I.J. Education and Management Engineering, **2019, 6, 25-34**
Published Online September 2019 in MECS (http://www.mecs-press.net)
DOI: 10.5815/ijeme.2019.06.03

Available online at ***http://www.mecs-press.net/ijem***

Text Analyzer for Competitive Examination

Ashwini Dalvi, Irfan Siddavatam, Sagar Ailani, Smith Dedhia, Shyamal Makwana

Department of Information Technology, K.J.Somaiya College of Engineering, Mumbai-400077,India

Received: 06 July 2019; Accepted: 30 August 2019; Published: 08 October 2019

Abstract

Competitive examination provide a platform to the user for gauging their verbal and literature skills. The tools available currently only provide some simple feature regarding text processing such as spelling correction and providing different synonyms of the selected words. A complete assessment is not done for the user's abilities and relevant details related to the context are not taken entirely into consideration. The following paper proposes a way to implement Natural Language Processing on text to provide feedback to the user for their competitive examinations. The assessment of the text will be done according to the parameter such as grammar, vocabulary; relevance to the context.

Some applications for web and mobile platform are available to offer assessment of English language essay but limited academic research available to validate research work in this domain. This work is effort to address requirement of text analyzer for English language evaluation methods incorporating natural language processing.

Index Terms: GRE, TOEFL, NLP, text processing, topic modeling

© 2019 Published by MECS Publisher. Selection and/or peer review under responsibility of the Research Association of Mode rn Education and Computer Science

* Corresponding author.
E-mail address:

1. Introduction

Today Every year thousands of students in India appear for English Language Proficiency Tests to apply for English-speaking Universities. The process of these tests is well-defined, and the parameters used to assess the candidates are also clearly described, and yet there are no full-fledged tools available for students, to help them prepare for these tests.

Most of the available solutions in the market are intended just as an English-Language writing-enhancement tool, and they mainly focus on helping users improve word choice and grammatical errors in their writing. There are some official mock-tests for these exams, but they rate the candidate's performance on sample answers. Most tools have limitation of 1 essay per day for free user in which they only provide a simple grade without pointing out the flaw in the respective essay.

Mr. Text Analyzer intends to help students for preparation of these tests by assessing their English Speaking skills based on parameters the current systems offer, as well as offering additional features such as improving vocabulary and suggesting relevant topics for the given text.

In this work we present a web application which helps for assessment of English language essay for competitive examination. We identify status of present work in this context in literature survey and research gap identified as no complete assessment is done to evaluate user performance.

2. Liturature Survey

Mr. Text Analyzer intends to help students for preparation of these tests by assessing their English Speaking skills based on parameters the current systems offer, as well as offering additional features by improving sentence structuring and rating relevance to the given topic of the text.

Spell Check functionality will be implemented using Levenshtein Distance, which finds the distance between the misspelled word, and possible correct word [5].

Relevance will be checked using a method called Explicit Semantic Analysis (ESA), it represents meaning in a high-dimensional space of natural concepts derived from Wikipedia [3]. Another approach that could be used for checking semantic relatedness is Latent Semantic Analysis (LSA), the problem with using LSA is that the computed concepts cannot be readily mapped into natural concepts manipulated by humans; ESA overcomes this problem as it uses Wikipedia to extract natural concepts which are defined by humans [3].

In this approach, stop words will be removed from further processing as they contribute very little to the meaning of the text, and then the stemming process is performed on the remaining words [3]. An Inverted-Index is created those maps from words to a weighted list of concepts, where term frequency-inverse document frequency (TF-IDF) scheme is used to quantify the strength of association of words and concepts [3]. Co-sine Metric will be used to compute semantic relatedness or relevance, by comparing weighted vectors of two texts [2].

This is a comparison of the features provided between the existing system and our proposed system [6, 7].

Vocabulary enhancement is implemented using Skip-gram word embedding. Skip-gram embeds both target words and contexts in the same low-dimensional space [2]. Candidates are selected from a single sentence based on part of speech. Apart from these candidates from each sentence the rest of the words in that sentence are considered as context, for selecting a replacement word. Two Sets of files are generated after preprocessing the Embedding files, one for words, and one for context. The embedding files have 150 dimensions. The modules load these two files in the RAM for faster processing.

Once the text is submitted as input, the report will be generated after the processing is complete. The report will state the misspelt words highlighted in red and their respective corrective words next to it highlighted in green. The vocabulary improvement suggestions will be highlighted in green next to the word which will be

highlighted in red. The relevance check will list words of topics that can be included in the text to improve the relevance of the text. We also store the previous result of the user in the database which includes all the answers provided by the user as well as the result provided by the system so that user can improve by referring to previous results.

Table 1: Comparison with Existing System

	Grammarly	TOEFL IBT	Mr. Text Analyzer
Free of Cost	No	No	Yes
Sentence Structuring	Yes	No	No
Text Score	Yes	Yes	Yes
Relevance	No	No	Yes
Cross Platform	Yes	No	Yes

There are various tools that evaluate Analytical Writing Assessment (AWA) by provide rating to the essay submitted by the user. Most these tools provide a simple rating without any detailed assessment or any guidance for improving the writing skill of the user.

MBA Crystal Ball provides a tool for rating the essay of the user. This tool is fully automated which uses natural language processing. User just needs to paste or type the essay in the text box. Click on check button to grade the essay. The tool will only provide a rating between 0-6 without any other details for free users. Their premium version is only available for test prep companies where they provide additional insight on the evaluated essay [13]. User does not need to include the essay topic for free evaluation.

AWA PROFESSOR is a paid service which will grade users essay twice. The first rating is provided manually by the professional essay rater. The second rating is by an automated Computer Intelligence Rater that evaluates structural and linguistic features in writing [14]. Grading of essay takes maximum of up to 48 hours.

TESTBIG's e-grader is an online service which helps user in grading user's essay by providing a rating out of 6. This tools limits 1 essay evaluation per day for free users. The e-grader does not examine the meaning of words and ideas whereas VIP users can receive further evaluations by advanced module of e-grader and human graders [15]. This tool grades grammar of the essay based on the readability, performance of part of speech, vocabulary words and sentences.

The e-rater engine is an Educational Testing Service (ETS) capability that identifies features related to writing proficiency in student essays so they can be used for scoring and feedback. The e-rater engine is used within the Criterion Online Writing Evaluation Service. The user needs to buy a subscription to use this service. This Service is a web-based instructor-led writing tool that helps students plan, write and revise their essays [16]. It gives them immediate diagnostic feedback and more opportunities to practice writing at their own pace.

The tools mentioned above are pay per use tools or provide limitation for free user. Therefore we propose an application that will help user to improve essay writing skills by suggesting better vocabulary and recommend topics for user to consider for writing the essay.

3. Proposed Methodology

The tools used for implementation of the application are:

- Python – For creating APIs, and processing
- MongoDB – For storing user data
- NLTK – Python library for NLP
- React.js - JavaScript library for building UI
- Postman – For testing API's

The proposed system will first take text as input and the said text will be checked for any spelling mistakes. After the spelling mistakes have been corrected the spell checked text will be used as input for vocabulary enhancement of the text. The next step is to check the relevancy of the text with the modal text which will be fetched from the database. Topics will be generated on the said modal text and displayed to the user. Finally, a report will be generated which will show the spelling mistakes, replacements for improving the vocabulary of the text and topic suggestions to improve the relatedness of the text.

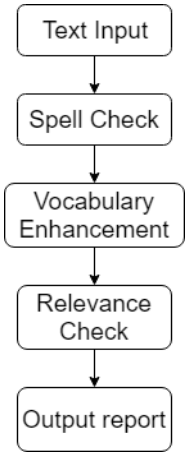

Fig. 1. Block Diagram of the modules

A. Spell-Check module

Spelling mistakes found in the text will be corrected using the Levenshtein distance. The Levenshtein distance will be checked between the misspelt words and the suspected correct words. Word with the highest frequency will be returned. This will be done for all words in the text which will return the corrected words. Figure 2 provides the flow of spell check module.

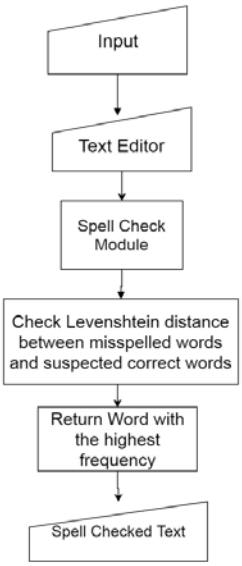

Fig. 2. Working of Spell check module

B. Vocabulary Enhancement

Vocabulary enhancement is implemented using Skip-gram word embedding. Skip-gram embeds both target words and contexts in the same low-dimensional space [2].

Candidates are selected from a single sentence based on part of speech. Apart from these candidates from each sentence the rest of the words in that sentence are considered as context, for selecting a replacement word.

Two Sets of files are generated after preprocessing the Embedding files, one for words, and one for context. The embedding files have 150 dimensions. The modules load these two files in the RAM for faster processing

Figure 3 describes the working of vocabulary enhancement below.

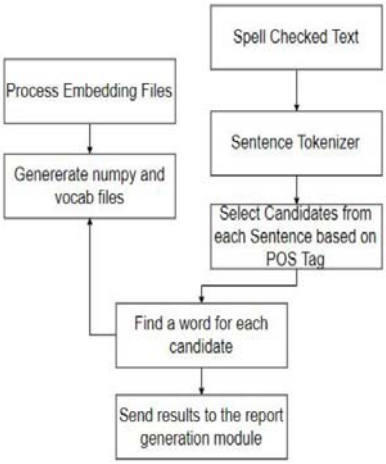

Fig. 3. Vocabulary Enhancement

C. Relevance Check

Relevancy check is achieved by using a method called Explicit Semantic Analysis (ESA). Explicit Semantic analysis is a way to extract meaning from a text. The text may be a single word, a couple of words, a sentence or a paragraph.

ESA represents meaning in a high-dimensional space of natural concepts derived from Wikipedia [1]. Another approach that could be used for checking semantic relatedness is Latent Semantic Analysis (LSA), the problem with using LSA is that the computed concepts cannot be readily mapped into natural concepts manipulated by humans, ESA overcomes this problem as it uses Wikipedia to extract natural concepts which are defined by humans [1].

In this approach, we have to remove stop words from further processing as they contribute very little to the meaning of the text, and then the stemming process is performed on the remaining words [1]. An Inverted-Index is created that maps from words to a weighted list of concepts, where term frequency-inverse document frequency (TFIDF) scheme is used to quantify the strength of association of words and concepts [1]. Cosine similarity metric will be used to compare the user provided answer with the modal answer.

In our application, we have used the 2015 Wikipedia dumps.

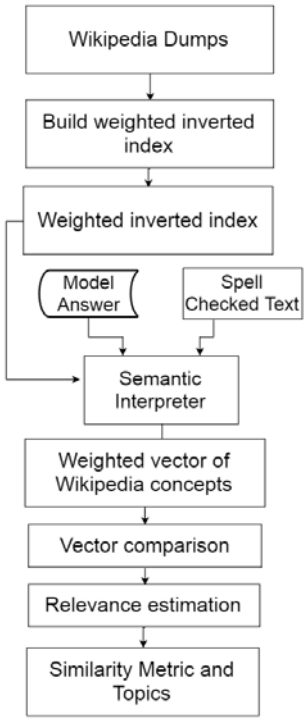

Fig. 4. Steps of Relevance check

D. Report Generation

Once the text is submitted as input, the report will be generated after the processing is complete. The report will state the misspelt words highlighted in red and their respective corrective words next to it highlighted in green.

The vocabulary improvement suggestions will be highlighted in green next to the word which will be highlighted in red. The relevance check will list words of topics that can be included in the text to improve the relevance of the text.

We also store the previous result of the user in the database which includes all the answers provided by the user as well as the result provided by the system so that user can improve by referring to previous results..

4. Result

The following are the screenshots of our application which include all modules mentioned above.

The outputs are displayed to the user in tabbed form which will help the user to have a better grasp in comprehending the result.

Fig. 5. Editor

In figure 5, we fetch the question from database for user to attempt. We have a ‘what you see is what you get’ (WYSIWYG) editor implemented using draft-js in our react application.

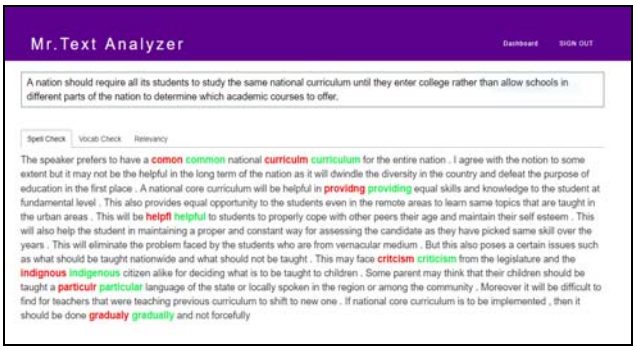

Fig. 6. Spell Check

In figure 6, we have the output for spell check module. The misspelled words are displayed in red font color whereas corrected words are in green color.

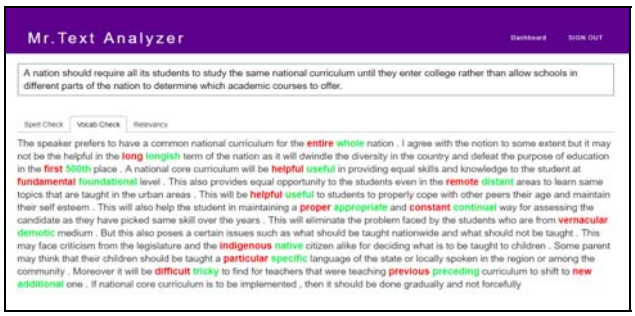

Fig. 7. Vocabulary Enhancement

In figure 7, the words used by the user are displayed in red font while the improved words are displayed in green font.

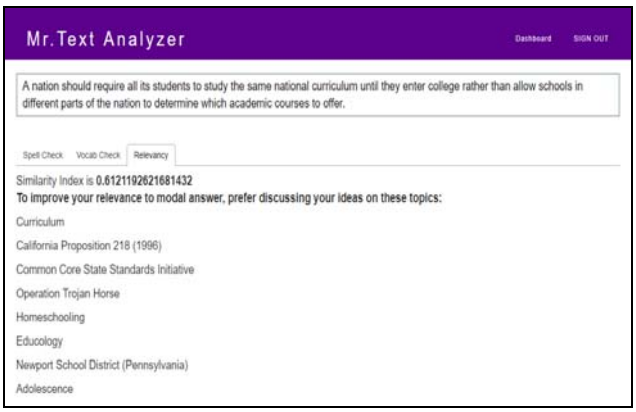

Fig. 8. Relevance check

In figure 8, application will display the result of cosine similarity between the modal answer and user's answer. Application will also display the generated topics from the modal answer.

5. Conclusion

Our application, Mr. Text Analyzer is a Web-App which will help user to improve their essay writing skills. The application will also help in widening vocabulary knowledge so that user can write new words in the essay. The application also helps in generating and recommending topics that are related to modal answer which user can mention in their essay for better context.

Our application will also save the previously submitted test so that user can gain new understanding from the previous suggestion offered at any time.

Our application makes use of Levenshtein distance for spell checks. Vocabulary enhancement is achieved using skip gram model. Use Explicit Semantic Modeling for topic generation. Report generated will help the user to enhance their skills for their competitive exams in an effective way.

In future a provision can be made where user will have the option to answer the questions using their own voice. The voice will be recorded and then can be converted to text using any Speech-To-Text API's

References

[1] Rada Mihalcea, Courtney Corley, Carlo Strapparava, "Corpus-based and Knowledge-based Measures of Text Semantic Similarity", 2006.
[2] E Gabrilovich, S Markovitch, "Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis" (2007). IJcAI 7, 1606-1611
[3] Moore, Brian J. "A Real-Time N-Gram Approach to Choosing Synonyms Based on Context."
[4] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, Jefferey Dean "Distributed Representations of Words and Phrases and their Compositionality," 2013
[5] Casey Whitelaw, Ben Hutchinson, Grace Y. Chung Gerard Ellis "Using the Web for Language Independent Spellchecking and Autocorrection," 2009 Google Inc.
[6] Monojit Choudhury, Markose Thomas, Animesh Mukherjee, Anupam Basu, Niloy Ganguly "How Difficult is it to Develop a Perfect Spell-checker? A Cross-linguistic Analysis through Complex Network Approach," 2007 Association for Computational Linguistics
[7] Chen, Danqi, Manning, Christoper "A Fast and Accurate Dependency Parser using Neural Networks," 2014 Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)
[8] Brian J. Moore, Robert Mercer "A Real-Time N-Gram Approach to Choosing Synonyms Based on Context," 2015 Electronic Thesis and Dissertation Repository
[9] Wikipedia Dumps, https://meta.wikimedia.org/wiki/Data_dumps
[10] Reading Wikipedia XML Dumps with Python, https://www.heatonresearch.com/2017/03/03/ python-basic-wikipedia -parsing.html
[11] Marian Neural Machine Translation, https://marian-nmt.github.io
[12] Oren Melamud, Omer Levy, Ido Dagan, "A Simple Word Embedding Model for Lexical Substitution". Proceedings of NAACL-HLT 2015
[13] Free Online GRE AWA Essay Grader - MBA Crystal Ball https://www.mbacrystalball.com/gre/gre-essay- grader
[14] AWA Professor: Expert GMAT & GRE AWA Essay Raters https://www.awaprofessor.com
[15] testbig.com | TOEFL IELTS GMAT GRE SAT ACT PTE ESL. https://www.testbig.com/
[16] ETS Criterion writing evaluation service http://www.ets.org/criterion

Authors' Profiles

Ashwini Dalvi joined the Department of Information Technology, K.J.Somaiya College of Engineering, Mumbai in 2006 as an Assistant Professor. She has published over 25 journal and conference papers in the areas of Security, Intelligent applications.

Irfan Siddavatam has received the PH.D. Degree from VJTI, affiliated to Mumbai University, in 2018.
In 2001, he joined the Department of Information Technology, K.J.Somaiya College of Engineering, Mumbai, as an Associate Professor. His research interests include Cyber Physical System Security, Artificial Intelligence, and Internet of Things.

How to cite this paper: Ashwini Dalvi, Irfan Siddavatam, Sagar Ailani, Smith Dedhia, Shyamal Makwana," Text Analyzer for Competitive Examination", International Journal of Education and Management Engineering(IJEME), Vol.9, No.5, pp.25-34, 2019.DOI: 10.5815/ijeme.2019.06.03