INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume 5, Issue 2)
Available online at: www.ijariit.com

Answer script evaluator: A literature survey

Sachin B. S.
sachinbet17@gmail.com
Vidyavardhaka College of Engineering. Mysore, Karnataka

Shivaprasad K.
shiva15997@gmail.com
Vidyavardhaka College of Engineering. Mysore, Karnataka

Somesh T.
somyash97@gmail.com
Vidyavardhaka College of Engineering. Mysore, Karnataka

Sumanth Hegde
sumanthhegde1029@gmail.com
Vidyavardhaka College of Engineering. Mysore, Karnataka

Radhika A. D.
radhika.ad@vvce.ac.in
Vidyavardhaka College of Engineering. Mysore, Karnataka

ABSTRACT

Every college, university, school conduct exams and most important part of exams are the results. In order to get these results, the exam papers have to be evaluated one by one manually. This process of evaluating the exam papers is time-consuming and requires more manpower. To overcome this solution, we have come up with a thought that removes the manual evaluation process. Our project focuses on developing a system that evaluates an answer script against a pre-uploaded marking scheme. Initially, the answers are taken in digital format and those digital answers are processed using algorithms such as word2vec where the word's similarity index is extracted and the words similar to it are noted. Using this we can also get the meaning of the paragraph and we can match it with the answer key and get the match percentage. Using this percentage, we can calculate the marks to be awarded.

Keywords— *word2vec, Deep Neural Networks, doc2vec*

1. INTRODUCTION

Just as electricity transformed the way industries functioned in the past century, artificial intelligence the science of programming cognitive abilities into machines has the power to substantially change society in the next 100 years. AI is being harnessed to enable such things as home robots, robot axis and mental health chatbots to make you feel better. Since our main concern is to process the input text we use an algorithm called word2vec. Word2vec is a two-layer neural net that processes text. Its input is a text corpus and its output is a set of vectors: feature vectors for words in that corpus. While Word2vec is not a deep neural network, it turns text into a numerical form that deep nets can understand.

The purpose and usefulness of Word2vec are to group the vectors of similar words together in vector space. That is, it detects similarities mathematically. Word2vec creates vectors that are distributed numerical representations of word features, features such as the context of individual words. It does so without human intervention.

Given enough data, usage and contexts, Word2vec can make highly accurate guesses about a word's meaning based on past appearances. Those guesses can be used to establish a word's association with other words (e.g. "man" is to "boy" what "woman" is to "girl"), or cluster documents and classify them by topic. Those clusters can form the basis of search, sentiment analysis and recommendations in such diverse fields as scientific research, legal discovery, e-commerce and customer relationship management.

2. EXISTING SYSTEM

The current system of evaluation of answer scripts involves manually checking all the answer scripts one by one and evaluating them by looking at the marking scheme. This system of evaluation takes a lot of time and also makes way for uncertain evaluation. It also takes a lot of time and manpower and involves each person to personally evaluate a specific subject answer paper one by one. The constraints in this system are time, manpower and efficiency. Efficiency depends on the number of people evaluating, number of papers, and the type of evaluators (different number of evaluators for different subjects). If a number of papers are huge and the number of evaluators is few, the efficiency is low. In the case of huge universities, this efficiency is low most of the times as there will be a huge number of students in the university and the number of answer scripts is also huge. In some universities, there is a system that uses scanned copies of the answer sheet instead of the physical paper itself. This provides an option to perform valuation digitally but it has to be done by a human.

3. PROPOSED SYSTEM

This system is an extension of digitization of answer sheets. In this system, along with digitization of answer sheets, even valuation is done by the system itself. The system uses deep neural networks that are used to perform extraction of keywords and this extracted text is then compared to the reference text using unsupervised learning. The inputs to this system are digital answer sheets and the reference text.

The main advantage of this system is the reduction of time of valuation drastically and it also diminishes man power. Using this system, any numbers of answer sheets can be evaluated (regardless of what subject they belong to).

This system can also be hosted online and the models to be used can be made available online so that the evaluator can be used in a wide range of devices and can be accessed remotely.

4. RELATED WORK

"NLTK: The Natural Language Toolkit" authored by Steven Bird and Edward. [1]

NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing. WordNet groups English words into sets of synonyms called synsets provides short definitions and usage examples

"A Compare-aggregate Model For Matching Text Sequences" authored by Shuohang Wang and Jing Jiang. [2]

Many NLP tasks including machine comprehension, answer selection and text entailment require the comparison between sequences. Matching the important units between sequences is a key to solve these problems. In this paper, a framework is presented that performs word-level matching followed by aggregation using Convolutional Neural Networks. The focus is mainly on the different comparison functions we can use to match two vectors.

"A Study on Word2Vec" authored by Nina Tahmasebi. [3]

According to this paper, a new way to use word vector to match its meanings is shown. Word2vec takes the input text and converts it to numerical form so that the underlying neural network can understand it. Consider this image in which a word polite is compared to its meanings along with the probability of a match.

"Progress in the Application of Natural Language Processing to Information Retrieval Tasks" authored by Alan F. Smeaton. [4]

Information retrieval is a discipline dedicated to the development of effective means of accessing textual information of any type by using a computer. Different words may be used to convey the same meaning: 'Stomach pain after eating' and 'postprandial abdominal discomfort' mean the same thing. Different people may have different perspectives on the same single concept: 'The accident' v. 'the unfortunate incident' could be describing the same thing. The same words may have different meanings in different domains: sharp can be a measure of pain intensity in medicine or the quality of a cutting tool in a gardener's handbook.

"An empirical evaluation of doc2vec with practical insights into document embedding generation" by Jey Han Lau and Timothy Baldwin [5]

In this paper, doc2vec is shown as an extension of word2vec and it presents a rigorous empirical evaluation of doc2vec over 2 tasks. It was observed that doc2vec performs extremely well when we use external corpora and can be further improved by using pre-trained word embeddings.

5. METHODOLOGY

The flow of the model is as follows:

(1) Cleaning of data: Set of prepositions and articles like "the, is, of" etc. are to be removed from the dataset. These are called as stopwords and they have to be removed from the dataset and only the meaningful words are to be retained. This process involves the removal of these stopwords.

(2) Tokenization of inputs: After removing the stop words from the dataset all that is left is meaningful data. Now that data has to be processed. In this step, tokenization of data is done in order to separate each word and each word becomes individual elements of a list

(3) Create a bag of words: Bag of words model is a representation that is used in information retrieval. In this model, the text is represented as a bog of its words, ignoring the grammar but keeping multiplicity. This model is used in classification where the occurrence of each word is used for training a classifier.

(4) Comparison of keywords and getting the comparison percentage: Similar words are to be matched and the meaning of the distinct word's probability has to be calculated. Based on this probability value, a word can be assessed if it matches the given meaning. A visualization of the above process is shown below:

```
w1 = ["polite"]
model.wv.most_similar (positive=w1,topn=6)

[('courteous', 0.9174547791481018),
 ('friendly', 0.8309274911880493),
 ('cordial', 0.7990915179252625),
 ('professional', 0.7945970892906189),
 ('attentive', 0.7732747197151184),
 ('gracious', 0.7469891309738159)]
```

Courteous has the highest probability that it matches the word polite. Hence it can be said that courteous and polite mean the same.

(5) Assigning marks based upon the comparison percentage: After comparing 2 texts (one from answer script and other from marking scheme), based upon the comparison percentage, marks are allocated. However, this depends on the threshold value that we set for the comparison percentage.

6. RESULT ANALYSIS

Though the machine learning concept is been in the market for over a decade, it still hasn't been widely used. The reasons behind this are:

- Implementing machine learning is not an easy task and it takes a lot of time for one to gain knowledge on the libraries that are used for machine learning.
- The accuracy of the future prediction of values depends highly on the data that is used to train the model. So finding the right dataset to train the model is very difficult and challenging.

After training the model with some valid data, we were able to achieve the following results. Here we compare the meanings of 2 words and the output is the similarity probability.

```
In [20]: model.wv.similarity(w1="stinky",w2="smelly")
Out[20]: 0.7834285814974897

In [21]: model.wv.similarity(w1="dirty",w2="clean")
Out[21]: 0.28372320434400145
```

As we can see above, 2 comparisons are made. One with the words: "stinky" and "smelly" which literally mean the same and

has 0.78 match probability. The other example is comparing "dirty" and "clean" which are the opposite. Hence the match probability is low.

However, the difference is not much. In an ideal world, when we compare clean and dirty, we must get 0 or negative values. But these values are dependent on the dataset that we use for training and hyperparameter values. Greater accuracy can be achieved by training the model with the larger dataset and training the model with a wide variety of domains.

7. CONCLUSION

In the existing trend, the answer scripts are evaluated manually by checking line by line which requires more manpower and time. But the model that we develop enables answer script to be evaluated at a very short time and doesn't require any human intervention. However, the model that is to be developed depends highly on the dataset that we use to train. Therefore, it is very important that we choose the dataset and the hyperparameters in such a way that maximum accuracy is obtained. By using a good dataset, we can be able to increase the gap between the synonyms and the antonyms, thereby increasing the overall accuracy of the model.

8. REFERENCES

[1] "NLTK: The Natural Language Toolkit" authored by Steven Bird and Edward: Introduction to text processing

[2] "A Compare-aggregate Model For Matching Text Sequences" authored by Shuohang Wang and Jing Jiang: matching vector values

[3] "A Study on Word2Vec" authored by Nina Tahmasebi: an introduction to word2vec

[4] "Progress in the Application of Natural Language Processing to Information Retrieval Tasks" authored by Alan F. Smeaton: finding out the same meaning in the different set of words

[5] "An empirical evaluation of doc2vec with practical insights into document embedding generation" by Jey Han Lau and Timothy Baldwin

[6] Working of word2vec URL: https://en.wikipedia.org/wiki/Word2vec

[7] Introduction to Doc2vec URL: https://medium.com/scaleabout/a-gentle-introduction-to-doc2vec-db3e8c0cce5e

[8] Bag of words URL: https://en.wikipedia.org/wiki/Bag-of-words_model.

[9] Gensim URL: https://radimrehurek.com/gensim/

