Malayalam Error Sentence Detection using Deep Learning with RNN-LSTM

Supriya L P, Chinchu M S

Abstract: ***Malayalam is a difficult Indian language and not easy for foreigners. Even as a mother tongue, you should spend more time learning as a child. In Malayalam, the use of the wrong word in a sentence may change the whole meaning and purpose. Many errors occur during the writing process. It is very difficult to find errors in the Malayalam language, and no one can remove those errors without their linguistic knowledge In this paper, we have proposed the structure of repetitive neural networks using long short-term memory (RNN-LSSTM) to detect Malayalam sentence errors.***

***Index Terms*: Deep Learning, Natural Language Processing, RNN, RNN-LSTM**

I. INTRODUCTION

Automatic text error detection is one of the main research areas of natural language processing. In-depth learning is the foundation of artificial intelligence, and deep learning is the frontier family of machine learning methods, which can be the basis of learning representation. Deep learning is an in-depth structured learning or hierarchical study that trains the computer to perform human-like tasks such as speech recognition, image recognition, or prediction.

Deep learning is also known as feature learning or representational learning, which is a set of machine learning algorithms that attempt to learn multiple layered models of the main input inputs of natural language processing. Deep neural networks involve many levels of non-linear operations. One of the main reasons for going deeper is that deep neural networks can more efficiently represent a nonlinear function.

Transfer learning is the benefit of deep learning. Transfer learning is a learning algorithm and different learning tasks for transferring knowledge across different tasks. In-depth learning algorithms commonly used are repetitive neural network (RVNN), repetitive neural network (RNN), convolutional neural network (CNN), and deep generative networks.

Natural Language Processing is a sequence of algorithms and techniques that mainly focus on teaching computers to understand the human language. Natural Language Processing tasks include document classification, text classification, translation, para-phrase identification, text similarity, summarization, and question answering.

The development of natural language processing is a challenging process due to the complexity and ambiguity of human language. In-depth learning methods have recently been able to demonstrate several attempts to achieve high accuracy in natural language processing tasks. Most of the natural language processing models follow a similar preprocessing step. The first decomposes the input text into words through tokenization, and the second reproduces these words in the form of vectors or n-grams. Word embedding is one of the main applications of Natural Language Processing.

II. LITERATURE SURVEY

This section will provide a detailed study of the various grammar examiners present. Grammar checking is a program for checking written text for grammatical accuracy. Grammar checkers are often implemented as one of the features of a program such as word processor, and are available as an application that can be activated from programs that run on editable text. Existing grammar checking programs for punctuation and style inconsistencies instead of the full range of possible grammatical errors. One of the first systems was called Writer's Workbench, and it contains a set of input tools embedded in UNIX systems in the 1970s. Later the area was technically advanced.

A. **Statistical Grammar Checker**

In a grammar test using a corpus [1]. The corpus is maintained from different journals, magazines, and documents. This quality of being very close to the truth or true number of the series of words that make sense and that have a subject and a verb series of words that make sense and that have a subject and a verb) using the corpus. There are two ways to check the input text. The series of words that make sense and that have a subject and a verb matches the input text, and it is tagged as grammatical errors, otherwise the legal punishment/time spent punished is true or false. The second way is that the maintenance corpus creates some rules and checks the input phrase using these rules. There is no update to the rules when maintaining the corpus or adding new data. There are some bad results or effects to these approaches, which can make it very hard to find the error in the series of words that make sense and that have a subject and to identify the error in the system.

.B. Rule Based Grammar Checking

The most commonly used approaches are rule-based grammar checking. In rule-based

Revised Manuscript Received on July 22, 2019.

Supriya L P, Assistant Professor, Dept. of computer science & Engineering, Sree Buddha College ofEngineering,Pathanamthitta, Kerala

Chinchu M S , Research Scholar , Dept. of computer science & Engineering, Anna University

grammar checking [1], It examines words that are meaningful and a subject and verb using rules that come from the dataset. Rules are created manually with respect to the approach to numbers. In the rule-based approach, it is easy to set rules and change these rules. An important advantage of this approach is that someone who does not have a programming language can handle the rules, and it also provides a detailed error message. The main features of this approach are that it deals with all the features of the language, and the sentences need to be completed and the input sentence can be easily misunderstood.

C. Hybrid Grammar Checking

The blended approach [1] combines rule-based and grammar testing. This makes it more strong and healthy and accomplishes or gains with effort higher wasting very little while working or producing something.

D. English Grammar Checker

Microsoft has developed some grammar checks including the word processor's English language. There is a lot of English grammar testing research. One of these grammatical checking (for truth) experiments is by Daniel Naber [2]. The goal of English grammar testing is to design a grammar and style checker. Grammar checker and style checker can be used as standalone and word processor systems Grammar checker and style checker take a range of words that are meaningful, a subject and a verb as input text and provide a list of possible errors. Its speech tag name, the verb, the word describing an adjective, the word describing an adjective or an adjective, and a strong formal verbal decision about what is meaningful and the subject and verb are determined. Many of them have different parts of speech tags depending on their big picture.

The English grammar test has 54 predefined grammatical rules, which are a series of symbols that must be matched. After verification, the text matches these predefined error rules. English grammar checking is an extension of grammar and style checker using dependency separation and analysis. The English grammar test found 42 errors. When errors are independent of one another, the grammar and style examiner identifies different errors in the sequence of words that are meaningful and have a subject and verb. Such errors cannot be found in the MS word, and multiple errors cannot be specified in an array of words that have the meaning and the subject and verb in the MS word.

E. Afan Oromo Grammar Checking

The Afan Oromo grammar test [3] is one of the most researched for language in Ethiopia. This grammar test is based on a rule-based approach. This grammar checker uses 123 different rules to identify errors that occur during a given law. The Afan Oromo grammar test consists mainly of five parts. The first is symbolic, which is divided into a series of words that make sense and have a subject and verb. The second is part of the conversation, which determines each word into a tag. The third one is the steamer, which accepts the tagged words and determines the route and connects to the tagged word. These three steps are also used to remove certain types of roots (things that are different). The fourth is the grammatical finder, which calculates the grammatical relationships between each word. These terms include subject, verb, subject, word that describes a noun, the main verb, and the verb under tension.

The terms work well in terms of rules. Rules take on the roots and attachments that speak in a vague way, i.e. check the agreement between words. This grammatical information presents the roots and connected rules. The fifth and final instruction is a series of similar words that make sense and have a subject and verb if an error occurs. The Afan Oromo grammar test, based on the rule-based approach, examined the number of errors and the number of errors correcting the system.

F. Portuguese Grammar Checker

COGROO [4] is one of the most advanced grammar testers. It is a Portuguese grammar detector based on Brazilian Portuguese change related to rules forming the language corpus. The person who works to find information is designed to deal with certain issues, including the small amount of the name and the oral agreement, the small amount of the use of the name, and the verbal government, the use of the words that describe the names. Describe Nouns. Portuguese grammar checking contains some rules set in the system. They are local laws, and they put together something and make it a strong law. These two error checkers are checked for local and keep something together and it makes strong errors. Local laws contain a lot of word rules, Something that involves a lot of intricate rules that can be interconnected and tight-lipped, which includes a very small amount and a verbal agreement, a name that describes a very small amount of verbal government, verb and word. Check the tagged words into a word that describes a noun and a verbal word that is meaningful, using a subject and verb checker. Finally, Grammar Relation Finder identifies the relationship between name and verbal terms, which have meaning and grammatical roles, including subject, object, and verb.

III. PROPOSED SYSTEM

A. System Architecture

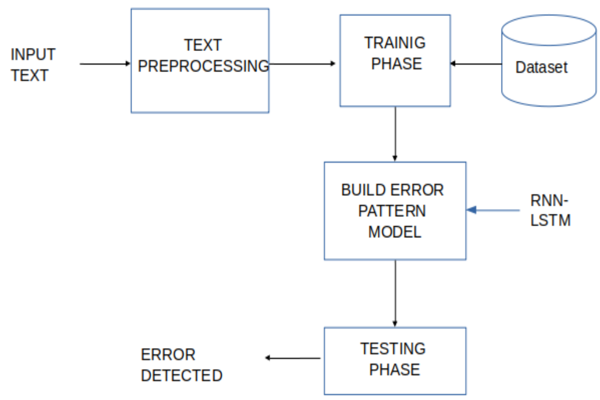


Fig-1: Proposed System Architecture

Automatically detecting error sentences is an important research area. In the paper, they suggested a mechanism for finding the correctness or in-depth study of Malayalam text or text. Natural language processing is one of the deep learning concepts for understanding human language. A mechanism for determining whether the sentence is correct or not is proposed here. In-depth study is divided into several categories based on application. Deep learning algorithms are repetitive neural networks.

System architecture is essentially three to two phases. They are the training phase and the testing phase. The input of the system is a sentence or text, which provides text preprocessing. There are mainly three steps involved in text preprocessing. They include tokenization, noise removal and normalization. Tokenization is the conversion of an input sentence into words or tokenize a sentence. Also known as tokenization text segmentation or lexical analysis.

The next preprocessing step is normalization, which converts the input text to the same case, removes the punctuation, and converts the digits to their words. The next preprocessing step is to remove the noise, such as removing text file headers and captions. After preprocessing, the dataset was prepared. Here, the prepared dataset contains a single Malayalam phrase with labeled data. So it uses a supervised deep learning algorithm. Then go through the training phase. In the training phase, we train a single neural network model labeled Dataset, and after the successful completion of the training phase build an error sentence classification model. The next step is the testing phase. During the testing phase, the model is tested using the test data and finally the accuracy of the tested data.

IV. RNN –LSTM

Recurrent neural networks (RNNs) are powerful and robust neural networks and include the most promising algorithm out there right now, because they only have internal memory. RNN [5] is used to process continuous information. The input of the neural network is always in vector form because it does not support the sentence in the form of words. So it converts the input to binary representation or machine language. One of the most commonly used binary encoded functions is hot encoded. Encoded text is the input of the network. RNN is widely used for natural language processing tasks. It has the capability to perform inherent features, and it performs continuous processing using modeling units.

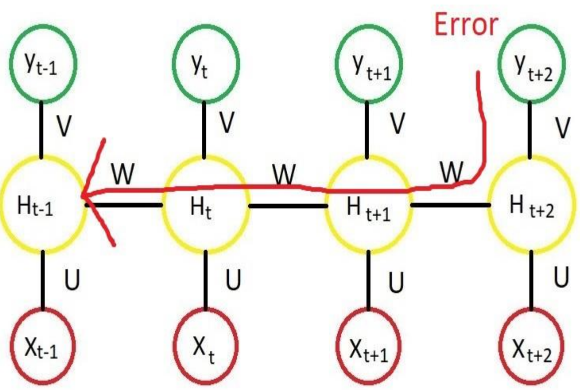


Fig 2: - Error occur at recurrent layer

RNN has all feedback loops in recurrent layers. RNN does not maintain previous node information. It is therefore difficult to solve problems because the gradient of the loss function is also called the vanishing gradient problem. To solve this problem, use LSTM (Long Short Term).

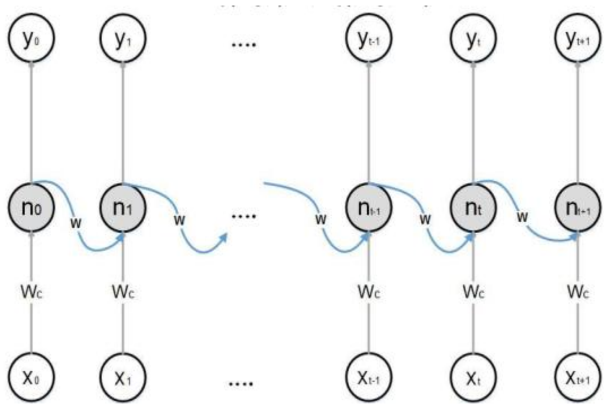
RNN-Long Short-Term Memory (LSTM) is a specific, repeating nerve-related/brain-related network used in deep learning. LSTM has reactions helpful returned information connections for the general computer. LSTM is also a type of RNN that uses special units. LSTM units include a 'memory cell' that can store information for a long time in memory.

LSTM consists of some of the key factors that control how much the value flows into the cell, how long the value stays in the cell and the value of the cell. GRUs are similar to LSTMs, but use a simpler structure. It also contains some gates that control the flow of information.

Fig-3: The sentence input in RNN-LSTM

V.CONCLUSION

Many applications of existing natural language processing, including grammar editing, word processors, and text generation. In this paper, we present a structure of a recurrent neural network - whether or not long-term short-term memory is correct for finding sentences. RNN is a deep learning algorithm that provides continuous information. Here, RNN - LSTM is used, which stores additional memory information on the node.



REFERENCES

1. Nivedita S. Bhirud, R.P. Bhavsar and B.V. Pawar"A SURVEY OF GRAMMAR CHECKERS FOR NATURAL LANGUAGES."
2. Daniel Naber. "A Rule-Based Style And Grammar Checker". Diplomarbeit. Technische Fakultät Bielefeld, 2003.
3. Tesfaye, Debela. "A Rule-Based Afan Oromo Grammar Checker". Jimma Institute of Technology. Ethiopia: Vol. 2, No. 8, 2011.
4. Kinoshita, Jorge; Nascimento, Laнs do; Dantas ,Carlos Eduardo. "CoGrOO: a Brazilian Portuguese Grammar Checker based on the CETENFOLHA Corpus". Universidade da Sгo Paulo (USP), Escola Politйcnica. 2003.
5. LR Medsker and LC Jain. 2001. RECURRENT NEURAL NETWORKS.

Published By:
Blue Eyes Intelligence Engineering & Sciences Publication

AUTHORS PROFILE

Prof. Supriya L P. has more than 13 years of experience in teaching, Research and industry. She completed her after-graduation in Computer Science from Madras University in 2003. She received her M.Phil. From the department of computer Science in 2007, Annamalai University (focused on doing one thing very well) in image processing. She received her Master of Engineering (M.E) degree from School of Figuring out/calculating, Sathyabama University, Computer Science and Engineering in 2009. Now she is chasing after her PhD. She started her career as a teachers/professors of Computer Science in 2004 at Chennai. She has got some (books, magazines, etc.) in (meetings to discuss things/meetings together) and Journals national/international.
College of Engineering, Kerala, India.

Asst.professor **Chinchu M S** the Bachelor's Degree in Computer Science from Anna University, Chennai in 2012 and also completed Mtech in Computer Science and System Engineering from Government Engineering, Idukki in 2016. She has 2 years of teaching and 2 years of industrial experience. Now she is pursing Ph.D from Annamalai University, Chennai.she has more than 5 years of experience in teaching, Research and industry .

Retrieval Number: B11580782S619/2019©BEIESP
DOI:10.35940/ijrte.B1158.0782S619

Published By:
Blue Eyes Intelligence Engineering & Sciences Publication