Dissertation on

"Automatic Question Paper Generator System-Blooms Taxonomy"

Submitted in partial fulfilment of the requirements for the award of degree of

Bachelor of Technology
in
Computer Science & Engineering

UE17CS490B – Capstone Project Phase - 2

Submitted by:

Jyosna S	**PES1201700792**
Megha M	**PES1201700948**
Radhika Sadanand	**PES1201701702**
Roshan Bhaskar Poojary	**PES1201701628**

Under the guidance of

Dr. Uma D
Professor
PES University

January - May 2021

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
FACULTY OF ENGINEERING
PES UNIVERSITY
(Established under Karnataka Act No. 16 of 2013)
100ft Ring Road, Bengaluru – 560 085, Karnataka, India

PES UNIVERSITY

(Established under Karnataka Act No. 16 of 2013)
100ft Ring Road, Bengaluru – 560 085, Karnataka, India

FACULTY OF ENGINEERING

CERTIFICATE

This is to certify that the dissertation entitled

‘Automatic Question Paper Generator System-Blooms Taxonomy’

is a bonafide work carried out by

Jyosna S	**PES1201700792**
Megha M	**PES1201700948**
Radhika Sadanand	**PES1201701702**
Roshan Bhaskar Poojary	**PES1201701628**

in partial fulfilment for the completion of seventh semester Capstone Project Phase - 2 (UE17CS490B) in the Program of Study - Bachelor of Technology in Computer Science and Engineering under rules and regulations of PES University, Bengaluru during the period Jan. 2021 – May. 2021. It is certified that all corrections / suggestions indicated for internal assessment have been incorporated in the report. The dissertation has been approved as it satisfies the 8^{th} semester academic requirements in respect of project work.

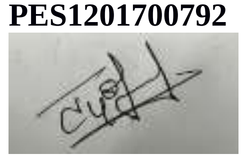
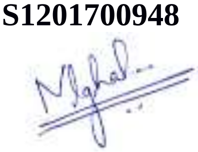
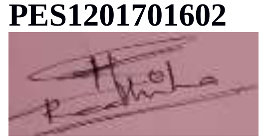
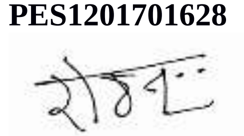
Signature	Signature	Signature
Dr. Uma D	Dr. Shylaja S S	Dr. B K Keshavan
Professor	Chairperson	Dean of Faculty

External VIVA

Name of the Examiners	**Signature with Date**
1.	______________________
2.	______________________

DECLARATION

We hereby declare that the Capstone Project Phase - 2 entitled **“Automatic Question Paper Generator System-Blooms Taxonomy”** has been carried out by us under the guidance of Dr. Uma D, Professor and submitted in partial fulfilment of the course requirements for the award of degree of **Bachelor of Technology** in **Computer Science and Engineering** of **PES University, Bengaluru** during the academic semester January – May 2021. The matter embodied in this report has not been submitted to any other university or institution for the award of any degree.

Jyosna S	**PES1201700792** 
Megha M	**PES1201700948** 
Radhika Sanadand	**PES1201701602** 
Roshan Bhaskar Poojary	**PES1201701628** 

ACKNOWLEDGEMENT

I would like to express my gratitude to Prof. Dr. Uma D, Department of Computer Science and Engineering, PES University, for her continuous guidance, assistance, and encouragement throughout the development of this UE17CS490B - Capstone Project Phase – 2.

I am grateful to the project coordinators, Prof. Rajashree Soman, for organizing, managing, and helping with the entire process.

I take this opportunity to thank Dr. Shylaja S S, Chairperson, Department of Computer Science and Engineering, PES University, for all the knowledge and support I have received from the department. I would like to thank Dr. B.K. Keshavan, Dean of Faculty, PES University for his help.

I am deeply grateful to Dr. M. R. Doreswamy, Chancellor, PES University, Prof. Jawahar Doreswamy, Pro Chancellor – PES University, Dr. Suryaprasad J, Vice-Chancellor, PES University for providing to me various opportunities and enlightenment every step of the way. Finally, this project could not have been completed without the continual support and encouragement I have received from my family and friends.

ABSTRACT

Assessments or exams generally play an essential role in education and as well as in a student's life. It is also considered as the primary indicator of students' learning process. Facing exams is also considered as academic readiness and also learning progress. One of the main forms of the assessment used is exams in learning.

Based on the classification process, a question paper can be redesigned by examiners. This study follows a rule-based approach to achieve multiclass classification. Questions are collected from University Question Papers, Online sources and Textbooks. These questions are related to various subjects under Computer Science like Operating Systems, Databases, Machine Learning, and Networking etc. These questions are preprocessed. To make it more dynamic and self-learning, we replaced the rule-based classifier with an SVM (Support Vector Machine) Classifier.

Then the method of weighting the classes accordingly is used to correctly assign the categories to the questions. This weighting technique is developed using Bloom's Taxonomy verbs. Once the questions are classified, based on user-specified input for bloom's levels expected in the question paper the questions are grouped and a question paper is generated. Results show that the hybrid classifier performs better than the SVM classifier alone.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER-1

INTRODUCTION

In the present world of education, the assessment process is a dominant and vital venture for educational establishments and universities for student performance evaluation. It is a demanding, time consuming and very tedious job for the tutors to come up with worthy questions.

There is much taxonomy that has been developed in order to define questions so that each question has significance for its presence. It might be conversational, accuracy-based, social or knowledge according to one Taxonomy whereas another research says one can classify questions into easy, hard and moderate but there was a need of in-depth classification of the cognitive levels of assessment.

Then in 1956 an American educational psychologist Benjamin Samuel Bloom came up with a Rubric that systematically classified the cognitive levels .This was later named as Bloom’s Taxonomy. This Taxonomy has three ranked models which are cognitive, psychomotor and affective domains.

The cognitive domain is a knowledge based model which is divided into six levels of objectives. These six levels are named as knowledge/remember, analysis/analyze, application/apply, comprehension/understand, evaluation/evaluate and synthesis/create. The affective domain is an emotion based model which is split into five levels. These five levels are responding, receiving, valuing, characterizing and organizing.

The psychomotor domain is an action based model which is divided into seven levels. These seven levels are perception, guided response, set, mechanism, origination, complex overt response and adaptation. Since the main focus here is on examination for educational establishments, the first cognitive (knowledge based) model is considered to be a better fit for this kind of research.

Asking and setting proper assessment questions in order to attain the expected result from the course is a difficult work for the examiner to deal with. So, the research on the whole focuses on categorizing the queries from all the cognitive level domains that follow Bloom’s Taxonomy with better model accuracy and on which technical skills are used to equally cover the questions from all the domain levels in bloom’s taxonomy.

CHAPTER 2

PROBLEM DEFINITION

Examination is an indispensable role in assessing a student's intellectuality as it is the central and foremost approach used. Hence, there is a need for the educators to systematically frame the assessments that include various types of questions that could precisely test the student’s understanding. Bloom’s Taxonomy’s Cognitive Model is being considered one of the best classification methods to effectively classify the learning objectives.

A question falls into more than one category or level which is the common issue faced by all the researchers that worked on this specific project. When this happens it is more difficult to identify the question as to which category does it belong to. Just making sure that the dataset is filled with effective standard questions alone would help a lot while assigning the weight to each taxonomy category. Hence, this particular study presents a solution to the issue in the form of an Automatic Question Paper Generator System/ which makes use of Bloom’s Taxonomy to detect the standard of the question paper generated.

CHAPTER 3

LITERATURE SURVEY AND REVIEW

This section deals with the background check and research carried out by us with respect to classification of Questions and Techniques for the same.

3.1 AUTOMATED ANALYSIS OF THE EXAM QUESTIONS BASED ON BLOOM’S TAXONOMY [1]

In this paper the authors have presented a system which based on a set of rules categorize questions into distinct cognitive categories of Bloom's nomenclature. The rule-based approach makes use of nlp methods to recognize key verbs that aids in the classification of the given input question.

3.1.1 INTRODUCTION

Main steps in the implementation is to design rules in a such a way that the rules should be able to identify the verbs in the input question and assign an appropriate bloom’s level to the question, if the identified verb is in multiple levels then they should refer to the expert's weights for the levels for that particular question to classify it to the correct level.

First before sending the data into this rule-based classifier the dataset of 70 training questions is first preprocessed using techniques in NLP like the removal of stop words, lemmatization, stemming and at last POS-tagging which helps in identifying the verbs in the question that aids in classification . Then this preprocessed data is fed to the rule based classifier in which rules are designed to categorize the input question.

So if the identified verb is unique to a particular level then this level is assigned to that question else if the verb is present in many levels of Bloom’s Taxonomy, then the category weighting technique is used where the experts assign weights to the levels to which the question is likely to belong based on their knowledge . In this way the questions are finally assigned a category which has the highest experts weight.

3.1.2 STRENGTH

During the classification of the questions, the rules used to identify and determine the taxonomy of the questions are based on the verbs and keywords, So there are ambiguous cases where the keywords in a question might belong to more than one category .This raises conflicts and ambiguities like in what level the input question might fall into. For handling this type of scenario the methodology used was **“category weighting”** that helped in deciding, where weights were allocated to conflicting levels.

3.1.3 WEAKNESS

The rules used to categorize the questions into the taxonomy are very general and simple as they consider only the verbs/keywords .In the category weighting the weights are nominated to the conflicting categories by the experts. Making use of the given strategy might result in a lack of consistency because of the diverse types of cognition knowledge of the experts. There are some cases where questions cannot be put into one category which is not handled by the suggested method.

3.2 COSINE SIMILARITY AND WORDNET-BASED CLASSIFIER OF EXAM QUESTIONS [2]

This paper is specifically and mainly centered on classifying the test questions accordingly into its learning objectives in Bloom’s taxonomy. Using nlp methods such as lemmatization, tagging, removal of stop words and tokenization are done before producing the rules that are used for categorization. Similarity based algorithms like wordnet and cosine algorithms are utilized to produce a set of rules that recognizes the category to which the question belongs to and the weight that are given for each question based on Bloom's taxonomy.

3.2.1 INTRODUCTION

Before implementation the test questions were extracted for the pdf documents provided by the user by using the pypdf package, and then by using regular expression they got the questions which were stored in the MySQL database. As a part of implementation, there are two methods that have been carried out that is the word net classifier and cosine based classifier. In wordnet classifier, the first step involves identification of verbs for each level in the taxonomy. Now the questions in MySQL database are converted into a list of tokens for each question after which enchant module was used to correct the tokens.

Then lemmatization is carried out on each word in the list instead of stemming as the lemmatization process leaves a apt word that retains the same semantic from the actual question. After this the lemmatized words in the list are tagged using POS-tagging with the help of which the verbs were extracted where the tag matched with V and W. Now for each verb in the input question the closeness is identified with the taxonomy verbs in each category which as a result gives wordnet score to each category for each input question.

Next in the cosine based classification method the first step is to identify tag patterns for each level in the taxonomy. Now using the classifier based tagger the to get the tag pattern of the input question .Then the input question’s tag pattern is then matched with the taxonomy’s tag pattern and get the cosine similarity between them .

Now the word net similarity and cosine similarity scores are multiple to get the final score based on which the questions were categorized into the top three highest score levels . At last rules were applied to recognize which among the three top levels the question must be put into.

3.2.2 STRENGTH

WordNet similarity algorithm accuracy was primarily dependent on key verbs which could be seen in test paper. Few verbs which were not present in Bloom's taxonomy levels have made the categorization of the test question into different levels. By combining the question tag patterns with cosine similarity it made obvious that questions can be classified into the correct category.

3.2.3 WEAKNESS

A question can be asked in various ways which may also change the tag pattern due to this might be a case where the same question due the various tag patterns might belong to many classes. Recognition of tag pattern was not accurate as the performance of tagger used was not 100% accurate. Because of this some tag patterns failed to merge and the resulting cosine score was zero for cases. That is how this leads to misclassification of the questions.

3.3 USING MODIFIED TF-IDF AND WORD2VEC, QUESTION CLASSIFICATION BASED ON BLOOM’S TAXONOMY COGNITIVE DOMAIN [3]

In order to classify the questions automatically, this research suggests a way by drawing out TF-POSIDF and word2Vec as two different features. The calculation of term frequency-inverse document frequency is one feature based on parts of speech, sequence of allocation a significant weight for essential and important words in the input. The process of classification was enhanced using the prior-trained word2vec as the second feature. Then, the blend of these features was sent into three different classifiers; Logistic Regression, K-Nearest Neighbor, and Support Vector Machine for the classification process.

3.3.1 IDEA BEHIND WORK

This research followed the basic steps involved in a classification process. This process started with the collection of data which is a collection of questions from previous work done or books and several other sources. The next step consisted of various pre-processing techniques like cleaning the data by normalizing it, tokenizing and also stemming. Since the main idea here is to calculate term frequency based on parts of speech, POS-tagging is also done as a preprocessing.

Then comes the most important portion of the research where they extract features required. One required feature here is to have a set of important words that are estimated using the TF-IDF method on POS-tagged words which can be used to develop a weighting method. The second feature is that it should boost the classification by mining semantic features of words based on pre-trained Word2Vec. Word2Vec model is being developed with the use of Google new dataset that contains more than a billion words.

The main idea behind POS-tagging is to give the verbs that occur in the questions a higher priority. However, this does not infer that other words hold no significance in the classification process. Because of this reason, semantic features are also taken into consideration so that words that have similar meaning are aligned together. Combinations of these two models produce a set of vectors that basically represent questions. These are fed into various classifiers discussed earlier. Classifiers used here are SVM, K-Nearest Neighbor and Logistic regression. The metrics used for evaluation here recall, precision, F1 measure and variations of the same. The results and performance of these 3 classifiers examined. LR and SVM achieved better results as compared to KNN.

3.3.2 STRENGTH

The taken out and proposed attribute Term Frequency-Inverse Document Frequency (TF-IDF) is highly dependent on Parts of Speech (POS) and is initiated as TFPOS-IDF. To find and decide the proper and suitable rank for word weight on the basis of Bloom’s taxonomy, various cases have been scrutinized.

$$w_{pos}(t) = \begin{cases} w_1 & \text{if } t \text{ is verb} \\ w_2 & \text{if } t \text{ is noun or adjective} \\ w_3 & \text{otherwise} \end{cases}$$

Figure 1: Weighting algorithm

3.3.3 WEAKNESS

The dataset mentioned in this paper has neither only open-ended questions, which consisted of neither true or false nor multiple choices questions. Usage of weighing the words into different ranks on the basis of its type (verb, noun, adjective) can wrongly classify some questions. This is because the method would rank up insignificant words along with ranking up significant words as it carries from question to question.

3.4 USING MULTI-CLASS TEXT CLASSIFICATION, IDENTIFICATION OF COGNITIVE LEARNING COMPLEXITY OF ASSESSMENT QUESTIONS [4]

The paper focuses on two descriptions of studies to identify the learning complexity of given cognitive questions automatically. The former approach used is the labeled Latent Dirichlet Allocation (LDA). This approach represents texts as arbitrary mixtures over quiescent topics, where everything could be represented by a diffusion of words in the corpus. The second approach uses the Bidirectional Encoder Representations from Transformers famously known as BERT framework that aids the multi-class text classification under deep learning. The algorithm uses prior-trained deep multi-directional presentations from the unprocessed text by jointly setting up either on right and left contexts.

3.4.1 LDA

When we are not sure of what we are looking for (unsupervised) with respect to classification, topic modeling is the method which is preferred. This is because the method discovers unseen concepts and keeps these concepts as a base for classification. LDA is one such method used for topic modeling. Implementing LDA has two diversions. One is something we know which are the words that occur in the dataset as a whole. Second is something we do not know which are words that occur in each topic (Category). In order to find these words, there are few assumptions made. These assumptions are that the numbers of topics are pre-decided and also the words under these topics are correct.

3.4.2 BERT

BERT is one of the most recent developments in the NLP field. This approach uses feature based learning and classification as a base. The Bi-Directional Transformer has two parts of working which are encoder and decoder. The encoder takes input and the decoder predicts the output for the task assigned. This is a very unique approach as the model processes texts both ways hence the name Bi-directional. Also, the model masks few of the words while processing and tries predicting them by keeping in mind the words that occur to the left and right of the masked word. This special feature of BERT makes it credible to be used for small datasets and the main work done here is on the input specifically rather than the quantity of input.

3.4.3 STRENGTH

Most machine learning models train them on the text input in a sequential manner, while the entire sequence is read at a go by the Transformer encoder. This particular feature allows the model to learn the factors of a word keeping in mind all of its surrounding words.

This study uses a wide range of questions collected from different subjects and sources making the training better than other studies. Questions without Bloom’s Taxonomy action verbs, too, were assigned a cognitive level correctly by the algorithms due to the usage of bidirectional approaches in understanding the context of a word.

3.4.4 WEAKNESS

The error occurred in cases where the text structure of the question stem was similar.
What does < subject > mean ? was tagged as remember / knowledge level and What does < subject > do ? was tagged as comprehension / understanding level, as the former has a more specific answer. Unless there is a lot of training data, the accuracy cannot be improved. Each training question needs to be carefully annotated by the annotators so that the training data is correct.

3.5 CLASSIFYING QUESTION PAPERS WITH BLOOM’S TAXONOMY USING MACHINE LEARNING TECHNIQUES [5]

The main goal of the researchers of this paper is to show how Bloom’s Taxonomy can be employed to determine the difficulty and the precision of the given university question paper.
They have used NLP techniques, feature extraction and many ML techniques with a view to classify the questions in the question paper and compare the respective results to determine the best model to serve the purpose.

3.5.1 IDEA BEHIND THE WORK

The first step of implementation is feature extraction to collect the most frequently appearing words that could aid question classification. This study required proper understanding of the ML models for the sake of determining the most suitable ones. This research aided in selecting few of the models that were later used in the later stages.

Before applying these models, an ample amount of work has to be done to the data to make it suitable to be fed into these models. This work includes tokenizing of a dataset containing 1042 labeled questions. Other basic preprocessing is done like Stopword removal and Punctuation removal. Since the data is being fed into ML models, the entire data is being split into test and train. Here intervenes the first step i. e feature extraction. After this, frequencies are determined for both training and testing data. The data is then fed into the studied models.

The models selected and used for this are K-Nearest Neighbours, Neural Networks, Logistic Regression, Random Forests, Linear Discriminant Analysis (LDA), Decision Trees and Support Vector Machine (SVM). These trained models enter into the testing stage where they are being tested against the test data. This gives us an idea as to how accurately the models are trained.

3.5.2 SETUP

A web app is done so as to make the entire system usable. The user enters the questions individually or as a set. These questions are analysed by the built model and are classified accordingly. The category percentage is outputted with respect to the number of questions in the input. If no question belonged to any category, it is shown as 0.00%.

3.5.3 STRENGTH

Since more than one ML models are used for training the data, it is known to be a better approach instead of using any one of the methods individually. Also, this way one can deeply examine the behaviour of these models on feeding the different types of data.

3.5.4 WEAKNESS

At the same time, the fact that models are trained on the dataset and so they work accordingly should not be left behind. This is because; even though the models are enhanced the main factor in better performance would be played by the training dataset.

3.6 FEATURE CONSTRUCTION AND SELECTION USING GENETIC PROGRAMMING AND A GENETIC ALGORITHM [6]

This paper proposes an improvised version of the decision tree classification algorithm by using genetic programming and genetic algorithms. Genetic programming is used for the feature construction and the Genetic algorithm for the feature selection. Classification is one of the vital tasks in data mining that involves the prediction of class value based on information about other attributes present. In recent days, one of the most commonly used techniques for classification is the Decision Tree learning Algorithm.

3.6.1 THE GENETIC ALGORITHM AND PROGRAMMING APPROACH (GAP) ALGORITHM

This approach has two phases. The first phase is Feature Construction.

3.6.1.1 FEATURE CONSTRUCTION

Feature Construction is a process that focuses on discovering hidden relationships between features, conjecturing new hidden features. The basic idea is that To construct features for the decision tree learning algorithm, GP individuals that consist of a number of automatically defined functions (adfs) or separate trees are used. Randomly, A population of 101 genotypes is created. Here, each genotype will have n trees where n is nothing but the number of numeric valued features. It can either be an automatically defined function or an original feature. Any tree in this genotype might contain one or more nodes. The probability of a node being a leaf node is calculated by the given formulae.

$$P_{leaf} = 1 - \frac{1}{(depth + 1)}$$

Figure 2: Probability Formula

Here, depth is the depth of the tree at the current node. So according to this, nodes at depth level 2 will have a probability of 0.67 of being a leaf node whereas root node will have a probability of 0.5 of being a leaf node. During initial creation, similarity between two nodes that belong to the same genotype is restricted. The evaluation is done for an individual by constructing a new dataset with one feature for each tree in the genotype. This dataset is passed to the decision tree learning algorithm whose performance is evaluated using 10-fold cross validation and fitness score is assigned. Comparison for the fitness of the individuals is done after every evaluation after several generations of selection, crossover, mutation and evaluation are performed. This is an evolutionary process and it continues till it arrives at a condition that is set at the beginning. Now termination selection is used to determine the fittest individual which will later be used as a seed for the feature selection stage.

3.6.1.2 FEATURE SELECTION

The fittest individual from the construction stage is evaluated to see if any of the original features do not appear to be used. Required actions are taken for such conditions. This new genotype replaces the initial one and this will further be a base for the second stage. Constructing of a new dataset with one feature for each tree in the extended genotype. Because of Randomization, it is often observed that the fitness score of individuals in the selection stage is less than the score of individuals in the creation stage but solutions must be more vigorous. A population of 101 bit strings is created randomly. After evaluating each genotype, a fitness score is assigned which will later be used in selecting the parents for the next generation. After this, the same steps in the previous stage are repeated and the fittest individual is obtained.

3.6.2 RESULTS

Ten well-known datasets that have more numerical attributes are selected from the UCI Repository. Tenfold cross validation is used for the Performance comparisons. The GAP Algorithm is seen to outperform the decision tree learning algorithm on eight out of the ten datasets that are considered. There are also no datasets on which the GAP has performed worse than the Decision Tree learning algorithm.

3.7 AUTOMATIC QUESTION TAGGING WITH DEEP NEURAL NETWORKS [7]

This paper focuses on automatic question tagging with the knowledge units. The researchers propose two models for automatic tagging of the questions with the knowledge units. Here the multiple-choice questions are taken as input which is used by the mechanisms employed in this paper to intensify the tagging performance with the help of the keywords in the input. The employed model uses deep neural networks to represent the input question with the help of contextual information.it shows how the results of these methods have outperformed many traditional classification techniques.

This paper shows a position based attention model and a keyword based model for tagging the question where both these models focus on the idea that answers help in deducing the knowledge units more effectively as in the case of multiple choice questions.

3.7.1 POSITION BASED ATTENTION MODEL

So the first model proposed by this paper adopts an RNN model to process the text which treats the text as a sequence where for every word it produces a vector based on the former words and itself .By this they get a contextual information of the question by the means of BiRNN. Since the text in the multiple choice question is very little the answers in the blanks play a very important role in determining the tags. So in this method as an initial step they integrate the answers into the blank of the question, then they split the question into tokens and represent it as a one hot representation vector. Now based on the position of the answer the attention is produced for every word. The attention is produced in a way such that more attention is given to the words of answers and other words are given less attention. Then in the second step this one hot representation vector is taken to do word embedding and produce an embedding vector which is later taken by the BiLSTM which produces another vector known as the feature vector. Then this feature vector is sent to adjust the attention along with the position attention based information. Now the attention score is updated using any of the three methods: attention scale adjustment (here attention is the same as the attention given initially), locally connected adjustment (here the attention of the words are updated based on the attention of the previous word). And fully connected adjustment (here the attention of the words is updated based on the attention to all the words in the question). The resultant vector is then normalized using the Softmax function. Then a Relu function is applied to the feature vector so as to get better performance in the tagging performance. Finally the feature vector is used to predict the tag for a given question.

3.7.2 KEYWORD BASED MODEL

The second model proposed by this paper is a keyword based model. This model is very similar to the position based attention model but here instead of just integrating one anwer into the blank all the answers are integrated to the blank in the question so this helps in extracting more contextual information by the BiLSTM . The rest of the steps are similar to the position based model: the feature vector is constructed and the tags are predicted based on this vector.

3.7.3 RESULTS

When the above methods are applied on a dataset we see that the keyword based model has performed a little better than the position based attention model. The results of the proposed model have shown that they have outperformed some of the traditional classification and topic methods like SVM , KNN , MLKNN, LDA SVM .The position based attention model showed an accuracy of 73 percent and the keyword based model showed an accuracy of 80 percent.

3.8 AUTOMATIC EMAIL CLASSIFICATION USING GENETIC ALGORITHM [8]

This paper mainly focuses on classifying the incoming emails as spam and non-spam based on the contents of the email. They propose a simple genetic algorithmic technique for identification of the spam mails. Since the genetic method can be used to observe regularities and converge to solutions they found this method to be useful in observing the regularities in the contents of the emails and filtering the spam emails from a collection of emails.

3.8.1 IMPLEMENTATION AND METHOD

So there are two types of email filtering technique one which filters based on the email address And the other based on the content of the email. Both these methods lack the intelligence and adaptation to the newly emerging spam mails. So the rules set for the spam mail are modified using the genetic algorithm. So in this algorithm they describe a rule set where the weight of the gene corresponding to the email is been calculated and checked with a standard number and based on that the mails are filtered as spam.

In this paper the experiment was carried out on 500 mails where 300 mails were spam and 200 mails were non spam. The minimum score was set to 3 where if the score of the gene of a particular mails is greater than or equal to 3 then it is filtered out as spam otherwise non-spam . The flow of the entire process goes like first a database is constructed which has the spam corpus . First the content of the input mail is extracted and then the words are checked if they belong to the spam corpus . Now if these words belong to the spam corpus then the probability of getting this word in the spam database is calculated with the help of the frequency of the spam word to the total number of words in the mail content . Then words are classified into different underlying categories for example like if the mail contains words like “adult”,”free” and “offer” then adult is classified to a category C1 and the words “free” and “offer” are classified to the category C3.

After classifying the words to different categories the weight of the words is calculated. Then using these weights the weights of the category are calculated by averaging all the weight in the group. These weights are then normalized to the range of 0.000 to 1.000. Then by using the hex representation the weight of the gene is calculated. Then chromosomes are constructed for all the mails the process of genetic algorithm starts and crossover takes place where both multipoint and single point crossover takes place. Then mutation takes place to recover the lost genes. The weights of the gene are checked and accordingly the mails are filtered as spam if they are greater than or equal to three.

3.8.2 RESULTS

The results when observed show improvement as the database corpus size increases so that more words would be there in the corpus and more categories can be formed . The number of datasets available in the dictionary plays an important role in improving the efficiency of the Genetic algorithm. The proposed methodology shows an accuracy of 81 percent.

3.9 A GENETIC FUZZY EXPERT SYSTEM FOR AUTOMATIC QUESTION CLASSIFICATION IN A COMPETITIVE LEARNING ENVIRONMENT [9]

3.9.1 BACKGROUND AND INTRODUCTION

In an assessment process, the correct evaluation of the challenging level of learning material, be it questions or items plays an important role and defines the assessment process, adaptive learning systems or standard setting methods. Nonetheless, there are hardly few studies about the perception and estimation of challenging levels by tutors or teachers.

In a nutshell, though there are not incontrovertible studies about the capacity of teachers when they classify questions by difficulty level, all researchers see an eye to eye on the difficulty of doing this classification. So, an automatic system that alters the challenging level of questions according to the students’ behavior would be a very helpful support tool and a primary component for a practical adaptive learning environment.

Basically, the paper is all about having introduced a telematics tool which is called QUESTOURnament that is integrated into the e-learning platform Moodle that lets teachers to put in order dynamic contests in any knowledge domain. Students must work out exercises or challenges within a given amount of time and as quick as possible, since the scoring function changes with time. To avoid stress and discouragement among the worst classified students, the system should group students by knowledge level so that students with similar skills face off together and solve questions with a challenging level right and suitable for them.

3.9.2 THE EXPERT FUZZY SYSTEM

An expert system has been planned and designed which is genetic fuzzy based and places rules right for each particular case. Fuzzy Model Generator is capable of identifying the different kinds of characteristics of the questions for every single difficulty level which as well as includes a genetic system helps in identifying features. The estimation or evaluation of the difficulty level then takes place in two stages. The Fuzzy Model Generator grasps from the Facts Base that are generated by the students’ response patterns and powerfully creates the categorization rules and the fuzzy sets of the input variables for the certain data during the first phase. The fuzzy expert system judges the difficulty level of each question in the second stage.

3.9.3 THE GENETIC SYSTEM

The purpose and target of the genetic algorithm for the put forward system is to give rise to groups of crisp sets that characterize the students’ responses for three challenging and difficulty levels: easy, average and difficult. Speaking of the steps taken during inception, the suggested system begins from scratch. Considering some data about the interaction of the students with the tool QUESTOURnament and the given initial difficulty level estimated by the tutor, it grasps both the adequate membership functions with their linguistic values as well as the fuzzy rules. Furthermore, for a question response patterns not only depend on the question itself but also the student's answering behavior, be it knowledge level and persistence are considered. The algorithm also integrates niching methods like sharing in order to encourage diversity and to be capable of categorizing each challenging level by different groups.

3.9.4 RESULTS

Speaking of, in order to inspect and infer the efficiency of this expert system, this one has been experimented with real data from a contest developed with the QUESTOURnament tool in an undergraduate course of Diploma in Telecommunications Engineering at the University of Valladolid (Spain). There is no defined solution with accuracy as expert systems are expected to yield at close to human expert levels and to answer problems without a defined correct solution; they should honestly be inferred against human experts.

3.10 SENTIMENT ANALYSIS OF TWEETS USING SVM [10]

Common people’s opinions and feedback have always proven to be the best resource for organizations and companies. As we all know, social media is the emerging trend among everyone and it makes way for unsupervised analysis for various aspects for which companies must depend on unusual, open to error and very time consuming methods earlier. All of this falls under “Sentiment Analysis”. In this paper, Support Vector Machine (SVM) is used for sentiment analysis in Weka. It is a widely used technique for textual polarity detection. Performance is evaluated using comparative analysis using Precision, Recall and F-Measure.

3.10.1 INTRODUCTION

Three main techniques have been used for sentiment analysis that are machine learning based, Lexicon based and their Hybrid. Lexicon based approach comprises a predefined dictionary which includes weightage of words and their sentiment orientation to determine the sentiment inclination of textual data. In a machine learning technique, some of the dataset with the pre identified output class is given to the algorithm in order to make the rules and then the real input data (test data) is given. The hybrid will have both of the above two implemented together. SVM has proved to be one of the best methods for text categorization.

3.10.2 METHOD

To analyze the accuracy and performance of SVM, two Pre-labeled twitter datasets have been used. Polarity erection is done using SVM and then compared with the labeled dataset. Weka is a widely used tool used for analysing the working of ML algorithms and data mining which is used in this particular research.

Two pre-labeled datasets of tweets are used in this research. First dataset contains a total of 7156 tweets and the second one contains a total of 3884 tweets. These tweets are basically categorized and labeled into 6 classes. These classes are Irrelevant, Very Positive, Slightly Positive, Neutral, Slightly Negative and Very Negative.

Pre processing is a very important stage in a classification algorithm. This is a phase where the dataset is normalized and prepared for the classification process such that smooth execution of the algorithm is ensured. In this research, default parameters for preprocessing offered by Weka tools are used.

SVM runs on the data that is normalized. Analysis of performance is done by comparing the output polarities with the already labeled polarities. As said earlier, Performance is evaluated based on comparative analysis between three parameters which are Recall, Precision and F Measure. Precision is calculated using the formulae

$$Precision = \frac{TP}{(TP + FP)}$$

Figure 3: Precision Formula

where TP is for correctly classified and FP is for wrongly classified sentences. Recall is calculated using the formulae

$$Recall = \frac{TP}{(TP + FN)}$$

Figure 4: Recall Formula

where FN is for non-classified sentences. F Measure is calculated using the below formulae.

$$\mathrm{F-measure} = \frac{Precision * Recall * 2}{(Precision + Recall)}$$

Figure 5: F-measure Formula

3.10.3 RESULTS

The first dataset that has data regarding self driving cars showed a Precision of 55.8 %, Recall of 59.9% and F-Measure of 57.2%. The second dataset had an average of 70% for all the three parameters approximately. The further work could be to do a research on which machine learning approach would work best on which type of datasets.

3.11 CONCLUSION

- In [1], a system where the questions are categorized into Bloom’s levels by using a set of rules which designed based on the verbs present in the question and category weighting method is used in a case where the verb belongs to more than one level of the taxonomy so that the question is assigned to the level which more weightage.

- In [2] ,classification of the questions are done based on combined results of WordNet and cosine classifier scores . The WordNet similarity gives the closeness score between the verbs in the question and the taxonomy verbs whereas the cosine similarity gives the closeness score between the input question tag pattern and the taxonomy tag pattern.

- Paper [3] proposes a procedure to classify open-ended questions on the basis of Bloom's cognitive levels. It uses a modified version of TF-IDF i.e TF-POSIDF in calculating frequency of words and later uses word2Vec word embedding model to align semantically similar words as vectors together.

- In [4], a feasible solution is provided with a computational approach that includes deep learning and ML techniques that are instructed over an exciting variety of assessment questions. Also, questions without action verbs were classified with the help of WH- words present in them.

- In [5], performance of various ML techniques is observed by feeding them with a dataset of 1024 questions. The study concluded with the fact that ML techniques like Logistic regression and Linear discriminant analysis showed better results compared to others.

- In [6], GAP Algorithm is used for the classification process that uses decision tree algorithms. There are two stages, one is feature construction and the other is feature selection. This particular technique performs well since it uses a hybrid of genetic algorithm and decision tree algorithm.

- In [7], It is a position based attention model and a keyword based model for tagging the question where both these models focus on the idea that answers help in deducing the knowledge units more effectively as in the case of multiple choice questions. Keyword Based model performs better than the position based model as it concatenates all the possible answers to the question and then analyzes it to find the appropriate tag.

- In [8], They propose a simple genetic algorithmic technique for identification of the spam mails. It mainly focuses on classifying the incoming emails as spam and non-spam based on the contents of the email. An additional weighting technique is used that significantly enhances the accuracy.

- In [9], A genetic fuzzy expert system is used to classify questions. The purpose and target of the genetic algorithm for the put forward system is to give rise to groups of crisp sets that characterize the students’ responses for three challenging and difficulty levels: easy, average and difficult.

- In [10], Support Vector Machine (SVM) is used for sentiment analysis in Weka. It is a widely used technique for textual polarity detection. Performance is evaluated using comparative analysis using Precision, Recall and F-Measure.

CHAPTER 4

PROJECT REQUIREMENTS SPECIFICATION

4.1 INTRODUCTION

This section describes the requirements needed for developing the automatic question paper generation system. It includes all aspects of requirements like functional and nonfunctional requirements.

4.1.1 PROJECT SCOPE

Assessments or exams generally play an essential role in education and as well as in a student’s life. It is considered as the primary indicator of students' learning process. Bloom’s Taxonomy cognitive model containing Knowledge, Comprehension, Application, Analysis, Evaluation and Synthesis is considered the best for classifying. The research on the whole focuses on classifying the queries on which technical skills are used to equally cover the questions from all the domain levels in bloom’s taxonomy. This study follows a rule-based approach to achieve multiclass classification. Coming up with a desired good accuracy and also a better one than all the research studies has been the notion.

4.2 PRODUCT PERSPECTIVE

Examination plays an indispensable role in assessing a student's intellectuality as it is the central and foremost approach used. Bloom’s Taxonomy Cognitive Model is being considered one of the best classification methods to effectively classify the learning objectives.

To design an Automatic Question Paper Generation System which makes use of Bloom’s Taxonomy to detect the standard of the question paper generated.

4.2.1 PRODUCT FEATURES

The system allows the user to enter the question bank and also the percentage of bloom’s cognitive complexity levels that has to be included in the question paper being generated. The system must then be able to classify the questions into different Bloom's levels. It should then generate a question paper based on percentage input given by the user.

4.2.2 USER CLASSES AND CHARACTERISTICS

•Institutions can use our application conducting admission tests.

•Tutors/Teachers can use it to test their students cognitive level of thinking.

•Our application can also be used by test generator platforms so that they can generate better test papers.

4.2.3 OPERATING ENVIRONMENT

Working PC with RAM > 8GB with Windows or Ubuntu OS.

4.2.4 GENERAL CONSTRAINTS, ASSUMPTIONS AND DEPENDENCIES

Assumptions

• The questions entered are true and meaningful.

•Questions belong to the same subject.

•Question bank covers all levels of Bloom’s taxonomy

Dependencies (Software)

•Python with suitable libraries (Dask and FPDF)

•Questions dataset

•Classification is dependent on rules.

4.2.5 RISKS

•Poor data Quality

•Wrong keywords leads to misclassification.

4.3 FUNCTIONAL REQUIREMENTS

● The system must accept no. of questions and questions(only text and double the number of questions expected in the paper) accordingly as input.

● The system must be well-trained so as to classify each question accordingly.

● The questions accepted as input must be classified into any of the classes(bloom’s taxonomy levels).

● The system must generate a high-quality Paper (pdf/word) as output.

4.4 EXTERNAL INTERFACE REQUIREMENTS

4.4.1 USER INTERFACES

● Using smart UI dimensions for any screen size. The resolution of the screen size should be divisible by 8. The GUI standards expected are clear and simple, responsive, maintainable, creative and familiar.

● The screen layout must be visually good in terms of space, grids and size. Standard functionalities might include select menus; drop down menu, help, about us.

● Error message displayed as a pop up for any invalid inputs.

4.4.2 SOFTWARE REQUIREMENTS

● Python 3

● SQlite 3 or higher

● Django 3.1.4

● Windows 10 or Ubuntu until 20.04

● Python libraries, Data processing libraries.

4.5 NON FUNCTIONAL REQUIREMENTS

4.5.1 PERFORMANCE REQUIREMENT

● The system must perform correct classification.

4.5.2 SAFETY REQUIREMENTS

● Broken authentication and session management.

● Prevent sensitive data exposure.

4.5.3 SECURITY REQUIREMENTS

● Usage of cookies.

● Prioritizing vulnerabilities and working on them.

CHAPTER 5

HIGH LEVEL DESIGN

5.1 INTRODUCTION

This section formally documents the High-level Design for the project “Automatic Question Paper Generator” which focuses on generating question paper from the given questions and determining the standard of the question paper using bloom’s taxonomy rules.The idea is to make the question paper generating process completely automatic, efficient with as little response time as possible so that the institution/ teachers can save time and indulge themselves in other activities.

5.2 CURRENT SYSTEM

There are some automatic question generating systems out there, some of them are random question paper generation which has less use in today’s education system since the question paper should have defined a set of questions based on the cognitive skills required to solve it. And some have implemented the classification of the questions based on bloom’s taxonomy. But there are weaknesses in those products.

The problems in some of those include

- The rule used in classification is very simple as they consider only verbs/ key words which can lead to wrong classification.
- In the category weighting the weights are assigned by the experts manually which can lead to inconsistency due to the variety of background knowledge of each experts
- Error occurs when the structures of the questions were similar

5.3 DESIGN CONSIDERATIONS

This section gives a brief idea about the design approach chosen for the system.

5.3.1 DESIGN GOALS

- The application’s goal is to make the best possible and suitable question paper by using the questions given by the user
- The efficiency, accuracy, speed must be better than already existing models
- Reduce the response time.

5.3.2 ARCHITECTURE CHOICES

The architecture has a front-end web interface with an option to upload documents full of questions. The application sends the document to the backend service or model which has classifier to extract the keywords to classify every question on the learning level basis

The keywords that make questions fall under two or more than two categories will be made undergo a weighting technique which will make sure that the questions get classified under one category. As the ambiguity will be gotten rid of through weighting technique there will a better model accuracy.

5.3.3 CONSTRAINTS, ASSUMPTIONS AND DEPENDENCIES

5.3.3.1 ASSUMPTIONS

- The questions entered are true and meaningful
- For classifying the cognitive level of a given question is that it should belong to only one particular class.

5.3.3.2 DEPENDENCIES

- Python with suitable libraries (Dask and FPDF)
- Questions dataset which should be provided by the user

5.4 HIGH LEVEL SYSTEM DESIGN

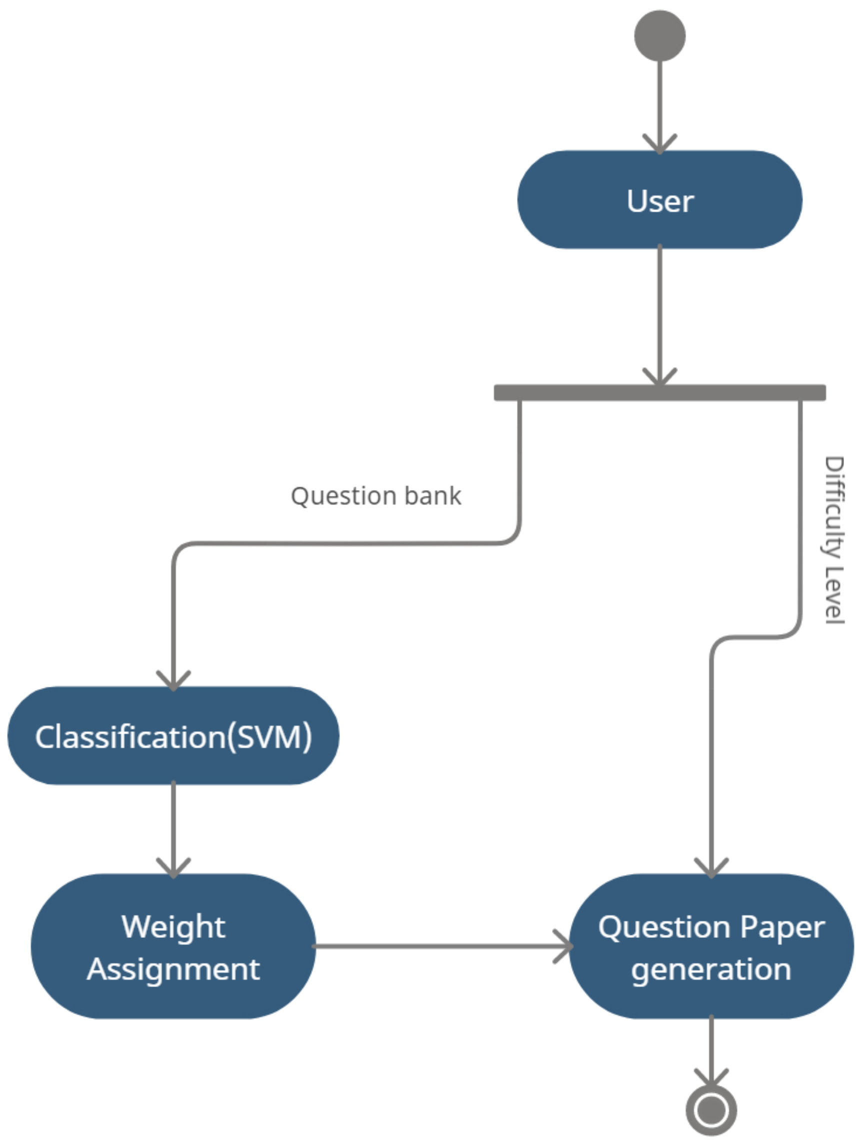


Figure 6: High level system design

The user needs to provide a question bank with the questions for the paper he’s preparing and the difficulty level he wants in that question paper. By difficulty level we mean that the percentage of questions from all 6 bloom’s taxonomy levels he wants in his question paper. After this the system will do the rest provided that the inputs are valid.

The system will take the input and classify the questions to their respective bloom’s taxonomy level, for those which got classified to multiple categories, we have prepared a weighting algorithm which makes sure that every question is classified to only one category.

After classification is over the system will create a question taking the difficulty level selected by the user to consideration. A final question paper is created which contains the number of questions specified by the user and questions from each Bloom’s levels are taken according to the percentages given by the user.

5.5 DESIGN DESCRIPTION

5.5.1 MASTER CLASS DIAGRAM

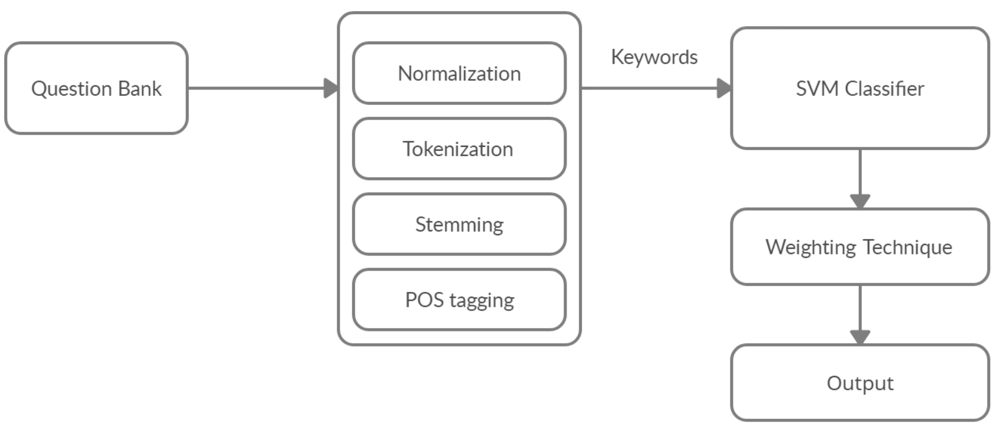


Figure 7: Master class diagram

5.5.2 REUSABILITY CONSIDERATIONS

Reusing the dataset of question to train the model for an accurate classification. By reusing the dataset we mean that we are using the training dataset to generate weights for Bloom’s levels into which the question from the user question bank is classified into by SVM classifier.

5.6 STATE DIAGRAM

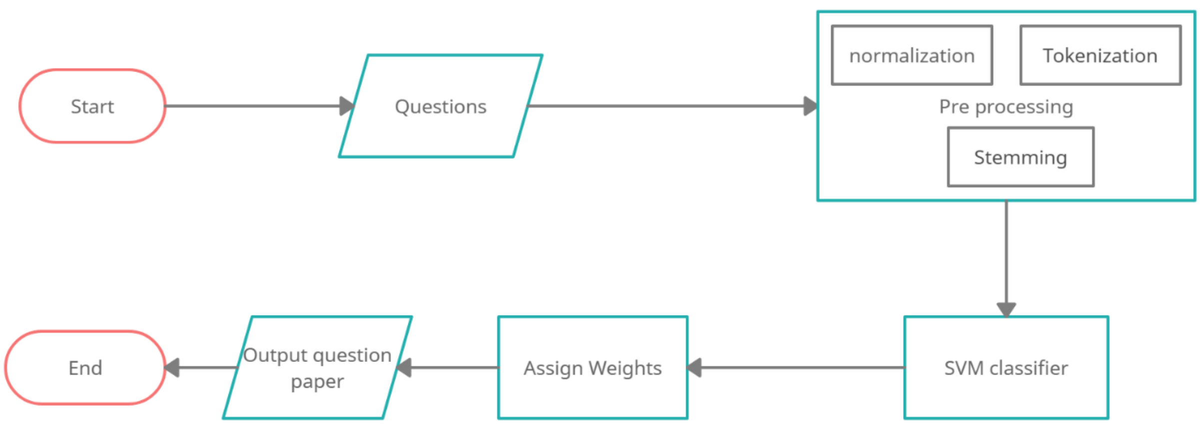


Figure 8: State Diagram

The entire process starts when the user gives the application a question bank input. After validating the given input the questions undergo pre-processing, normalization, tokenization and stemming takes place after these steps the verbs we gathered will undergo svm classification and get classified to their respective bloom’s taxonomy class. But here some verbs may have been classified into multiple categories. So we have implemented a weighting technique which will classify those multiple category verbs to a single category. now since we have all the questions already categorised we take the input from the user regarding the percentage of every bloom’s taxonomy level he wants in his question paper. Then we provide him with the question paper.

5.7 USER INTERFACE

The user will have an option to enter the questions dataset and an option to choose the difficulty level, after uploading the documents and selecting the application will do it’s work and present the user with the final question paper.

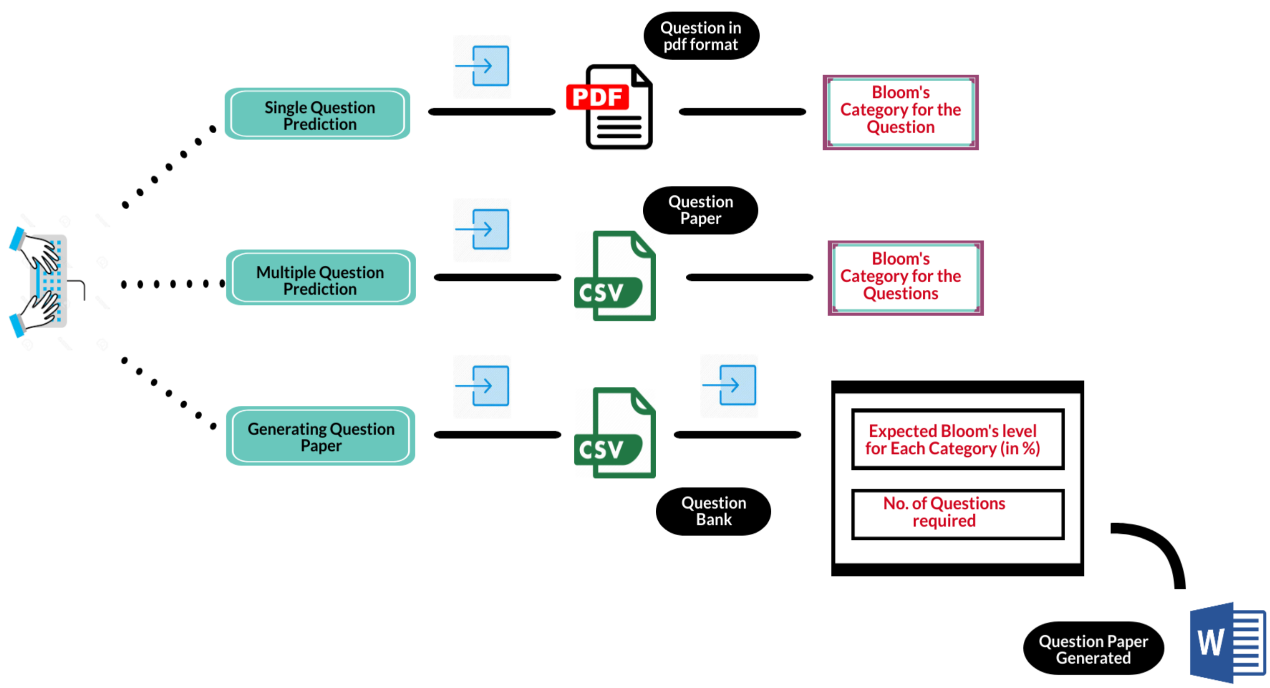


Figure 9: Working of system in a user’s view

5.8 PACKAGING AND DEPLOYMENT DIAGRAM

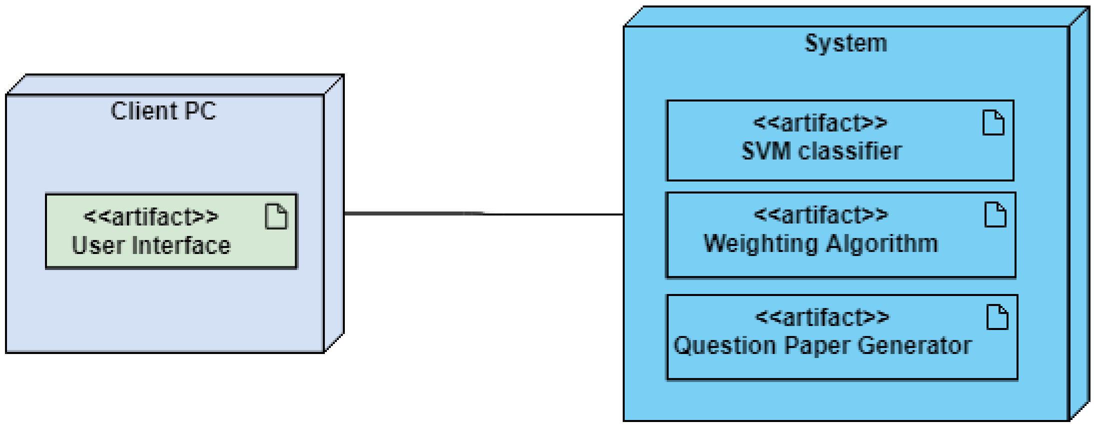


Figure 10: Packaging and Deployment Diagram

CHAPTER 6

LOW LEVEL DESIGN

6.1 INTRODUCTION

6.1.1 OVERVIEW

This document has got the information about the project “Automatic Question Paper Generator” which focuses on generating question paper from the given questions and determining the standard of the question paper using bloom’s taxonomy rules.

6.1.2 PURPOSE

Low-level design is made based on the high-level design.The main purpose is to serve the internal logical design of the Automated Question paper Generator system that has been built.

6.1.3 SCOPE

- Institutions can use our application with the help of which they can set a standard question paper for conducting admission tests.
- Tutors/Teachers can also use this application to create a question paper to test their student’s cognitive level of thinking which actually helps them in saving their time.
- Our application can also be used by test generator platforms so that they can generate better test papers in such a way that they cover all cognitive levels instead of randomly choosing the questions from the question bank given by their users.

6.2 DESIGN CONSTRAINTS, ASSUMPTIONS, AND DEPENDENCIES

6.2.1 ASSUMPTIONS

• The questions entered are true and meaningful.

• Questions belong to the same subject.

• Question bank covers all levels of Bloom’s taxonomy.

6.2.2 DEPENDENCIES (SOFTWARE)

• Python with suitable libraries (Dask and FPDF)

• Questions dataset

• Classification rules

• classifiers and algorithms

6.2.3 RISKS

• Poor data Quality

• Wrong keywords leads to misclassification.

6.3 DESIGN DESCRIPTION

The user should be able to add percentages for levels easily on the interface.The system must perform correct classification according to Bloom’s taxonomy .Classification should be done properly even in the case of ambiguity,as to get rid of ambiguity in classification,weights are assigned to keywords or verbs according to algorithm.

6.3.1 MODULE

Module consists of user interface, primary classification, secondary classification using an algorithm and in the end, generation of questions file according to the given learning levels.

6.3.1.1 DESCRIPTION

User Interface: Automatic Question Paper generator interface uses the SQLite. For using smart UI dimensions for any screen size, expected GUI standards are clear and simple. Standard functionalities may include selecting menu, drop down menu, help and about us. A Pop - up notification is displayed for invalid inputs.

Classification Models: In the very first step of classification, the svm classifier converts the words to vectors as a primary step of classification. As a second step of classification, a weighting algorithm based model has been chosen and implemented. Weights of the keywords are compared and for keyword with highest number of occurrence is given the maximum weight and classified under specific learning level. Dask library has been used which is a python library that can hold larger datasets on a single CPU and one more library, that is FPDF ,used for generating pdfs.

6.3.1.2 USE CASE DIAGRAM

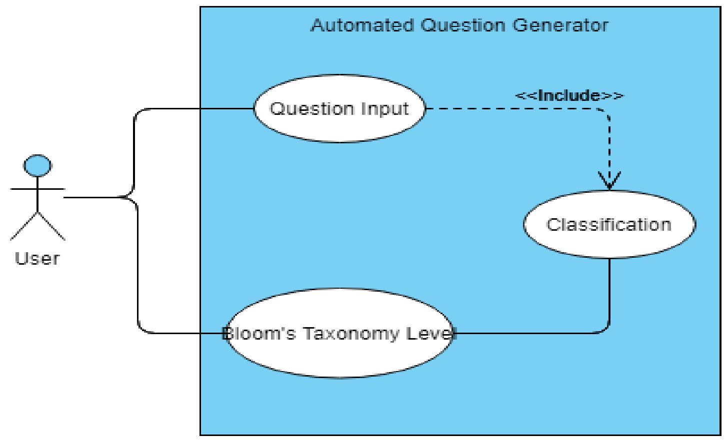


Figure 11: Use Case Diagram

6.3.1.3 MASTER CLASS DIAGRAM

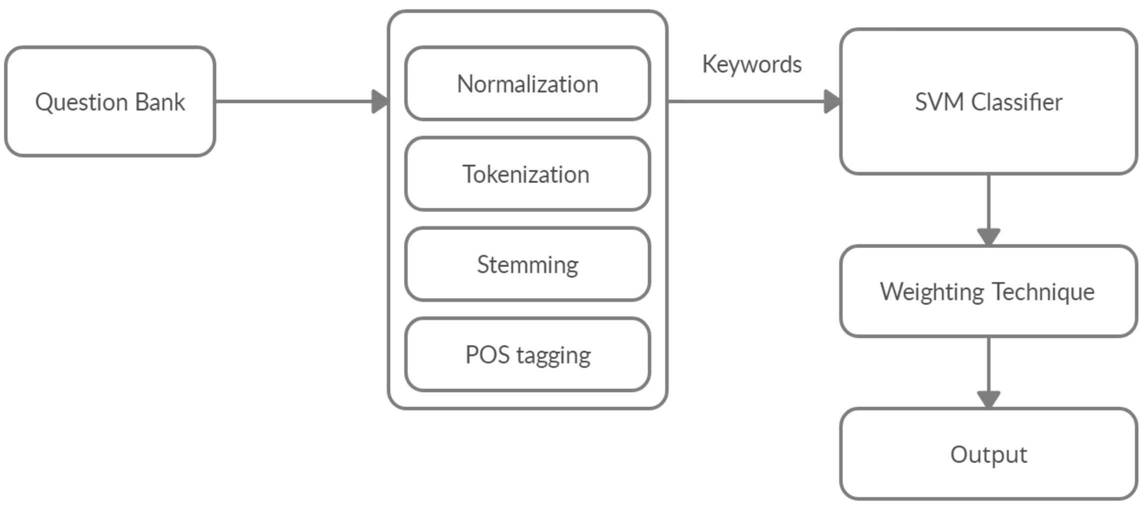


Figure 12: Master Class Diagram

6.3.1.4 SEQUENCE DIAGRAM

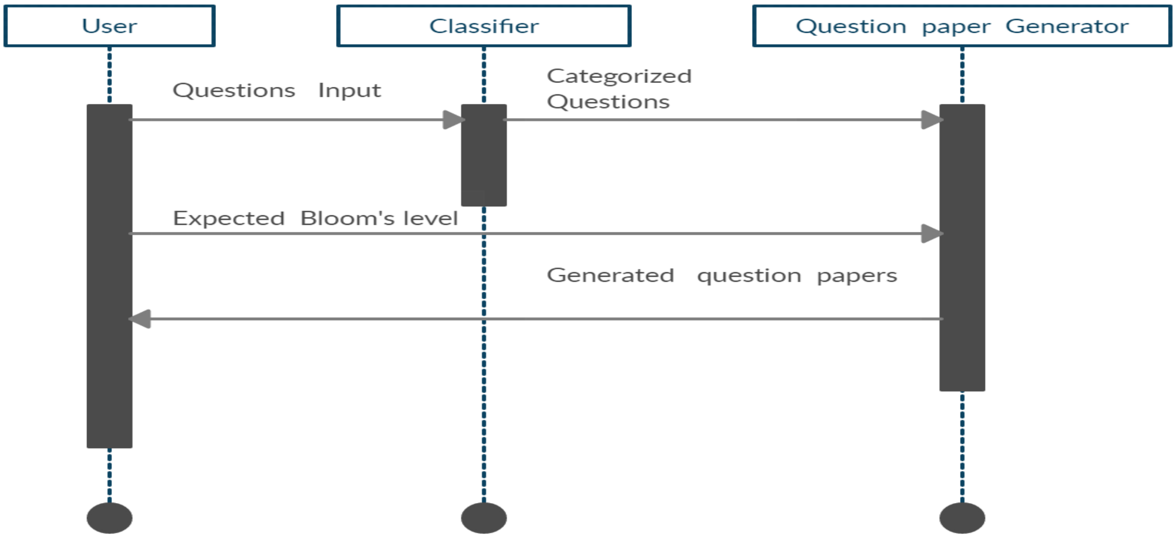


Figure 13: Sequence Diagram

As the diagram shows, the user gives the question paper as an input to the classifier, questions falling under multiple learning levels along with questions falling under only one category will be fed to the weighting algorithm which assigns weights to verbs/keywords. On the basis of a keyword with highest assigned weight, classification will be done. The final classification can have a question under a single category. In the end, a document of accurately classified questions will be generated and given as the output to a user.

6.4 PROPOSED METHODOLOGY / APPROACH

The approach that has been already designed has few constraints like grammatically true and meaningful sentences ,Question bank covering all levels of Bloom’s taxonomy and questions belonging to the same subjects in questions datasets to overcome which ,more advanced implementation techniques such as auto correction can be included. As for questions belonging to the same subjects constraint, can be overcome by using some approach before SVM classification. Question bank covering all levels of Bloom’s taxonomy , can be a no matter ,as long as it has the same keywords used to train classifiers and algorithms.

6.4.1 ALGORITHM AND PSEUDOCODE

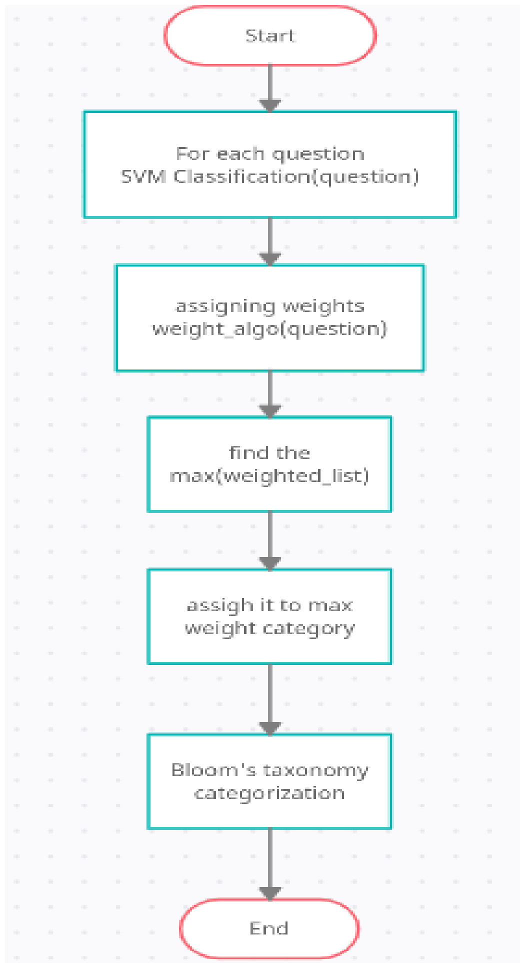


Figure 14: Steps for classification

Pseudo code :

for each question in question bank:

list=SVM_classsification(question)

weighted_list=weight_algo(question,list)

find the max (weighted_list)

classify that question to the max weight category

write the question into the that particular category dataset

6.4.2 IMPLEMENTATION

● The research on the automatic question paper generation focuses on classifying the queries on which technical skills are used to equally cover the questions from all the domain levels in bloom’s taxonomy including data pre-processing, data collection and data labeling.

● In the first step of classification, As verbs play an important role in this particular study, the svm algorithm converts the words to vectors and such that similar words have similar vector representation. The questions falling under multiple learning levels are supposed to be taken care of with better model accuracy excluding the ones falling under only one .learning level.so as a second step of classification that holds much more of importance among all the steps, a weighting algorithm based model has been chosen and implemented that would help us take care of, maintain and regulate the requirements, results, specifications and conclusions.

● In the second step classification, keywords that describe each learning level are collected as many as possible and a large list is made out of those available keywords. From the training data set, counts of each keyword are taken in a dictionary. The more number of times a keyword occurs, the more and varying count value it will hold in the dictionary. As there are totally six learning levels in Bloom’s taxonomy, six dictionaries are maintained .One more dictionary is created and maintained to hold the count of a specific keyword, as in it will have count and the keyword classified into specific levels.

● Keywords that hardly occur will have count zero and those that occur most of the times will have higher count value. As the questions get processed one by one from the input file like one at a time, it will be made to undergo tokenization which helps model to extract keywords ,at this point of time if any keyword that belongs to different learning levels is present, a weight which varies from zero to one is considered , as in the probability of a specific keyword occurring in the training dataset obtained by dividing the weight of keyword count from one of the six different dictionaries by weight of the keyword count from the larger dictionary. So as to classify, weights of the keywords are compared and that learning level is given the opportunity to have the question under it whose keyword has the highest weight and assigned to different learning levels accordingly.

● Through the full-fledged user interface a user can input and output the question papers according to their requirements.

6.4.3 RESULTS

● As for the SVM classifier alone, the accuracy found to be around 65-68% which is also an average accuracy.

● To improve the accuracy, the primary classifier was blended with the one more weighting algorithm based model classifier which led the accuracy up to around 80% .

● The hybrid model which is SVM combined with weighting gives better accuracy and performance than SVM alone.

CHAPTER 7

SYSTEM DESIGN

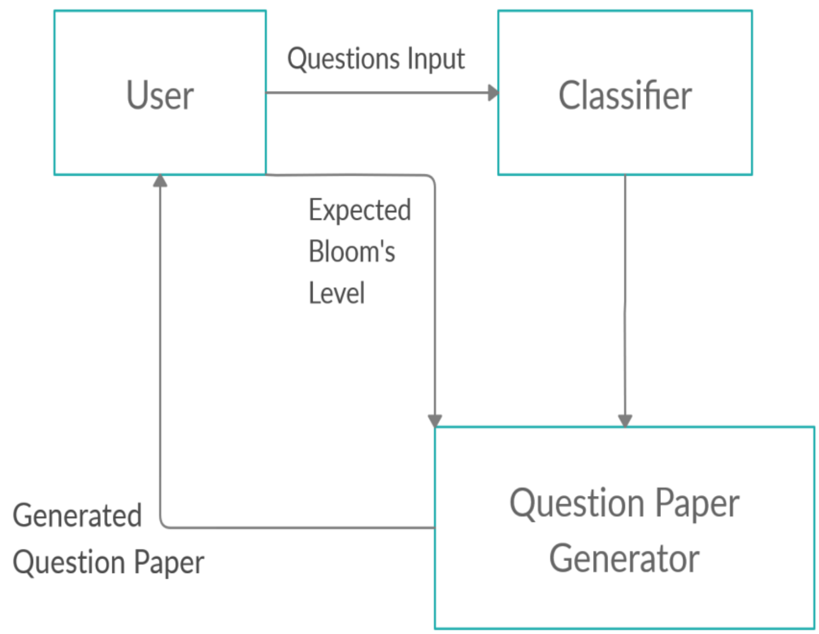


Figure 15: System Design

● This study needs questions that are labeled correctly according to bloom’s taxonomy levels.

● The system must perform correct classifications and even in the case of ambiguity. The architecture has a front-end web interface with an option to upload Excel sheets of required number of questions. The user should be able to add percentages for bloom’s taxonomy cognitive levels easily. The application sends the sheet to the backend service or model which has a classifier to extract the keywords to classify every question on the learning level basis.

● Speaking of classifiers, as an initial and primary step of classification SVM machine learning algorithm has been used which is also used for regression challenges sometimes.

● As verbs play an important role in this particular study, svm algorithm converts the words to vector and such that similar words have similar vector representation. SVM classifier is trained using the training data .Questions are tested with the classifier that is trained

● The keywords that make questions fall under two or more than two categories will be made to undergo a weighting algorithm based model.

● A weighting technique that classifies the questions into one category using confidence level to be unambiguous enough to get classified under only one category. As the ambiguity will be gotten rid of through weighting algorithm based model, there will be a better model accuracy than other ML techniques.

● Once the second classifier categorizes the questions accurately, a sheet of questions with specified allotted marks for different learning levels is delivered as a question paper to users.

CHAPTER 8

PROPOSED METHODOLOGY

● Bloom’s taxonomy cognitive model containing knowledge, comprehension, Application, Analysis and Evaluation is considered the best for the study as an initial, basic and primary step.

● Primary and basic classifier is trained using the training data .Questions are tested with the classifier that is trained.

● SVM classifier ,a supervised machine learning algorithm based model for initial classification is to be implemented and questions to be classified using SVM classifier taking the questions as input from a sheet file with average accuracy and for better accuracy, another classifier is used to get rid of the ambiguity or multi-class classification.

● The objective of the support vector machine algorithm is to find a hyper plane in an N-dimensional space (N — the number of features) that distinctly classifies the data points. Support vectors are data points that are closer to the hyper plane and influence the position and orientation of the hyper plane. Using these support vectors, the margin of the classifier is maximized.

● In the second step of classification, a weighting-based algorithm is used which holds much more importance as it reduces the classes predicted by the SVM classifier to a single Bloom’s class.

● Once the classification is done, a sheet of questions with specified allotted marks classified into different learning levels is delivered to users through user interface.

CHAPTER 9

IMPLEMENTATION AND PSEUDOCODE

This study needs a detailed understanding of the Bloom's Taxonomy levels. Below shown figure depicts the hierarchical model of the cognitive model of Bloom's Taxonomy starting from remember (being the least) to create (being the highest). The figure also shows what each of these levels signifies.

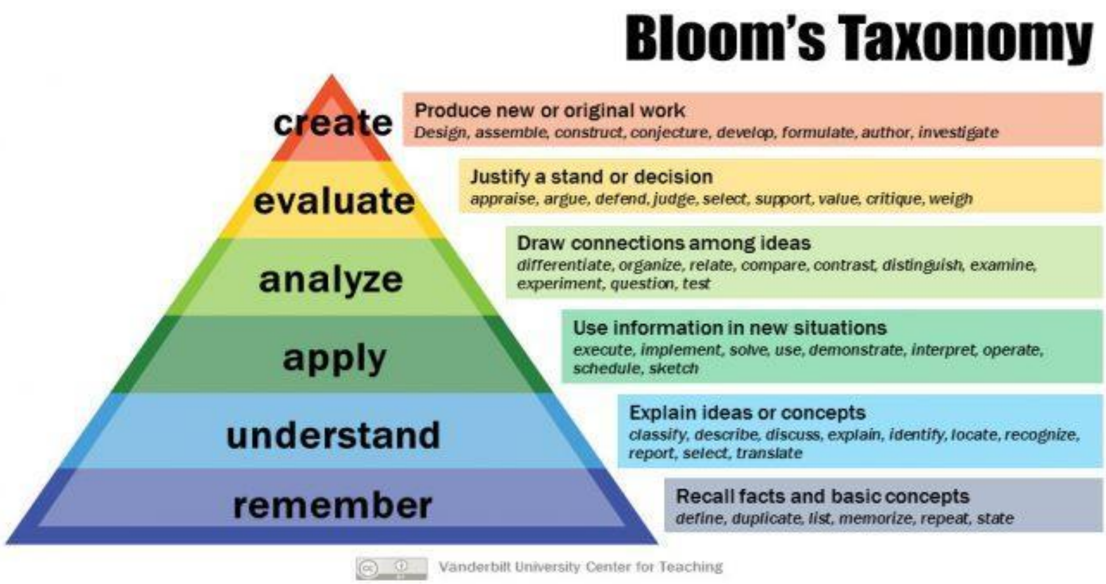


Figure 16: Bloom's Taxonomy Levels

9.1 DATA CREATION

As seen above, the main requirement with respect to data is to have a proper dataset which consists of a set of questions that are labeled according to the above mentioned bloom's taxonomy levels.

The questions for creating the dataset are collected from University Question Papers, Online sources and Textbooks. These questions are related to various subjects under Computer Science like Operating Systems, Databases, Machine Learning, Networking etc., Here the data set has two columns; one is questions columns and another one is its respective classified classification column. So when we look at the very first question ,the question starts with or has a keyword which is ‘what’, which has been classified under knowledge domain level, just like this remaining keywords like recite, define, name etc., are classified into knowledge domain level. Just like this the classification goes on with different key words in different questions.

The dataset consists of 1100 labeled Questions.

Name the factors that affect the performance of the network?	Knowledge
Define the terms Unicasting, Multiccasting and Broadcasting	Knowledge
Describe at one disadvantage of a peer to peer network.	Comprehension
How can you manage a network using a router?	Application
One way of securing a network is through the use of passwords. What can be considered as good passwords?	Analysis
What happens when you use cables longer than the prescribed length?	Knowledge
You need to connect two computers for file sharing. Is it possible to do this without using a hub or router?	Evaluation
When you move the NIC cards from one PC to another PC, does the MAC address gets transferred as well?	Analysis
How are IP addresses arranged and displayed?	Knowledge
When troubleshooting computer network problems, what common hardware-related problems can occur?	Analysis
How does dynamic host configuration protocol aid in network administration?	Application
Explain profile in terms of networking concept?	Comprehension
How do bridges pass spanning tree information between themselves?	Application
Describe a recent short term stressful situation and how you managed it.	Comprehension

Figure 17: Training Dataset

9.2 PREPROCESSING

STEPS:

- The entire data is converted into lower case to make the further steps of data processing easier
- Removal of Punctuations: Questions usually consist of Question marks, Hyphens, colons and semicolons. These are unwanted symbols which are of no use for classification processes.
- Removal of stopwords: This is an important step in Pre-processing as Questions consists of many stopwords that occur commonly among all the Questions and do not add any unique information for the classification process. The nltk corpus of stopwords includes wh- words and similar words which actually are important for the classification in this

case as we deal with questions. So a list of these words is excluded from the process of removal.

- Tokenization: Before feeding the questions to the classifier, tokenization is done as a part of pre-processing.

9.3 CLASSIFIER

As mentioned in the earlier sections, the main implementation lies in the classifier that classifies the questions into the respective Bloom’s categories. This proposed classifier is a hybrid of two different techniques. The first technique is a Machine Learning approach whereas the second technique is a Weighting Algorithm. Each of this technique’s implementation is described in the following section.

9.3.1 SVM CLASSIFIER (ML Approach)

The main objective of this algorithm is to find a hyper plane that distinctly classifies the data points, in an N-dimensional space where N is the number of features.

The labeled dataset consisting of 1100 questions is split into training and testing data. This step is done since it's a prominent step in any classification process that includes a machine learning algorithm. The training data (which consists of both questions that are preprocessed and their respective categories) is sent to the SVM(Support Vector Machine) Classifier. The SVM classifier is trained with this data and the resulting trained classifier is stored in an intermediate file called *model.joblib*. This is done for ease of use in the further steps.

```
text_clf.fit(X_train, y_train)

predicted = text_clf.predict(X_test)

dump(text_clf,'model.joblib')
```

Figure 18: SVM classifier training

When tested, the questions were classified into more than one Bloom’s levels by the SVM Classifier. So, we realized that there is a need for an additional technique to classify these questions into one single category appropriately. For this sake, a weighting algorithm is introduced.

9.3.2 WEIGHTING ALGORITHM

Each of the Bloom’s levels have a set of verbs that occur frequently in the questions that belong to the respective levels. Few of the verbs are listed below categorically.

Bloom’s Taxonomy Verb List					
COGNITIVE DOMAIN					
Knowledge	**Comprehension**	**Application**	**Analysis**	**Synthesis**	**Evaluation**
cite	add	acquire	analyze	abstract	appraise
define	approximate	adapt	audit	animate	assess
describe	articulate	allocate	blueprint	arrange	compare
draw	associate	alphabetize	breadboard	assemble	conclude
enumerate	characterize	apply	break down	budget	contrast
identify	clarify	ascertain	characterize	categorize	counsel
index	classify	assign	classify	code	criticize
Indicate	compare	attain	compare	combine	critique
label	compute	avoid	confirm	compile	defend
list	contrast	back up	contrast	compose	determine
match	convert	calculate	correlate	construct	discriminate
meet	defend	capture	detect	cope	extimate
name	describe	change	diagnose	correspond	evaluate
outline	detail	classify	diagram	create	explain
point	differentiate	complete	differentiate	cultivate	grade
quote	discuss	compute	discriminate	debug	hire
read	distinguish	construct	dissect	depict	interpret
recall	elaborate	customize	distinguish	design	judge
recite	estimate	demonstrate	document	develop	justify
recognize	example	depreciate	ensure	devise	measure

Figure 19: Verbs respective to each Bloom’s levels

As seen above, few verbs occur in more than one category. So, we learned that this critical information could be used in designing the weighting algorithm. First, we listed all possible verbs that occur in all 6 categories together. Then we created a dictionary which has these verbs as keys and the count of questions in which they occur as the values. Then we also created 6 separate dictionaries with the same keys and values mentioned earlier but the only difference is that the values (count of questions in which they occur) are category-specific. All of this data is used while categorizing the questions.

When a question is passed through the SVM classifier, We obtain a list of predicted categories by the SVM classifier. The weighting has to be done among these categories to obtain a single category with highest weightage. For this, a weighting algorithm is designed to assign weights. The assigned weights are nothing but the probabilities of words in the question occurring in the dataset (1100 questions) for each category in the predicted list.

```
for each question in question bank:
    list=SVM_classsification(question)
    weighted_list=weight_algo(question,list)
    find the max (weighted_list)
    classify that question to the max weight category
    write the question into the that particular category dataset
```

Figure 20: Hybrid Classifier algorithm

9.3.3 PREDICTION OF BLOOM’S LEVELS

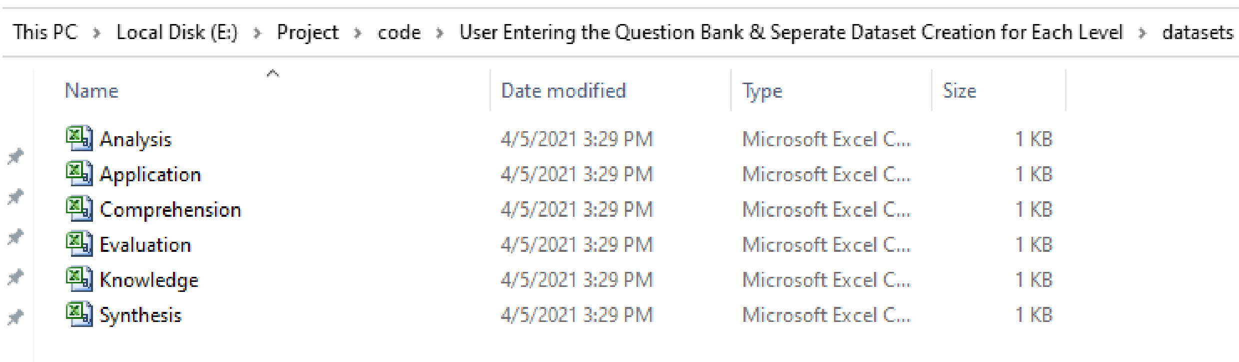
From the previous phase, we have a hybrid classifier that classifies the questions into one single category. The project has two main aspects. One is predicting the Bloom’s category for the questions in the question paper and another one is generating the question paper using the question bank given by the user. The purpose of the first aspect is to help understand the standard of the question paper.

For prediction, the user is prompted to input a question paper in the form of a csv file. This file is processed by passing each question to the hybrid classifier and the Bloom’s category for each question is predicted. The generation of question paper is explained in the next section.

9.3.4 QUESTION PAPER GENERATION

From the previous phase, we have questions classified into their respective Bloom’s categories (exactly one). Now, a question paper has to be generated from a given question bank. Also, the user-specified Bloom’s levels (in percentage) that are expected in the question paper must also be taken into consideration while choosing the questions from the question bank.

STEP 1: Once a question bank is selected for generating a question paper, it's passed to the hybrid classifier and the questions are grouped separately based on the predicted Bloom’s classes. The same is shown below.


This PC > Local Disk (E:) > Project > code > User Entering the Question Bank & Seperate Dataset Creation for Each Level > datasets

Name	Date modified	Type	Size
Analysis	4/5/2021 3:29 PM	Microsoft Excel C...	1 KB
Application	4/5/2021 3:29 PM	Microsoft Excel C...	1 KB
Comprehension	4/5/2021 3:29 PM	Microsoft Excel C...	1 KB
Evaluation	4/5/2021 3:29 PM	Microsoft Excel C...	1 KB
Knowledge	4/5/2021 3:29 PM	Microsoft Excel C...	1 KB
Synthesis	4/5/2021 3:29 PM	Microsoft Excel C...	1 KB

Figure 21: Dataset for each levels containing classified questions

STEP 2: Now that we have questions categorized, we consider the percentage of bloom’s levels expected by the user and pick the questions accordingly from each of these datasets and put them in the final Question paper. The total number of questions expected by the user is also considered while generating the question paper.

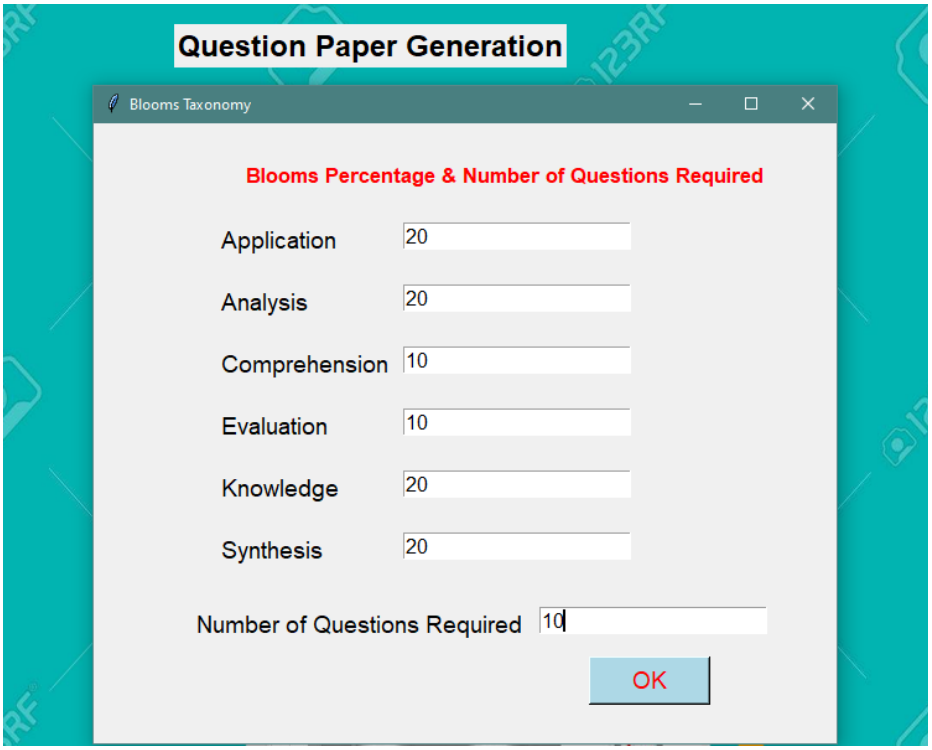


Figure 22: Inputting user specified percentages and number of questions

CHAPTER 10

EXPERIMENTATION RESULTS AND DISCUSSIONS

10.1 SINGLE QUESTION PREDICTION

In this project we saw predicting Bloom’s level of a given question is an important aspect , so once the classifier is built it has to be tested for prediction for a single question as well as a set of multiple questions. For a single question prediction a pdf file is taken as input.

Example is shown below.

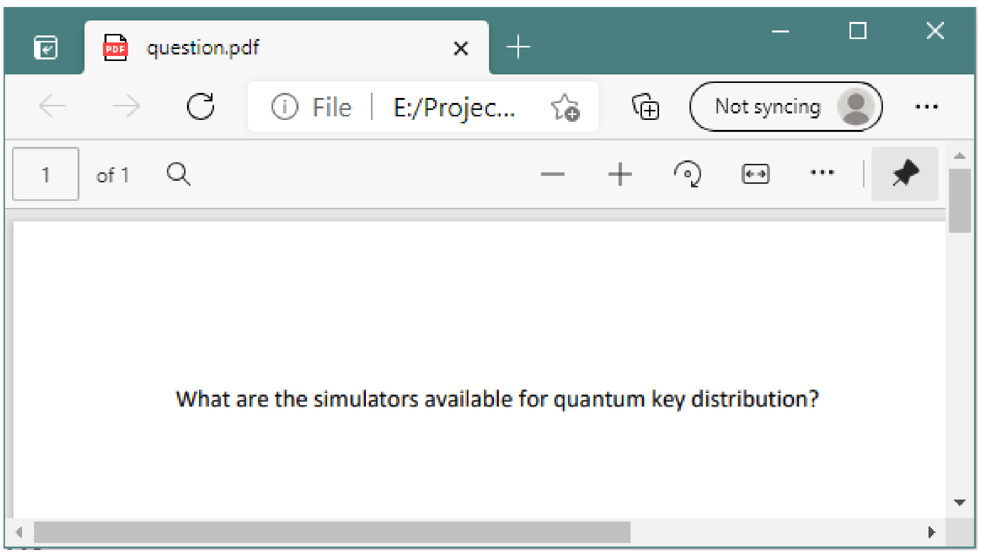


Figure 23: PDF file having single question for prediction

The question present in the pdf is extracted, the same is tokenized and passed to the hybrid classifier. The classifier predicts the class for the same.

```
>>>
 RESTART: E:\Project\code\User Entering the Question Bank & Seperate Dataset Creation for
Each Level\GUI-FINAL.py
[nltk_data] Error loading punkt: <urlopen error [Errno 11001]
[nltk_data]     getaddrinfo failed>


--------------------------Prediction_Single_Question--------------------------------
['what', 'are', 'the', 'simulators', 'available', 'for', 'quantum', 'key', 'distribution']
Knowledge
```

Figure 24: Prediction for one question

10.2 MULTIPLE QUESTION PREDICTION

For multiple questions prediction an excel (csv) file is taken as input. Each question present in the excel file is extracted, the same is tokenized and passed to the hybrid classifier. The classifier predicts the class of each of these questions. The same is shown in the below picture where the first line is the question, second line is the classes predicted by SVM and third line shows the hybrid classifier prediction.

```
--------------------------Prediction_Multiple_Questions--------------------------------

 Define the terms Unicasting, Multiccasting and Broadcasting
('Analysis', 'Synthesis', 'Knowledge', 'Comprehension')
Knowledge

 Describe in prose what is shown in graph form.
('Application', 'Knowledge', 'Comprehension')
Comprehension

 analyze the positive and negative points presented concerning the abolition of guns and write a brief (2-3page) narrative of your analysis.
('Knowledge', 'Synthesis', 'Analysis', 'Comprehension')
Analysis

 Compare and contrast our school to other communities.
('Synthesis', 'Analysis', 'Comprehension')
Analysis

 Construct the 4-Bit Ripple carry Adder Circuit
('Application', 'Knowledge', 'Comprehension')
Application

 Prepare a book jacket that illustrates the kind of book as well as the story.
('Synthesis', 'Knowledge', 'Comprehension')
Synthesis

 Evaluate the corrosion monitoring needs of a chemical processing plant
('Evaluation', 'Synthesis', 'Knowledge', 'Comprehension')
Evaluation
```


Figure 25: Predictions for multiple questions

10.3 QUESTION PAPER GENERATION

The next important aspect is to generate a question paper taking the question bank as input from the user. The classifier is trained with the training dataset and is stored in an intermediate file called model.joblib. Later a question bank is taken as input from the user example for the same is shown in the below figure.

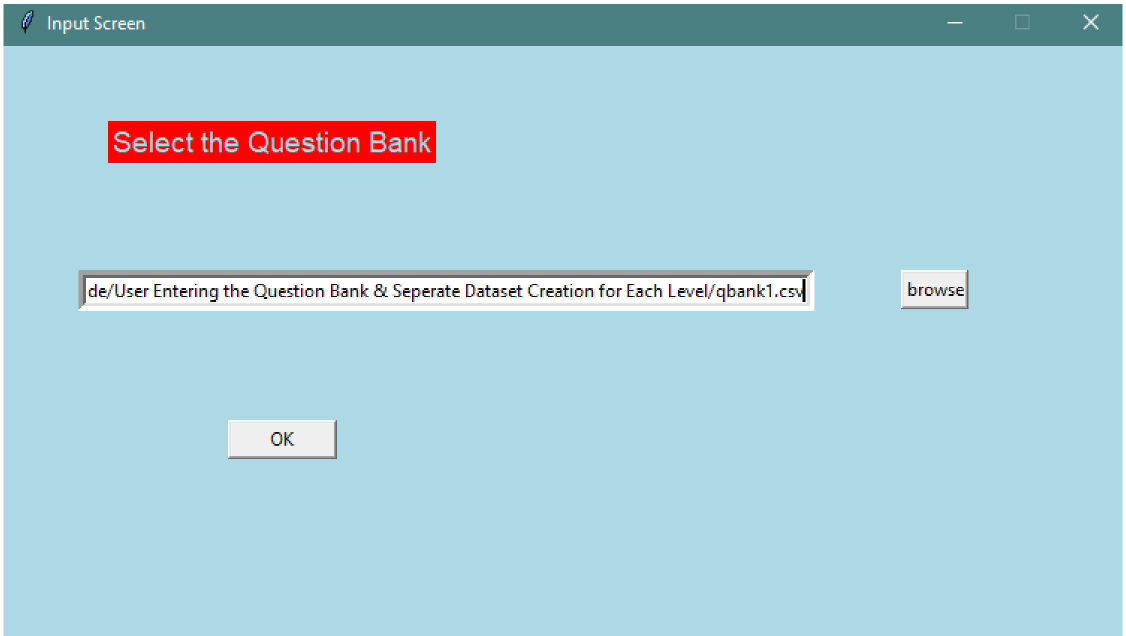


Figure 26: Inputting user question bank

Once the question bank is selected it is passed to the classifier for classification. Classes are predicted for the question using the classifier as shown in the previous section. To these predicted classes weighting algorithm is applied which further reduces the classification to one single class. First, we listed all possible verbs that occur in all 6 categories together. Then we created a dictionary which has these verbs as keys and the count of questions in which they occur as the values. Then we also created 6 separate dictionaries with the same keys and values mentioned earlier but the only difference is that the values(count of questions in which they occur) are category-specific. All of this data is used while categorising the questions.

When a question is passed through the SVM classifier, We obtain a list of predicted categories by the SVM classifier. The weighting has to be done among these categories to obtain a single category with highest weightage. For this, a weighting algorithm is designed to assign weights. The assigned weights are nothing but the probabilities of words in the question occurring in the dataset(1100 questions) for each category in the predicted list.These probabilities are calculated by using the frequencies of the words occurring in the data which are stored as dictionaries as mentioned above.

After classification, questions are stored in different csv files based on their predicted classes. Now that we have questions categorised, we consider the percentage of bloom's levels expected by the user and pick the questions accordingly from each of these datasets and put them in the final Question paper. The total number of questions expected by the user is also considered while generating the question paper. All these steps are shown in the below figure.

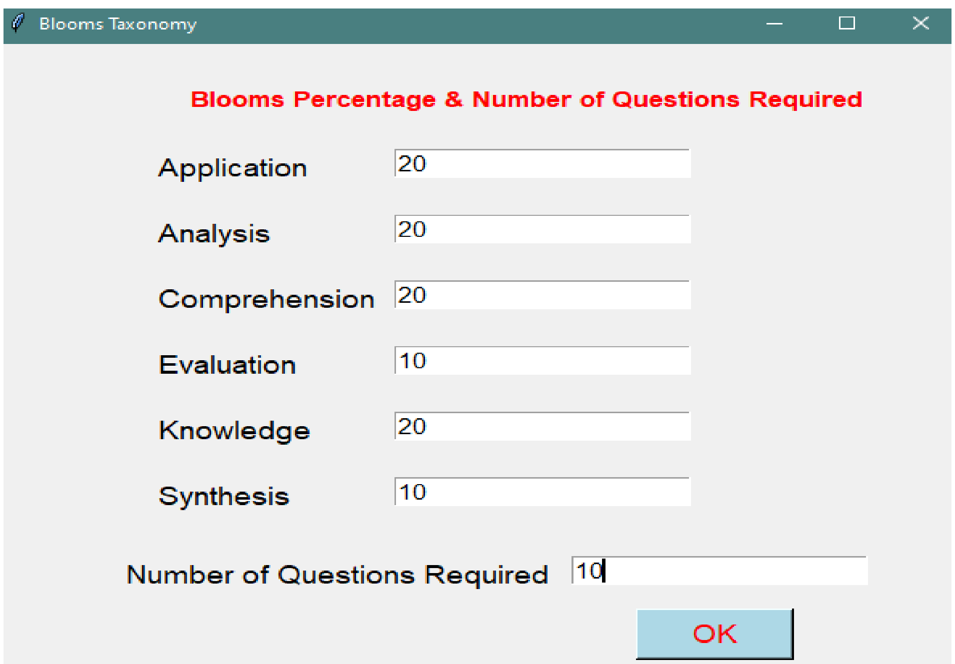


Figure 27: Inputting percentages and number of questions

Question Paper: ADA

Branch: COMPUTER SCIENCE

[Date: 30-03-20] [Marks:100]

Note: Answer all the following questions

---

1)given a snippet of code , decide which one of the processes gets executed first .",

2)how would you distinguish between spooling and buffering ?,

3)can you distinguish between mutex and semaphore ?,

4)describe micro kernel and macro kernel .,

5)evaluate the prerequisite for understanding convoy effect .,

6)do you agree if we say sstf is one of the better and widely known disk scheduling algorithm,

7)make a list of different operating systems and it 's examples .,

8)recite the main purpose of an operating system .,

9)invent different conditions to detect deadlocks .,

10)decide if all the algorithms that you have learnt in operating systems were helpful while applying to few of your daily problems .,

Figure 28: Question paper

10.4 PERFORMANCE

10.4.1 DATASET INFORMATION

The dataset consists of questions that belong to all six categories of Bloom’s Taxonomy. The same is shown in the below figure.

```
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\mamatha\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
Comprehension    201
Knowledge        191
Analysis         184
Application      179
Synthesis        170
Evaluation       164
Name: Category, dtype: int64
```

Figure 29: Data Information

10.4.2 TESTING WITH DIFFERENT ML TECHNIQUES

For classification of the questions into Bloom's levels we considered to train different ML techniques with our training dataset to see their performance . So the machine learning considerations are SVM(Support Vector Machine), Linear regression, MultinomialNB and Random forest classifier.After training and testing we obtained the below results.

ML ALGORITHM	ACCURACY
SVM ACCURACY	68.86446
Logistic ACCURACY	58.36441
MultinomialNB ACCURACY	56.36433
Random Forest ACCURACY	48.16674

Figure 30: Accuracy of different ML techniques

So as we see in the above image, the SVM classifier performance is better compared to other techniques so we used the SVM classifier which resulted with multiple categories being predicted for each question. In order to solve this ambiguity the primary classifier was blended with a weighting algorithm based model classifier which reduced the prediction to exactly one .The hybrid model which is SVM combined with weighting technique performs better than other ML algorithm which predict multiple classes.

CHAPTER 11

CONCLUSIONS AND FUTURE WORK

● The hybrid, as in SVM along with the weighting algorithm-based classifier performed better than other ML algorithm which predict multiple classes. This performs better and is also more dynamic than rule-based classifier that we tried to use during the first phase of the project.

● As for future work, collecting more keywords belonging to the six different learning levels as much as possible. Adding more rules to the second classifier can lead to the better model. A hybrid of SVM, weighting and rule based classifier is a good option to go with so that accuracy can be increased by including more rules and including more number of keywords.

REFERENCES/BIBLIOGRAPHY

[1] Nazlia Omar, Syahidah Sufi Haris, Rosilah Hassan, Haslina Arshad, Masura Rahmat, Noor Faridatul Ainun Zainal, and Rozli Zulkifli. "Automated Analysis of Exam Questions According to Bloom's Taxonomy" Procedia - Social and Behavioral Sciences, vol. 59, 2012. doi:10.1016/j.sbspro.2012.09.278

[2] Jayakodi, Kithsiri & Bandara, Madhushi & Perera, Indika & Meedeniya, Dulani. (2016). WordNet and Cosine Similarity based Classifier of Exam Questions using Bloom’s Taxonomy. International Journal of Emerging Technologies in Learning (iJET). 11. 142. 10.3991/ijet.v11i04.5654.

[3] Mohammed M, Omar N (2020) Question classification based on Bloom’s taxonomy cognitive domain using modified TF-IDF and word2vec. PLoS ONE 15(3): e0230442. https://doi.org/10.1371/journal.pone.0230442

[4] Das, Syaamantak & Mandal, Das & Basu, Anupam. (2020). Identification of Cognitive Learning Complexity of Assessment Questions Using Multi-class Text Classification. Contemporary Educational Technology. 12. ep275. 10.30935/cedtech/8341.

[5] Jain M., Beniwal R., Ghosh A., Grover T., Tyagi U. (2019) Classifying Question Papers with Bloom’s Taxonomy Using Machine Learning Techniques. In: Singh M., Gupta P., Tyagi V., Flusser J., Ören T., Kashyap R. (eds) Advances in Computing and Data Sciences. ICACDS 2019. Communications in Computer and Information Science, vol 1046. Springer, Singapore. https://doi.org/10.1007/978-981-13-9942-8_38

[6] Smith, Matthew & Bull, Larry. (2003). Feature Construction and Selection Using Genetic Programming and a Genetic Algorithm. 10.1007/3-540-36599-0_21.

[7] Bo Sun, Yunzong Zhu, Yongkang Xiao, Rong Xiao and Yungang Wei. Automatic Question Tagging with Deep Neural Networks. : DOI 10.1109/TLT.2018.2808187, IEEE .vol 12 issue 1

[8] Mandeep Choudhary , V. S. Dhaka Jaipur National University, Jaipur, India. Automatic emails Classification Using genetic Algorithms. Mandeep Choudhary et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 6 (6) , 2015, 5097-5103

[9] Verdú, Elena & Verdú, María & Regueras, Luisa & De Castro, Juan Pablo & García, Ricardo. (2012). A genetic fuzzy expert system for automatic question classification in a competitive learning environment. Expert Systems with Applications. 39. 7471–7478. 10.1016/j.eswa.2012.01.115.

[10] Ahmad, Munir & Aftab, Shabib & Ali, Iftikhar. (2017). Sentiment Analysis of Tweets using SVM. International Journal of Computer Applications. 177. 975-8887. 10.5120/ijca2017915758.

11. https://cft.vanderbilt.edu/guides-sub-pages/blooms-taxonomy/

APPENDIX A DEFINITIONS, ACRONYMS AND ABBREVIATIONS

Bloom's taxonomy: Bloom's Taxonomy is a classification of the different educational learning levels with objectives and skills that instructors/tutors/examiners set for their students. It has six levels which are Knowledge (remembering), Comprehension (understanding), Application (applying), Analysis, Synthesis (creating), and Evaluation.

NLP: Natural Language Processing
BERT: Bidirectional Encoder Representations from Transformers
LDA: Latent Dirichlet Allocation

Normalization: Normalization is a procedure that transforms a list of words to a more consistent or uniform sequence

Tokenization: Tokenization is the process of splitting given larger character sequence and a defined document unit up into pieces, called tokens.

Stemming: Stemming is the task of lessening a word to its word stem that affixes or suffixes and prefixes or to the roots of words called lemma.