International Journal of Engineering Technology and Management Sciences[IJETMS]

Website: ijetms.in Issue:4, Volume No.2, November-2018
DOI: 10.46647/ijetms.2018.v02i04.004

Designing a Framework for Automated Essay Grading using Recurrent Neural Networks and Natural Language Processing: A Systematic Review

Dr J.Keziya Rani
Asst.professor, Dept.of.CS&T,S.K.University,Anantapuramu

Abstract: In the educational system, grading is critical in determining student achievement. Individual assessments are used in the present evaluation system. As the ratio between teacher and student is increasing regularly, the human (manual) evaluation system becomes complex. The disadvantage of manual supervision is that it takes more time and is less reliable. This online examination technique originated as a replacement for pen and paper-based approaches. The current computer-based assessment method only operates for multiple-choice questions (MCQs); there is no mechanism for evaluating essays or short replies. Until recently, grading an essay by taking into account all characteristics such as the relevance of the material to the topic, progression of thoughts, coherence, and logic has been a significant difficulty. Many academics concentrated on content-based evaluation, whereas many others concentrated on style-based interpretation. After conducting a thorough literature study and reviewing the Artificial Intelligence (AI) and Machine Learning (ML) algorithms used to assess autonomous essay grading, the limits of current studies and research trends were examined. After implementing LSTM (Long Short Term Memory – a type of Recurrent Neural Networks (RNN)) and Natural Language Processing (NLP), we obtained testing accuracy of 90% of the generated system, which may be raised by raising the frequency of epochs and incorporating more layers / nodes to the present system. The experimentations are done in python using Tensorflow and Keras Libraries.

Keywords: Automatic Essay Grading. Recurrent Neural Network, Long Short Term Memory, Natural Language Toolkit

1 Introduction

The old learning process in educational system has been turned into an online educational system as a result. All the educational institutions such as schools, colleges and universities become accustomed the online education system. The evaluation is critical in determining a student's cognitive capacity. Most computerized tests are available in MCQs, true/false, match the following, and other formats. However, evaluating short and essay responses remains a challenge. The education system is shifting to an online approach, such as computer-based tests and computerised grading. It is a crucial educational application that employs NLP and ML methods. Essay assessment is unachievable using basic coding technologies and approaches like as pattern matching and language processing. The issue here is that for a single question, we will have many replies from students, each with a distinct rationale. As a result, we must examine all of the responses to the question.

The appropriateness of the content to the question, development of ideas, cohesion, clarity, and topic expertise should all be considered when assessing essays and short answers. The correctness of the grading system is defined by the proper examination of the factors indicated above. However, all of these characteristics cannot serve an equivalent significance in essay and short answer grading. Domain knowledge is necessary in a short response assessment, such as the difference between the definitions of "cell" in physics and biology. And, while judging essays, concepts must be implemented in relation to the prompt. The system should also evaluate the replies' completeness and give comments.

2. Deep Learning

DL is a subset of machine learning (ML) that predominantly uses artificial neural networks (ANN). DL is a sort of imitation of the human brain because NN are made to emulate it. Everything doesn't have to be intuitively programmed in DL. The DL phenomena is not new. It has been in existence for some time. Because we didn't have as much data and processing power back then, it's more common now. DL and ML have emerged as a result of the dramatic rise in computer complexity over the past 20 years. The technical term for DL is neurons. An illustration of a single neuron from the human brain, which contains approximately 100 billion neurons. Thousands of its partners are connected to each neuron.

The process of learning in typical ML is supervised, and the coder must be exceedingly detailed when instructing the algorithm what sorts of things it should search for to determine if a picture includes or does not include an output. This is a time-consuming procedure known as

feature extraction, and the machine's efficiency rate is totally dependent on the coder's competence to precisely specify a feature set. The benefit of DL is that the software creates the feature set without intervention.

Long short-term memory
LSTM is a type of ANN that is used in AI and DL. Unlike traditional feed-forward NN, LSTM includes feedback connections. A RNN of this type may analyse not just single input elements (such as photos), but also complete data streams (such as voice or video). Unsegmented, linked handwritten identification, voice recognition, automatic control, video gaming, and healthcare are all applications of LSTM.

3 Review of Literature

Many researchers conducted studies on Automated Essay Grading Systems (AEGS) and proposed several techniques such as Bayesian Essay Test Scoring System (BESTY), Project Essay Grader (PEG), Intellimetric (ITLM), Automated Essay Scoring (AES) and E-Rater (ER). Researchers employed various emerging technologies like ML, DL, and Deep Neural Networks (DNN) to AEGS.
Shermis et al. (2001) introduced PEG, a linguistic verification system with a connection between human assessors and the engine.
ER was suggested by Powers et al. (2002), ITLM by Rudner et al. (2006), and BESTY by Rudner and Liang (2002). These algorithms employ NLP approaches that concentrate on structure and style to produce an essay score.
Burrows et al. (2015) evaluated AES methods across six components: dataset, NLP approaches, model creation, grading methods, assessment, and model efficacy. A computer-based evaluation process that autonomously rates or marks student replies based on acceptable attributes is known as AES. They did not discuss feature extraction methodologies or problems in feature extractions. Only ML models were discussed in their research.

The following research objectives are identified (RO).

RO1: The various datasets available for research on AEGS.

RO2: The features extracted for the assessment of essays

RO3: The assessment measures provided for assessing model's accuracy.

RO4: The ML and DL techniques used for AEGS implementation.

4 Methodology

Scoring the selected essay from number of essays of students based on the information in it. All data are in the form of CSV file. For scoring essays, used deep learning models like LSTM in RNN. These models are defined are defined in Natural Language Processing. The evaluation is based on grammar, semantics, spellingand other grading information. Linear Regression is used for predicting analysis.

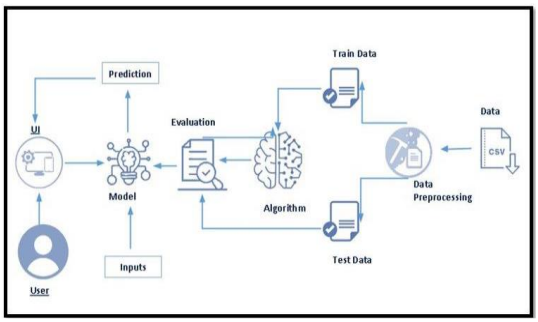

Figure 1. Architecture

TensorFlow (TF) is a complete publicly available ML platform. It features a rich, adaptable set of tools, modules, and community services that enable academics to push the boundaries of ML while programmers can simply design and deliver ML-powered apps. It is a conceptual math toolkit that employs logical data flow and distinguishable computation to handle multiple roles related to DNN training and inference. It enables programmers to build ML models by leveraging a variety of tools, frameworks, and community networks.

The Tensor Framework (TF) accepts inputs in the form of a multi-dimensional array called Tensor, allowing you to design stream processing networks and architectures to indicate how information flows across a network. It allows for the development of a chain of operations that can be performed on these inputs and that start at one end and end with a result at the other.

Keras is built on publicly available machine frameworks like as TF, Theano, and Cognitive Toolkit (CNTK). CNTK is a sophisticated publicly available library introduced by Microsoft that can be leveraged to generate ML prediction

International Journal of Engineering Technology and Management Sciences[IJETMS]

Website: ijetms.in Issue:4, Volume No.2, November-2018
DOI: 10.46647/ijetms.2018.v02i04.004

models. Theano is a Python module for doing rapid mathematical computations. Keras is a simple framework that allows for the creation of DL models using TF or Theano. Keras is intended to let you easily build DL models. Keras is the best framework for data-intensive applications.

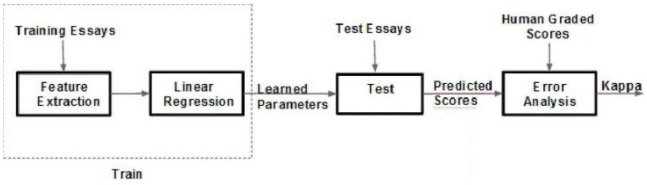

Figure 2. Implementation Methodology

4.1 Data Preparation and Code

The process for getting data ready for a machine learning algorithm can be summarized in three steps.

Step 1: Data Selection – It deals with type of data is available, missing data and the removed data. Dataset is downloaded from kaggle.com

Step 2: Data Pre-processing

Step 3: Data Splitting – with the help of Scikit-learn (SKlearn) library, splitting into training and testing sets by importing the function train_test_split.

```
import numpy as np
import nltk
import re
from nltk.corpus import stopwords
from gensim.models import Word2Vec
```

```
X = pd.read_csv(r'C:\Users\HP\Desktop\Automated-Essay-Grading-main\data\training_set_rel3.tsv', sep='\t', encoding='ISO-8859-1')
```

```
def essay_to_wordlist(essay_v, remove_stopwords):
    """Remove the tagged labels and word tokenize the sentence."""
    essay_v = re.sub("[^a-zA-Z]", " ", essay_v)
    words = essay_v.lower().split()
    if remove_stopwords:
        stops = set(stopwords.words("english"))
        words = [w for w in words if not w in stops]
    return (words)
```

4.2 Algorithm

Used deep neural networks for building the model. The activities include the following steps.

STEP1: Import the model building Libraries

STEP2: Initializing the model

STEP3: Preparing the dataset of inputs and outputs pairs encoding integers.

STEP4: Constructing the LSTM Model

STEP5: Input Layer

STEP6: LSTM Layer

STEP7: Training and testing the model

STEP8: Saving the model

4.3 Constructing the LSTM Model

The objective is to develop a Deep Network by overlaying LSTM layers such that the system learns complicated and lengthy words efficiently. This deep LSTM model may be created by employing variety of packages and procedures such as Keras, TF, Theano, and CNTK. The LSTM is made up of several distinct layers, as well as an activation function and a learning optimizer.

Input Layer: In order to take input sequence, this layer is responsible.

LSTM Layer: It calculates the output using LSTM units and returns hidden and cell states. Initially, 100 units were put to the layer, which may be fine-tuned afterwards.

Dropout Layer: This layer is responsible for regularization which means it prevents over-fitting. This is accomplished by deactivating certain neurons in the LSTM layer.

Output Layer: This calculates the likelihood of our forecast.

```
def get_model():
    """Define the model."""
    model = Sequential()
    model.add(LSTM(200, dropout=0.4, recurrent_dropout=0.4, input_shape=[1, 200], return_sequences=True))
    model.add(LSTM(64, recurrent_dropout=0.4))
    model.add(Dropout(0.5))
    model.add(Dense(1, activation='relu'))

    model.compile(loss='mean_squared_error', optimizer='rmsprop', metrics=['mae'])
    model.summary()

    return model
```

```
# Load glove embeddings
corpus = []
for essay in X['essay']:
    corpus.append(essay_to_wordlist(essay, True))

embedding_dict={}
with open('C:/Users/HP/Desktop/Automated-Essay-Grading-main/data/glove.6B.200d.txt','r', encoding="utf8") as f:
    for line in f:
        values = line.split()
        word = values[0]
        vectors = np.asarray(values[1:],'float32')
        embedding_dict[word] = vectors
```

5 Results and Discussion

Python DL was used for training and testing data. When working with datasets, a DL model operates in two phases, and we normally split the data between testing and training phases by 20 percent to 80 percent. The attained validation accuracy of 90% of the built model may be raised by extending the number of epochs and incorporating more layers/nodes to the present system.

6 Conclusion

AEGS is a very important ML application. It has been examined several occasions, with various approaches such as latent semantic interpretation being used. This present technique attempts to characterize language qualities such as language proficiency, grammatical accuracy, and domain knowledge quality of essays, with the goal of fitting the optimal polynomial basis functions.

The outcomes that can be produced will be both uplifting and genuine. We might be able to attain an average absolute error that is substantially lower than the standard deviation of human scores. The strategy is likely to perform pretty effectively across all fields. The given problem's future scope may enlarge across other domains. One example is the exploration for and modelling of good lexical and pragmatic characteristics. Different semantic parsers and other tools can be employed for this. Another area of research might be to develop a stronger method than linear regression with polynomial basis functions, such as NN.

NLTK, WordVec, and Stopwords NLP packages are used for feature extraction; nevertheless, these packages have significant constraints when translating a phrase into vector form. Aside from feature extraction and building ML models, no tool has accessibility to the totality of the article.

7 Future Scope

Although LSTM is an effective system, it is operationally costly and necessitates the use of GPUs to adapt and build the system. They are now employed in a variety of application scenarios such as voice assistants, smart virtual keyboards and autonomous chatbots, sentiment analysis, and so on. The present LSTM's accuracy can be improved in upcoming studies by adding more layers and nodes to the system and using the concept of transfer learning on the same problem space.

References

[1]. Adamson, A., Lamb, A., & December, R. M. (2014). Automated Essay Grading.

[2].Ajetunmobi SA, Daramola O (2017) Ontology-based information extraction for subject-focussed automatic essay evaluation. In: 2017 International Conference on Computing Networking and Informatics (ICCNI) p 1–6. IEEE

[3]. Basu S, Jacobs C, Vanderwende L. Power grading: a clustering approach to amplify human effort for short answer grading. Trans Assoc Comput Linguist (TACL) 2013;1:391–402. doi: 10.1162/tacl_a_00236. [Cross Ref] [Google Scholar]

[4]. Burrows S, Gurevych I, Stein B (2015) The eras and trends of automatic short answer grading. Int J ArtifIntell Educ 25:60–117. https://doi.org/10.1007/s40593-014-0026-8

[5]. Chen M, Li X (2018) "Relevance-Based Automated Essay Scoring via Hierarchical Recurrent Model. In: 2018 International Conference on Asian Language Processing (IALP), Bandung, Indonesia, 2018, p 378–383, doi: https://doi.org/10.1109/IALP.2018.8629256

[6]. Educational Testing Service (2008) CriterionSM online writing evaluation service. Retrieved from http://www.ets.org/s/criterion/pdf/9286_CriterionBrochure.pdf .

[7]. Foltz PW, Laham D, Landauer TK (1999) The Intelligent Essay Assessor: Applications to Educational Technology. Interactive Multimedia Electronic Journal of Computer-Enhanced Learning, 1, 2, http://imej.wfu.edu/articles/1999/2/04/ index.asp

[8]. Neural Networks for Automated Essay Grading: https://cs224d.stanford.edu/reports/huyenn.pdf

[9]. Automatic Text Scoring using Neural Networks: https://da352.user.srcf.net/publications/acl2016.pdf

[10]. Predicting Grammaticality on an Ordinal Scale: http://aclweb.org/anthology/P/P14/P14-2029.pdf

[11]. Automated essay scoring by maximizing human-machine agreement: http://www.aclweb.org/anthology/D13-1180

Website: ijetms.in Issue:4, Volume No.2, November-2018
DOI: 10.46647/ijetms.2018.v02i04.004

[12]. Task-independent features for automated essay grading: http://www.aclweb.org/anthology/W/W15/W15-0626.pdf
[13]. Automated Essay Scoring with E-rater: https://www.ets.org/Media/Research/pdf/RR-04-45.pdf.
[14]. Ajetunmobi SA, Daramola O (2017) Ontology-based information extraction for subject-focussed automatic essay evaluation. In: 2017 International Conference on Computing Networking and Informatics (ICCNI) p 1–6. IEEE