding dense layer with sigmoid activation function



```
Model: "sequential_3"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_3 (Embedding)     (None, 300, 32)           96000

 lstm_3 (LSTM)               (None, 100)               53200

 dense_3 (Dense)             (None, 1)                 101

=================================================================
Total params: 149,301
Trainable params: 149,301
Non-trainable params: 0
```

Figure 4. Model Summary

Figure 4 depicts the model summary of our model. The model consists of different layers and we have used LSTM in our project.

RESULTS AND ANALYSIS

Our model performed very well in identifying the language of the text from a multilingual text document.

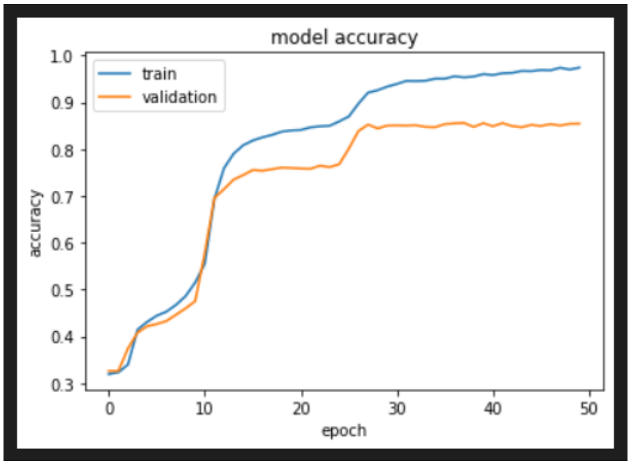


Figure 5. Model Accuracy Graph

Figure 5. Show the accuracies of our developed model. The model gave an accuracy of 99% and 82.53% of validation accuracy.

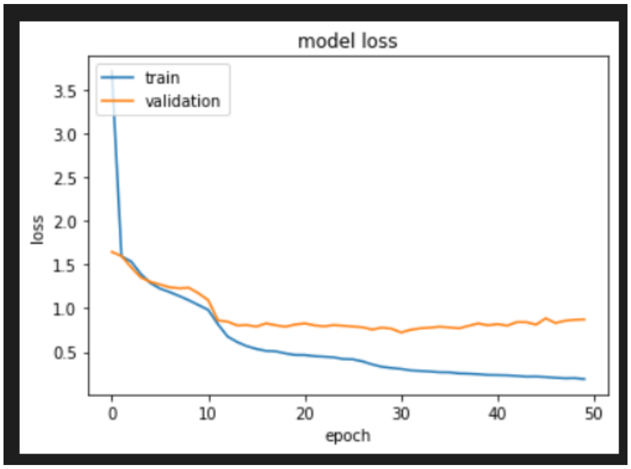
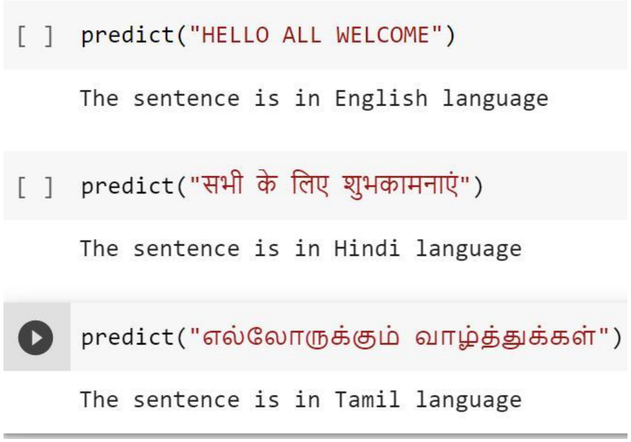


Figure 6. Model Loss Graph

Figure 6 depicts the model loss graph of the developed model. The model performed well in identifying the language of the text from the given input text with minimum loss.



```
[ ]  predict("HELLO ALL WELCOME")

     The sentence is in English language

[ ]  predict("सभी के लिए शुभकामनाएं")

     The sentence is in Hindi language

     predict("எல்லோருக்கும் வாழ்த்துக்கள்")

     The sentence is in Tamil language
```

Figure 7. Predictions

Figure 7 shows the predictions made by the model. The model performed very well in predicting the language of the given input text accurately.

CONCLUSION

In India we have a large number of native languages like each ad every state has its own native language. But most of the people knew only few languages apart from their mother tongue. So the Language identification/detection gained an importance. So that we developed an application that is used to identify the language name from the given input text. It is very useful to identify the language names in a multilingual document. The model performed very well in identifying the languages with an accuracy of 99% and 82.53% validation accuracy. In future the application can be developed to identify and translate the text from one language to the other language.

ACKNOWLEDGEMENTS

We thank S R University (Formerly S R Engineering College) Warangal, India for providing us immense support during the project.

REFERENCES

1. Julian Brooke and Graeme Hirst. 2020. Measuring Interlanguage: Native Language Identification with L1-influence Metrics. In Proceedings of Language Resources and Evaluation (LREC). pages 779–784.
2. Serhiy Bykh and Detmar Meurers. 2019. Exploring Syntactic Features for Native Language Identification: A Variationist Perspective on Feature Encoding and Ensemble Optimization. In Proceedings of COLING 2019, the 25th International Conference on Computational Linguistics: Technical Papers. Dublin, Ireland, pages 1962–1973.
3. Serhiy Bykh, Sowmya Vajjala, Julia Krivanek, and Detmar Meurers. 2018. Combining shallow and linguistically motivated features in native language identification. In Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications. Atlanta, Georgia, pages 197–206.
4. Radu Tudor Ionescu. 2020. A Fast Algorithm for Local Rank Distance: Application to Arabic Native Language Identification. In Proceedings of the International Conference on Neural Information Processing. Springer, pages 390–400.
5. Lan Wang, Masahiro Tanaka, and Hayato Yamana. 2019. What is your Mother Tongue?: Improving Chinese Native Language Identification by Cleaning Noisy Data and Adopting BM25. In Proceedings of the International Conference on Big Data Analysis (ICBDA). IEEE, pages 1–6.
6. Shervin Malmasi and Mark Dras. 2018. Language identification using classifier ensembles. In Proceedings of the VarDial Workshop.

7. Joel Tetreault, Daniel Blanchard, and Aoife Cahill. 2018. A Report on the First Native Language Identification Shared Task. In Proceedings of the Eighth Workshop on Building Educational Applications Using NLP. Atlanta, GA, USA.
8. Shervin Malmasi and Marcos Zampieri. 2016. Arabic Dialect Identification in Speech Transcripts. In Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3). Osaka, Japan, pages 106–113.
9. Cyril Goutte, Serge L´eger, Shervin Malmasi, and Marcos Zampieri. 2016. Discriminating Similar Languages: Evaluations and Explorations. In Proceedings of Language Resources and Evaluation (LREC).
10. Victoria Bobicev. 2019. Native language identification with ppm. In Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications. Atlanta, Georgia, pages 180–187.
11. Scott Jarvis, Yves Bestgen, and Steve Pepper. 2020. Maximizing classification accuracy in native language identification. In Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications. Atlanta, Georgia, pages 111–118.