MARVIN - The Intelligence Evaluator

Kumar Nityan Suman, **Aisha Begam, Abhinish Kumar, Aparna Singh**

Information Science and Engineering, PES University

Abstract- Conducting examination is a hectic process. It is an assessment intended to measure a test-taker's knowledge, skill, aptitude, and so on. Standard approach of conducting an examination is expensive, resource consuming and time taking. Major tasks involved in conducting a successful examination includes questions paper generation, answer key generation, fair conduction of test and standardized evaluation. Question paper setting takes a lot of time and requires a skilled human work. Same goes with the answer key. Manual grading of answers takes up a significant amount of valuable time, money and other resources. Air conduction of examination is another challenge that organization faces with current infrastructure. Standardized evaluation of answer sheets are another concern which will always have human bias playing a part in the current scenario. Our work aims to solve this problem by building an automated examination platform using cutting-edge machine learning, natural language processing and web technologies. We aim to provide an inexpensive alternative to the current examination system.

Index Terms- Automated Examination Platform, Answer Evaluation, Exam Assessment Analysis, Keyphrase extraction, Question Identification, Text Similarity Ranking

I. INTRODUCTION

An examination (informally, test, exam or evaluation) is an assessment intended to measure a test-taker's (candidates) knowledge, skill, aptitude, or classification in many topics. In the current modern mass-education system, the style of examination is mostly fixed, with the stress on standardized papers to be sat by large numbers of students. Both World War I and World War II demonstrated the necessity of standardized testing and the benefits associated with these tests. Tests were used to determine the mental aptitude of recruits to the military. The US Army used the Stanford–Binet Intelligence Scale to test the IQ of the soldiers. Thus it is very important that examination needs to be conducted in a fair and standardized manner for any unfair advantage or bias against the other.

The purpose of this work (MARVIN **-** **M**achine **A**ssessment using **R**eactive **V**iew **IN**telligence) is to build an inexpensive, accurate and efficient software platform which automatically forms questions along with their respective answers to conduct an online examination (or evaluation). Automated examination, if proven to match or exceed the reliability of human evaluators then it will reduce costs, resources and time required for conducting manual examination. Also it will be easier to understand the underlying patterns behind how students respond to a test.

II. RELATED WORK

The literature survey was done to gain insights on prior work done on question formation using classical natural language processing, answer evaluation and platform dependent challenges.

Manvi Mahana, Mishel Johns, Ashwin Apte's paper, *"Automated Essay Grading Using Machine Learning",* took a shot on grading essays which were categorized into 8 classes based on the context. The approach was able to achieve a *kappa score* of 0.73 across all 8 essay sets. Total number of essays taken into consideration for this experiment was ~13K from *kaggle.com* . They used 5-fold cross validation to train and test the model rigorously. Jason Zhao's attempt to score essays in the paper, *"Essay Scoring using Machine Learning",* was very different than others. Their focus for this essay grading was the style of the essay, which is an extension on the studies conducted determining the quality of scientific articles by adding maturity to the feature set (Louis and Nenkova, 2013). The dataset used is from *kaggle.com*, containing ~13K, categorized into 8 topics based on the context.

As far as our knowledge is concerned, there are various online examination platform available in present day. All of them fail to create a question on the fly and also to identify answers. They mainly maintain a collection of question - answer pairs, which are manually created and identified. No work has been done in this manner to automate the entire examination system using machine learning and natural language processing to remove human intervention and its resulting bias in the system.

III. OUR MODEL

Our platform consists of four parts: keyphrase extraction module, question formation module, answer identification module and response evaluation module. Suppose the input document (standard .txt format) is S which consists of multiple sentences = $\{s_1, s_2, s_3, \ldots, s_T\}$, T being the total number of sentences in the document. The goal here is to form questions, identify answers to the respective questions, which is then used to evaluate candidate responses. A detailed report is generated based on the number of correct responses. Two types of test can be generated: objective and subjective. Objective test will have questions with four probable answers given as option, only one being correct. On contrary, subjective tests have long type answers. No probable solution is provided in this case. User/candidate is expected to form the answer all by him/herself. The answers are evaluated on the basis of contextual similarity and not naive text similarity metric.

Keyphrase Extraction

The goal of keyphrase extraction is to get most important keyphrases, where length of keyphrase i.e., number of words allowed in a keyphrase, *1 <= number_of_words <= 3.* Keyphrase extraction module uses TF-IDF with ELMo based sentence embeddings to prune out the unnecessary keyphrases. TF-IDF, short for term frequency–inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. ELMo is a deep contextualized word representation that models both (1) complex characteristics of word use (e.g., syntax and semantics), and (2) how these uses vary across linguistic contexts (i.e., to model polysemy). These word vectors are learned functions of the internal states of a deep bidirectional language model (BiLM), which is pre-trained on a large text corpus. ELMo is also used in answer evaluation. Contextual vector representation of original answer and the response is generated for similarity measure using a siamese BiLSTM model architecture.

Bidirectional Language Model

Given a sequence of N tokens, (t_1, t_2, ..., t_N), a forward language model computes the probability of the sequence by modeling the probability of token t_k given the history (t_1, ..., t_{k-1}):

$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_1, t_2, \ldots, t_{k-1}).$$

Figure 3.1: Forward Language Model

A backward LM (language model) is similar to a forward LM, except it runs over the sequence in reverse, predicting the previous token given the future context:

$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_{k+1}, t_{k+2}, \ldots, t_N).$$

Figure 3.2: Backward Language Model

A biLM combines both a forward and backward LM.

ELMo

Unlike most widely used word embeddings (Pennington et al., 2014), ELMo word representations are functions of the entire input sentence, as described in this section. They are computed on top of two-layer biLMs with character convolutions, as a linear function of the internal network states. This setup allows us to do semi-supervised learning, where the biLM is pre-trained at a large scale and easily incorporated into a wide range of existing neural NLP architectures. ELMo is a task specific combination of the intermediate layer representations in the biLM. For each token tk, a L-layer biLM computes a set of 2L + 1 representations:

$$\begin{aligned} R_k &= \{\mathbf{x}_k^{LM}, \overrightarrow{\mathbf{h}}_{k,j}^{LM}, \overleftarrow{\mathbf{h}}_{k,j}^{LM} \mid j = 1, \ldots, L\} \\ &= \{\mathbf{h}_{k,j}^{LM} \mid j = 0, \ldots, L\}, \end{aligned}$$

Figure 3.3:
Where, $h_{k,\,0}$ is layer and $h_{k,\,j}$ = [For. $h_{k,\,j}$;Bac. $h_{k,\,j}$], for each biLSTM layer.

The identified phrases are used by question formation module to create question based on the predefined question templates. Once questions are created, answer identification module takes both questions and phrases to identify appropriate answers. Answer module uses pre-trained BiDAF (*Bidirectional Attention Flow*) from *Microsoft Inc.*, to identify answer for the given question from a set of probables. Ideally BIDAF architecture is used in machine comprehension but here we have used it as a question - answering model. It selects an answer from the given probables. The questions formed are used to test candidates and the received responses are evaluating w.r.t the identified answer using cosine distance or Euclidean between features vectors from Siamese LSTM neural network architecture.

Siamese LSTM

The model is outlined in the Figure below. There are two networks LSTMa and LSTMb which each process one of the sentences in a given pair, but we solely focus on Siamese architectures with tied weights such that LSTMa = LSTMb in this work.

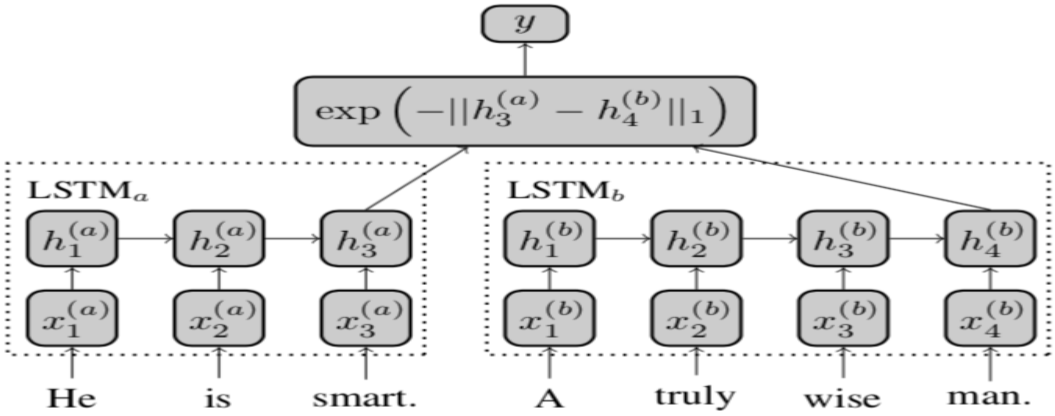

Figure 3.4: Siamese LSTM Model Architecture

The LSTM learns a mapping from the space of variable length sequences of d_{in}-dimensional vectors into R_{drep} (d_{in} = 300, drep = 50 in this work). More concretely, each sentence (represented as a sequence of word vectors) x1 , . . . , xT , is passed to the LSTM, which updates its hidden state at each sequence-index via equations (1)-(7). Below are the updates performed at each t ∈ {1,...,T} in an LSTM parameterized by weight matrices Wi, Wf , Wc, Wo, Ui, Uf , Uc, Uo and bias-vectors bi,bf,bc,bo:

$$h_t = \text{sigmoid}\left(Wx_t + Uh_{t-1}\right) \qquad (1)$$

$$i_t = \text{sigmoid}\left(W_i x_t + U_i h_{t-1} + b_i\right) \qquad (2)$$

$$f_t = \text{sigmoid}\left(W_f x_t + U_f h_{t-1} + b_f\right) \qquad (3)$$

$$\widetilde{c}_t = \tanh\left(W_c x_t + U_c h_{t-1} + b_c\right) \qquad (4)$$

$$c_t = i_t \odot \widetilde{c}_t + f_t \odot c_{t-1} \qquad (5)$$

$$o_t = \text{sigmoid}\left(W_o x_t + U_o h_{t-1} + b_o\right) \qquad (6)$$

$$h_t = o_t \odot \tanh(c_t) \qquad (7)$$

The two vector representation are measured on their similarity using the Euclidean distance between the two. Here we have an exponent of a negative (in this case) the output will be between 0 and 1. Based on the similarity score, answers are evaluated. Higher the score, higher the performance of the candidate. A detailed analysis report is generated based on the candidates' performance. It includes number of wrong attempts, number of correct attempts, type of mistakes made and similarity of mistakes.

IV. IMPROVEMENT AND FUTURE WORK

Future works include question generation using deep recurrent neural networks. Answer evaluation is done using a Siamese LSTM model. A more accurate result can be achieved by using bidirectional LSTM to counter the forward and backward context. One of the limitations of the proposed framework is its inability to generate computational problems and evaluate the same. Computer vision using deep convolutional neural networks comes to the rescue here and can play a vital in detecting mathematically formulas and converting them to a format more understandable to the model. Due to limited computational resources (GPUs), the models are not tuned to the best of its capacity. A good hyper-parameter tuning can help enhance the performance further.

V. CONCLUSION

In this paper, we proposed a platform framework to automate the current examination system. The platform forms questions based on a text corpus, then identifies answer to the formed questions and gives the candidate the opportunity to test his/her skills using an objective or subject type of examination. We achieved an accuracy of 88% in terms of subjective answer evaluation and a whooping 97% for objective answer evaluation, ~80% in terms of objective question formation and a good accuracy of 71% in terms of subjective question formation. The question formation is based on classical keyphrase extraction. The system developed is user-friendly and can be easily used by a naive user with little or no overhead knowledge. The system performs unbiased and consistent evaluation, which is a common problem encountered in offline examination systems. The platform can be accessed through any device as long as it supports a client browser and has internet connectivity.

REFERENCES

[1] *Automated Essay Grading Using Machine Learning* - Manvi Mahana, Mishel Johns, Ashwin Apte CS229 Machine Learning - Autumn 2012 Stanford University.

[2] Shihui Song & Jason Zhao (2012) , "*Automated Essay Scoring Using Machine Learning*".

[3] Peters, Matthew E. and Neumann, Mark and Iyer, Mohit and Gardner, Matt and Clark, Christopher and Lee, Kenton and Zettlemoyer, Luke, "*Deep contextualized word representation*".

[4] *Siamese Recurrent Architectures for Learning Sentence Similarity* - Jonas Mueller, Aditya Thyagarajan, AAAI Conference on Artificial Intelligence (AAAI -16).

[5] Livescu. 2016. *Charagram: Embedding words and sentences via character n-grams*. In *EMNLP*.

[6] Bennani-Smires, Musat, Hossmann, Baeriswyl, Jaggi, "*Simple Unsupervised Keyphrase Extraction using Sentence Embedding",* Swisscom AG.

[7] Quoc Le, Tomas Mikolov, and Tmikolov Google Com. 2014. *"Distributed Representations of Sentences and Documents"*, ICML.

[8] Shang, Liu, Jiang, Ren, R Voss, Han -*"Automated Phrase Mining from Massive Text Corpora"*, 2017.

[9] Sepp Hochreiter; Jürgen Schmidhuber (1997). *"Long short-term memory".* Neural Computation 9(8): 1735-1780, 1997.