Witty – The Smart Recruiter

Pratheeksha S Karandoor [1], Prithviraj S Amin1, Ranjith Raj B R[1]

[1]Dept. of Computer Science and Engineering, Sahyadri College of Engineering and Management, Adyar, Mangalore, India - 575009

ABSTRACT

Short-listing the candidates based on their verbal knowledge, intellectual capability and other aspects has been a burdensome task for hiring. The application will provide sophisticated details of the candidates, their strengths, their weakness and level of preparation along with a comprehensive character evaluation. The question generating system that currently exists is totally based on database and quantitative evaluation. In the proposed system intelligent generation of question or task shall be a collective outcome of previously attempted task. Candidate will be evaluated based on the character analysis considering various parameters such as difficulty level, nature of the task and the time taken to complete the task. The system will generate the questions or tasks based on the candidate's behavior and generate a collective assessment report from the questions.

Keywords:

Binary decision trees, Computational intelligence, Intelligent systems, Personality detection, Machine learning, Text analysis.

1. INTRODUCTION

Previously aptitude and coding question for hiring were generated manually, which was time consuming. To overcome this problem new system will be implemented that generates the questions based on the performance from the previously answered question. The question generating system that currently exists is totally based on database and quantitative evaluation takes place. In the proposed system intelligent generation of question will take place and each question or task shall be a collective outcome of the previously attempted task.

Evaluation of the candidate will occur based on character analysis taking into account various parameters such as task difficulty, nature of the task, time taken to complete the task etc. The system shall adapt to the candidate's behaviour and generate questions or tasks accordingly and generate a collective assessment based on analysis of all the questions, thus behaving as an intelligent interviewer.

This system is mainly developed to eradicate the problems faced during the manual process of recruitment. The basic recruitment process is a tedious task where the recruiter has to go through a series of phases for a prolonged time which adds to increased costs. Quantitative evaluation of candidates does not yield efficient result in talent acquisition. Witty will take up all the above processes as per the requirement of the administrator and automate the initial recruitment process thereby acting as an intelligent interviewer. Based on the report administrator will select the candidate. Witty helps the administrator to conclude the areas in which the candidate has strong hold.

2. TERMINOLOGIES

A few terminologies that will be used frequently is explained in detail in this section.

2.1 Quantitative Evaluation

There are basically two types of evaluation: Quantitative evaluation and Qualitative evaluation. Quantitative evaluation is made by utilizing logical instruments and estimations. The outcomes can be estimated or checked, and some other individual attempting to quantitatively survey a similar circumstance should finish up with similar outcomes. Qualitative evaluation is characterized in science as any evaluation made by utilizing the five senses. Since individuals regularly achieve distinctive translations when utilizing just their faculties, subjective assessment winds up more enthusiastically to repeat with precision; two people assessing something very similar may finish up with various or clashing outcomes. In this project, the former type of evaluation has been used.[9]

2.2 Intelligent system

An intelligent system is a machine with an installed, Internet-associated PC that has the ability to assemble and examine information and communicate with different devices. Other criteria for intelligent system are to incorporate the ability to gain as a matter of fact, security, availability, the capacity to adjust as indicated by current information and the limit with respect to remote monitoring and management.[8]

2.3 Word2vec

The set of related models which are used to produce word embeddings is called Word2vec. Word embedding involves a set of language modeling and feature learning techniques in natural language processing(NLP) where all the words or phrases from the vocabulary are mapped to vectors of real numbers. Word2vec models are shallow, two-layer neural networks that are trained to reconstruct linguistic contexts of words. Word2vec takes as its input a large data in the form of text and produces a vector space, of several hundred dimensions, with each unique word in the data being assigned a corresponding vector in the space.Word vectors are positioned in vector space in such a way that the words that share common contexts are located closely to one another in the space. [7]

3. RELATED WORKS

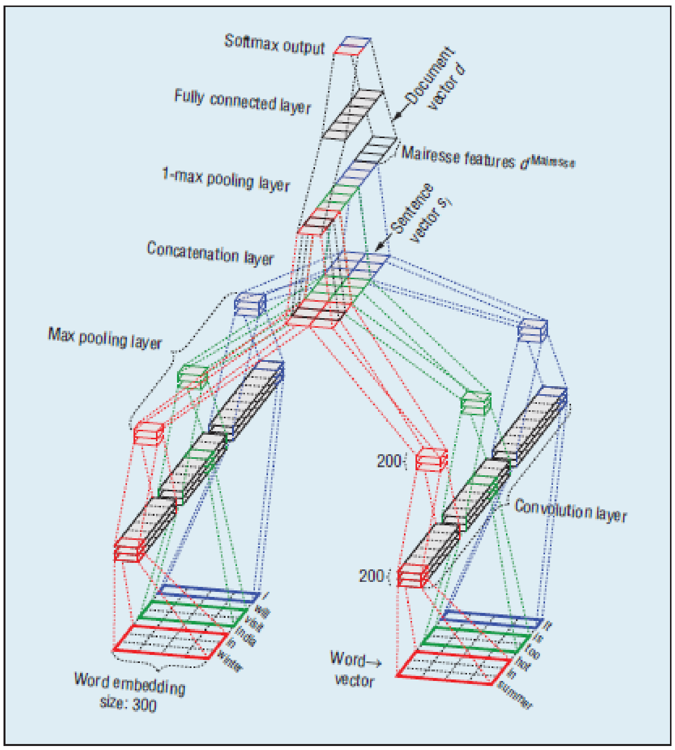

Figure 1. The network consists of seven layers. The input layer (shown at the bottom) corresponds to the sequence of input sentences (only two are shown). The next two layers include three parts, corresponding to trigrams, bigrams, and unigrams. The dotted lines delimit the area in a previous layer to which a neuron of the next layer is connected—for example, the bottom-right rectangle shows the area comprising three-word vectors connected with a trigram neuron.

The network comprises seven layers: input (word vectorization), convolution (sentence vectorization), max pooling (sentence vectorization), 1-max pooling (document vectorization), concatenation (document vectorization), linear with Sigmoid activation (classification), and two neuron SoftMax output (classification). Figure 1 depicts the end-to-end network for two sentences.[1]

Document vector *d*	Filter	Classifier	Convolution filter	Personality traits				
				EXT	NEU	AGR	CON	OPN
N/A	N/A	Majority	N/A	51.72	50.02	53.10	50.79	51.52
Word *n*-grams	Not used	SVM	N/A	51.72	50.26	53.10	50.79	51.52
Mairesse[12]	N/A	SVM	N/A	55.13	58.09	55.35	55.28	59.57
Mairesse (our experiments)	N/A	SVM	N/A	55.82	58.74	55.70	55.25	60.40
Published state of the art per trait[12]	N/A	N/A	N/A	56.45	58.33	56.03	**56.73**	60.68
CNN	N/A	MLP	1, 2, 3	55.43	55.08	54.51	54.28	61.03
CNN	N/A	MLP	2, 3, 4	55.73	55.80	55.36	55.69	61.73
CNN	N/A	SVM	2, 3, 4	54.42	55.47	55.13	54.60	59.15
CNN + Mairesse	N/A	MLP	1, 2, 3	54.15	57.58	54.64	55.73	61.79
CNN + Mairesse	N/A	SVM	1, 2, 3	55.06	56.74	53.56	56.05	59.51
CNN + Mairesse	N/A	sMLP/FC	1, 2, 3	54.61	57.81	55.84	**57.30**	62.13
CNN + Mairesse	Used	sMLP/MP	1, 2, 3	**58.09**	57.33	**56.71**	56.71	61.13
CNN + Mairesse	Used	MLP	1, 2, 3	55.54	58.42	55.40	56.30	**62.68**
CNN + Mairesse	Used	SVM	1, 2, 3	55.65	55.57	52.40	55.05	58.92
CNN + Mairesse	Used	MLP	2, 3, 4	55.07	**59.38**	55.08	55.14	60.51
CNN + Mairesse	Used	SVM	2, 3, 4	56.41	55.61	54.79	55.69	61.52
CNN + Mairesse	Used	MLP	3, 4, 5	55.38	58.04	55.39	56.49	61.14
CNN + Mairesse	Used	SVM	3, 4, 5	56.06	55.96	54.16	55.47	60.67

Table 1. Accuracy obtained with different configurations

In the variant marked MLP in Table 1, we used the network shown in Figure 1, which is a multiple-layer perceptron (MLP) with one hidden layer, trained together with the CNN. In the variant marked SVM (support vector machine) in the table, we first trained the network shown in Figure 1 to obtain the corresponding document vector *d* for each document in the dataset, and then used these vectors to train a polynomial SVM of degree 3. In the variant marked sMLP/ MP in the table, in a similar manner we used the vectors *d* (the max pooling layer) to train a stand-alone MLP (using 50 epochs) with the same configuration as the last two layers in Figure 1 (that is, using the 1-max pool layer from Figure 1 as input). In another experiment, we fed to the stand-alone MLP the values from the fully connected layer instead of *d*; this variant is marked as sMLP/FC in Table 1. For baseline experiments not involving the use of CNN, we used only a linear SVM.[1]

4. METHODOLOGY

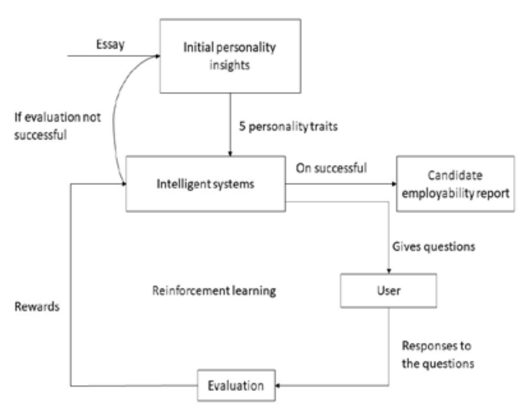

Figure 2. Architecture diagram of Witty - The Smart Recruiter

System architecture used for Witty – The Smart Recruiter is shown is “Figure 2”.

4.1 Initial personality insights

A person’s suitability for a certain job is determined by the personality traits. There are 5 personality traits which are the binary (yes/no) values:

- Extroversion (EXT): Is the person approachable, communicative and dynamic versus introvert and desolate?
- Neuroticism (NEU): Is the person emotional and uneasy versus defended and positive?
- Agreeableness (AGR): Is the person dependable, sincere, benevolent, and humble versus deceptive, problematic, tenuous, and arrogant?
- Conscientiousness (CON): Is the person competent and coordinated versus mediocre and irresponsible?
- Openness (OPN): Is the person innovative and interested versus stubborn and vigilant?[2]

4.2 Intelligent system

Intelligent systems work on the basis of reinforcement learning. This intelligent system is the heart of the system as it is used to generate the questions for the evaluation of the candidate. On successful evaluation it plays a major role is generating the candidate’s employability report. The output as reward points from the evaluator is received by the intelligent system based on which the employability report is generated. If the reward points and the initial personality traits decided do not tally, then the intelligent system will ask the user to rewrite the initial assignment till the personality traits and the reward points match.

4.3 Evaluator

Evaluator helps is evaluating the answers given by the user. The probability of the answer correctness of the particular question will be checked. Every answer is rewarded with certain points. The value for the reward points are based on the time taken by the user to answer the question and the difficulty level of the question. These rewards are then given as the input to the intelligent system.

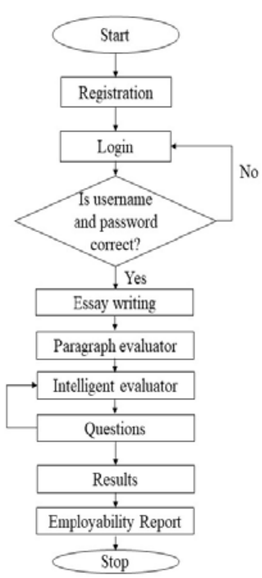

Figure 3. Flow chart of Witty – The Smart Recruiter

“Figure 3” represents the flow chart of Witty – The Smart Recruiter. This represents the workflow of the project. Initially the process starts with the user by registering with the system. If the user has already registered with the system, it directly goes to the login page. Otherwise the user has to first register and then log in to the system. The correct username and the password are verified by the system. If there is any mismatch with username and password, the user is not allowed to continue the process. Once the user has successfully logged in to the system, an essay topic is provided to the user for which the user is expected to write the essay. The paragraph evaluator analyses the essay written by the user, based on which the intelligent evaluator generates and produces the questions for the user to answer. Based on the answer chosen, the accuracy and the time taken by the user to answer the question, the intelligent evaluator generates the next question. This process is continued till a near accurate employability report is generated. Once the employability report is generated the process is stopped.

4. ESSAY EVALUATION

The 5 personality traits as explained in section 3.1 are obtained after the extensive evaluation of the essay provided as the input to the system. There are 5 steps to be followed. They are:

4.1 Preprocessing

Initially a text is split into a sequence of sentences at the question mark and period. Then every sentence is split into words at whitespace character. Every letter is converted to lowercase and all the characters except ASCII letters, digits, exclamation mark, single and double quotation marks are eliminated. If there are very long sentences consisting of more than 150 words without a question mark or a period, the sentence is broken down into sentences of 20 words each.[1]

4.2 Extracting document level features

“http://farm2.user.srcf.net/ research/personality/recognizer .html” has been used to extract 84 Mariesse features from the document.[3]

4.3 Sentence filtering:

It is assumed that every sentence will have atleast one emotionally charged word. Once the document level features are extracted and before extracting the word2vec features, all the sentences without an emotionally charged word is discarded. To obtain emotionally charged words NRC Emotion Lexicon(http://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.html) was used This lexicon contains 14,182 words tagged with 10 attributes: anger, anticipation, disgust, fear, joy, negative, positive, sadness, surprise, and trust. A word is considered to be emotionally charged if it had at least one of these attributes; there are 6,468 such words in the lexicon. So, if a sentence contained none of the 6,468 words, it is removed before extracting the word2vec features from the text.[5]

4.4 Extracting word-level features

We used the word2vec embeddings to convert words into 300-dimensional vectors. If a word was not found in the list, we assigned all 300 coordinates randomly with a uniform distribution in [−0.25, 0.25].[4]

4.5 Classification

A deep CNN is used for classification. The text is processed by the initial layers in a hierarchical manner. In the input every word is represented as a fixed length feature vector using word2vec, and the sentences are represented as a variable number of vectors. At some layer, this variable-length vector is reduced to fixed length vector of each sentence, which is a kind of sentence embedding in a continuous vector space. At that level, documents are represented as a variable number of such fixed-length sentence embeddings. Finally, at a deeper layer, this variable-length document vector is reduced to a fixed-length document vector. This fixed-length feature vector is then concatenated with the document-level features giving a fixed-length document vector, which is then used for final classification.[1]

5. RESULTS

- Implementation of Deep Learning-Based Document Modelling for Personality Detection from Text using Python.
- Testing and failure analysis of the Deep Learning-Based Document Modelling for Personality Detection from Text.
- Selection of IBM Watson API for personality insights.
- Implementation of IBM Watson API on a web-based interface.

6. CONCLUSION

- An interface for the candidate to write an essay of his interest which will predict his personality and the careers that will suit him. This will also result in predicting the 5 personality traits.
- Based on the evaluation of the essay written by the user, a set of aptitude questions will be generated for the candidate to answer.
- Based on the time taken by the candidate to answer the question and the closeness to the correct answer the candidate will be provided with the next set of questions.
- An intelligent application which will test the skills of the candidate and generates a performance report along with character analysis.
- Web based application for the candidate to interact with the intelligent application and for the admin to generate Performa reports.

ACKNOWLEDGEMENT

In the accomplishment of this project successfully, many people have bestowed upon us their blessings and their heart pledged support, this time we are utilizing to thank all the people who have been concerned with this project.

Primarily we would thank our project guide **Dr. J V Gorabal, Head of the department**, Department of Computer Science and Engineering, Sahyadri College of Engineering and Management for their constant encouragement and support extended throughout.

We owe our deep gratitude to our project coordinator **Dr. Pushpalatha K, Associate professor**, Computer Science and Engineering for her invaluable support and guidance.

We would like to thank our Principal, **Dr. Srinivasa Rao Kunte** for providing all the facilities and a proper environment to work in the college campus.

We are thankful to all the teaching and non-teaching staff members of Computer Science.

We would also like to thank both teaching and non-teaching staff for their invaluable support and guidance.

REFERENCES

[1] N. Majumder, S. Poria, A. Gelbukh and E. Cambria, "Deep Learning-Based Document Modeling for Personality Detection from Text," in *IEEE Intelligent Systems*, vol. 32, no. 2, pp. 74-79, Mar.-Apr. 2017. doi: 10.1109/MIS.2017.23

[2] J. Digman, “Personality Structure: Emergence of the Five-Factor Model,” *Ann. Rev. Psychology*, vol. 41, 1990, pp. 417–440.

[3] F. Mairesse et al., “Using Linguistic Cues for the Automatic Recognition of Personality in Conversation and Text,” *J. Artificial Intelligence Research*, vol. 30, 2007, pp. 457–500.

[4] T. Mikolov et al., “Efficient Estimation of Word Representations in Vector Space,” *Computing Research Repository (CoRR)*, 2013; http://arxiv.org/abs/1301.3781.

[5] S. Mohammad and P. Turney, “Emotions Evoked by Common Words and Phrases: Using Mechanical Turk to Create an Emotion Lexicon,” *Proc. NAACL-HLT Workshop Computational Approaches to Analysis and Generation of Emotion in Text*, 2010, pp. 26–34.

[6] https://en.wikipedia.org/wiki/Text_corpus

[7] https://en.wikipedia.org/wiki/Word2vec

[8] https://whatis.techtarget.com/definition/intelligent-system

[9]https://www.theclassroom.com/difference-between-qualitative-quantitative-evaluation-8281411.html