An automatic classifier for exam questions in Engineering: A process for Bloom's taxonomy

Conference Paper · December 2015

DOI: 10.1109/TALE.2015.7386043

Kithsiri Jayakodi
Wayamba University of Sri lanka

Madhushi Bandara
University of Technology Sydney

Indika Perera
University of Moratuwa

An Automatic Classifier for Exam Questions in Engineering: A Process for Bloom's Taxonomy

Kithsiri Jayakodi
Department of Computing and Information Systems
University of Wayamba
Sri Lanka
itkith@yahoo.com

Madhushi Bandara, Indika Perera
Department of Computer Science and Engineering
University of Moratuwa
Sri Lanka
madhushi@cse.mrt.ac.lk, indika@cse.mrt.ac.lk

***Abstract*—Assessment is an essential activity to achieve the objective of the course being taught and to improve the teaching and learning process. There are several educational taxonomies that can be used to assess the efficacy of assessment in engineering learning by aligning the assessment tasks in line with the intended learning outcomes and teaching and learning activities. This research is focused on using a learning taxonomy that fits well for computer science and engineering to categorize and assign weights to exam questions according to the taxonomy levels. Existing Natural Language Processing (NLP) techniques, Wordnet similarity algorithms with NLTK and Wordnet package were used and a new set of rules were developed to identify the category and the weight for each exam question according to Bloom's taxonomy. Using the result the evaluators can analyze and design the question papers to measure the student knowledge from various aspects and levels. Prior evaluation was conducted to identify most suitable NLP preprocessing techniques to the context. A sample set of end semester examination questions of the Department of Computer science and Engineering (CSE), University of Moratuwa was used to evaluate the accuracy of the question classification; weight assignment and the main category assignment were validated against the manual classification by a domain expert. The outcome of classification is a set of weights assigned under each taxonomy category, indicating the likelihood of a question to fall into a certain category. The highest weight category was considered as the main category of the exam question. According to the generated rule set the accuracy of detecting the correct main category of a question is 82%.**

Keywords*— Question classification; Assessment in Engineering; Teaching and Supporting Learning; Bloom's taxonomy; Learning Analytics; Natural Language Processing

I. Introduction

A question is an element that is used to evaluate the achievement of learning. Effective style of question is always important when assessing the learning outcome of the course module and degree program. Educational taxonomies such as Bloom's [1] and SOLO [2] can help to identify the question level and categorize accordingly. Questions included in the end semester exam papers play a vital role in testing the student's overall cognitive levels. When questions are prepared, there should be an effective balance between questions which assess the high level of learning and questions which assess the basic level of learning [3]. Often examiners may be unaware of the level of difficulty of the questions that are included in the final exam. Poorly designed assessments can fail the goals of examination and lead to unsatisfactory achievement of learning outcomes, which can ultimately be catastrophic to the educational institute.

In engineering courses an exam question often falls into a combination of multiple levels of assessment to cater both higher and lower levels of student learning capabilities, which can be challenging to identify. Therefore this research was carried out to develop a reliable rule set to categorize the questions according to an educational taxonomy and assign weights for each category of the question using NLP; best NLP preprocessing techniques and the best semantic similarity algorithms were identified as part thereof.

II. Literature Review

A. Educational Taxonomy

Educational taxonomy can be used to provide a shared language for describing learning outcomes and performance in assessments. Educational objectives can be divided into three domains such as cognitive, affective and psychomotor. Among the popular taxonomies Bloom's taxonomy considers each of these as a dimension in the taxonomy [1] and the revised Bloom's taxonomy describes these as a matrix [4]. Apart from that SOLO taxonomy describes a mixture of quantitative and qualitative differences between the performances of the students [5]. Furthermore, learning taxonomies can be used to define the curriculum objectives of a course [6], which can be used to describe the learning stages that a learner is operating for a certain topic. For example, in computer science, a student may be capable of explaining what is meant by the concept polymorphism yet may not be able to implement that using a programming language. Lecturer can assess a student at a chosen level through a suitable choice of questions or examples [3]. Taxonomies can also be used to give an insight on the level of understanding about the subject and their studying techniques [6].

Among many learning taxonomies available today Bloom's taxonomy is widely used [3]. There are six major categories of cognitive processes in Bloom's taxonomy as shown in Fig. 1. Bloom's taxonomy was revised by Anderson et al [4]. They changed the nouns listed in Bloom's into verbs (Table I). The categories can be thought of as degrees of difficulties. The taxonomies define the level of performance that might be expected for any given content element. These taxonomies can be considered as sequential learning processes. Bloom's revised taxonomy identified a matrix with knowledge dimensions of factual, conceptual, procedural and metacognitive levels [8]. SOLO taxonomy [2] does not refer to cognitive characteristics of the learner's performance or to the affective dimensions. It focuses on the content of the learner's response to what is being assessed. It encourages students to think about the current position with their learning, and what they should do in order to progress. There are five main stages to be followed sequentially: Pre-structural, Uni-structural, Multi-structural, Relational, and Extended Abstract. Compared to Bloom's, SOLO is more abstract in defining learning progression; Bloom's keywords provide direct application of the model to question formation, however.

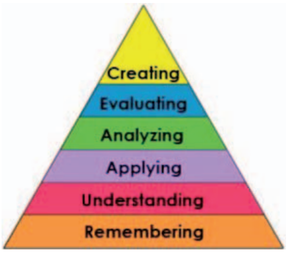


Fig. 1. Bloom's Taxonomy [1]

TABLE I. ANDERSON'S REVISED BLOOM'S TAXONOMY

Category	Cognitive Verb list of Anderson Taxonomy	
	Description	***Verb list***
Categories	Recall or retrieve previous learned information.	defines, describes, identifies, knows, labels, lists, matches, names, outlines, recalls, recognizes, reproduces, selects, states
Understand	Comprehending the meaning,and interpretation of instructions and problems.	comprehends, converts, defends, distinguishes, estimates, explains, extends, generalizes, gives an example, infers, interprets, paraphrases
Apply	Use a concept in a new situation or unprompted use of an abstraction	applies, changes, computes, constructs, demonstrates, discovers, manipulates, modifies, operates, predicts
Analyze	Separates material or concepts into component parts	analyzes, breaks down, compares, contrasts, diagrams, deconstructs, differentiates, discriminates, distinguishes, identifies, illustrates
Evaluate	Make judgments about the value of ideas or materials	appraises, compares, concludes, contrasts, criticizes, critiques, defends, describes, discriminates, evaluates, explains, interprets
Create	Builds a structure or pattern from diverse elements	categorizes, combines, compiles, composes, creates, devises, designs, explains, generates, modifies, organizes, plans, rearranges, reconstructs

B. Educational taxonomy for exam evaluation

Bloom's taxonomy has been widely researched for increasing teaching efficiency. Disregarding the Bloom's taxonomy was found out as the leading source of engineering laboratory course failures in universities [9]; lack of lower level knowledge components inhibits application of medium level elements required for laboratory exercises. Use of Bloom's taxonomy was examined for Electrical Engineering courses for deep and surface learning [10]. In computer science courses certain disagreements were reported between the academics that design and deliver the course against the ones who review the assessment tasks about the level at which the assessment was carried out [11]. This is partly due to the domain nature of the modules. A reason for this could be the difficulty of determining the taxonomic level of the assessment without having an intimate knowledge of the way in which the material being assessed was taught. Revised Bloom's taxonomy, in general, is found to be appropriate to review the exam questions in terms of how the subject was taught. Reading the questions does not always give a clear indication of the cognitive skill involved in answering the question. For example, in a large program there may be parts that require *Apply* (i.e. apply a design pattern) but the whole question could come under *Create* [11]. Therefore this research aimed at giving an appropriate solution to assign the weights for the categories of each question using revised Bloom's taxonomy.

C. Natural Language Processing Used in Education

Challenges in NLP involve natural language understanding, that is, enabling computers to derive meaning from human or natural language input and generation. Natural languages are challenging to be categorized as grammatical or ungrammatical due to various factors [13]. Due to the complexity of the languages and the human nature in processing and interpreting them the field of NLP has moved towards statistical NLP. Therefore statistical NLP has a solid theoretical foundation and is much responsible for recent development in the field of NLP. Standard text corpora and lexical tools such as Wordnet are useful tools in statistical NLP. Statistical NLP techniques can be used for question classification problems. Pavel [12] has discussed the usage of NLP techniques in teaching and learning process. In his research different areas such as Learning Management System support for users, question answering and assessment generation, language learning and course preparation were identified for NLP application.

Chang [14] has presented an online testing system that can be used to classify cognitive levels of exam questions based on Bloom's taxonomy. The system segments questions and stores the verbs, which are related to Bloom's taxonomy. Then the system compares the verb tenses and look for keywords found in the test item. Weights are assigned for matching keywords in a particular question with respect to other questions. There are four match situations to indicate matching items; Correct Match Items, Partial Match Items, No Keyword Items and No Match Items [15]. Some researchers even tried Neural Network with different featured method trained with the gradient learning algorithm [16]. Auto marking is a system developed to mark the answers of a question based on semantic techniques of NLP. The student answer is evaluated with the semantic meaning of the sentences [14]. Question categorization with keyword mapping is not the appropriate solution every time. If the semantic meaning do not map with the question, it is difficult to map the question according to a given educational taxonomy [14]. In NLP educational research steps include stemming, lemmatization and part-of-speech tagging [17].

NLP gives several advantages such as high expressiveness, flexibility, representation of reality, require no training to use, freedom of expression and no indexing necessary. The disadvantages however are such as difficulty in making generic searches, problems with synonyms, homographs, and false drops, and non-standardization etc. [18]. With the development of Wordnet package most of the above mentioned problems were addressed [19].

1) Wordnet: This is a lexical database for English [19] use to solve the language ambiguity. It groups words into sets of synonyms. Wordnet includes the lexical categories of nouns [20], verbs [21], adjectives [22] and adverbs but ignores prepositions, determiners and other function words. Wordnet has been used for word sense disambiguation, information retrieval [23], automatic text classification, automatic text summarization, and machine translation [24] to name a few.

Wordnet based similarity detection between words can be used to identify the semantic of an exam question and to categorize and assign weight for each category of the question according to a given learning taxonomy. Various algorithms have been proposed, and these include measuring the distance among the words and *synsets* in Wordnet's graph structure, such as by counting the number of edges among synsets [21]. The intuition is that closer the two words or synsets are, the closer their meaning. Many Wordnet-based word similarity algorithms are implemented in Python package NLTK [25].

2) Wordnet based algorithms for semantic similarity: Wordnet consists of number of algorithms, which is used to measure the semantic similarity and relatedness between pair of concepts (synsets) [26]. Path similarity is an algorithm which was implemented in Wordnet that returns a score denoting how similar two word senses are, based on the shortest path that connects the senses in the is-a (hypernym/hypnoym) taxonomy [27].The score is in the range 0 to 1. Leacock-Chodorow Similarity is another algorithm [28]. It returns a score denoting how similar two word senses are, based on the shortest path that connects the senses and the maximum depth of the taxonomy in which the senses occur. The relationship is given as -log(p/2d) where p is the shortest path length and d the taxonomy depth. Wu-Palmer Similarity[29] returns a score denoting how similar two word senses are, based on the depth of the two senses in the taxonomy and that of their Least Common Subsumer (most specific ancestor node). The Jiang-Conrath Similarity[30] returns a score denoting how similar the two word senses are, based on the Information Content (IC) of the Least Common Subsumer and that of the two input synsets. The relationship is given in (1).

$$1 / (IC(s1) + IC(s2) - 2 * IC(lcs)) \quad (1)$$

Lin Similarity[31] returns a score denoting how similar the two word senses are, based on the Information Content (IC) of the Least Common Subsumer and that of the two input Synsets. The relationship is given in (2).

$$2 * IC(lcs) / (IC(s1) + IC(s2)) \quad (2)$$

3) Classification Techniques for Question Categorization: Question classification is a special branch of text classification [31] and many studies were conducted on improving the accuracy of question classification using machine learning techniques such as support vector machines (SVM) [32], rule based classification [33], and head word usage [34]. An artificial neural network is proposed in [35], in which back-propagation neural network is used as text classifier classifying question into three difficult levels: easy, medium, and hard. SVM is successfully used in the classification of open-ended questions [36].

III. METHODOLOGY

As discussed previously, the main challenge is that exam questions are not categorized and weights are not assigned for each category properly in final exam papers in engineering courses. To address this challenge this research followed its methodology as shown in Fig. 2 with revised Bloom's Taxonomy. NLP techniques and rules were used to categorize the questions and assign weights for each. Tokenization, stemming, lemmatization and tagging were carried out before generating the ruleset to categorize and weighting of the questions. There are seven stages used to categorize the questions automatically as described below.

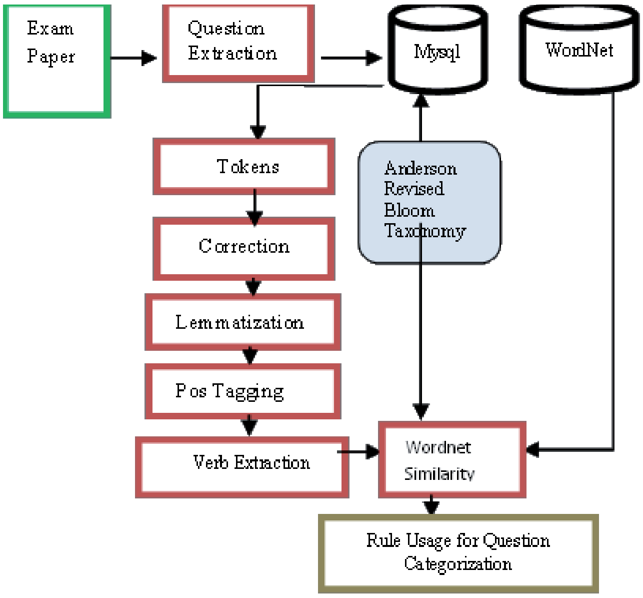


Fig. 2. The proposed system architecture for question analysis

A. Steps of the Question processing with NLP techniques

1) Question Extraction: Pypdf package was used to extract exam questions in PDF. Then questions were identified with a set of regular expressions describing the character patterns we are interested in. For example, following regular expression was used to identify the main description of the question.

r"\[Q" + str(QMain) + r"\](.*?)\[Q" + str(QMain+1) + r"\]"

QMain was given the question number as (Q1) and **QMain +1** is (Q2). The expression **(.*?** derives all the text in between the questions.

2) Tokenization: Once questions were stored in MySQL database, each question is then tokenized into three sentences. These sentences then are split into individual words. Word tokenizer, wordpuncttokenizer, treebanktokenizer, standfordtokenizer and regexptokenizer have been used to identify the most appropriate tokenizer [37]. When tokenizing, concern has given to break the sentences into words. If a tokenizer break a sentence due to other factors such as full stop, brackets explanation marks and other non-alphanumeric characters then the stemmers and taggers has to work with unwanted portion of the tokens to tagging, stemming and lemmatization. Regexptokenizer tokenizer has developed to token based on spaces of the sentences. Therefore it producers a less number of appropriate tokens to proceed with the subsequent steps of the process (Fig. 3).

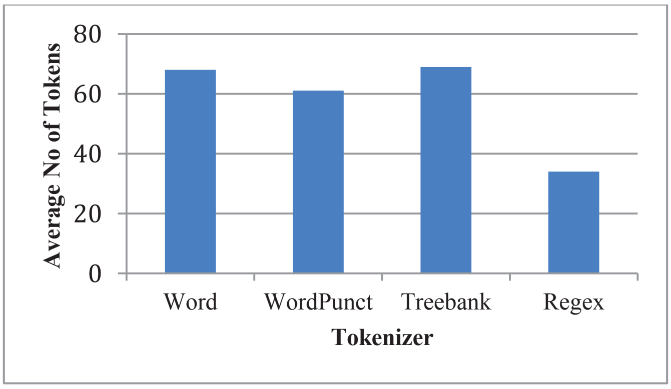


Fig. 3. Average number of tokens generated with tokenizers

3) Question Correction: Enchant package (spelling correction API) was used for question correction. A python module was developed with Enchant to correct the word after tokenization. This module starts by creating a reference to an enchant dictionary. If the word is not found in the dictionary, it looks up a list of suggestions and returns the best suggestion or allows the user to select the word through the python GUI.

4) Stemming and Lemmatization: Stemming is a technique that is used to remove the affixes from a word. Lancester, porter [17], snowball and regular expression stemmers were tested to examine the validity of the usage according to our research. Lemmatization is more appropriate according to the research. Unlike stemming here we are always left with a valid word with the same meaning as in the original sentence. Lemmatization words just do not chop. For example WordnetLemmatizer in Wordnet always tries to find a matching valid root word. Therefore it can be appropriate to lemmatize the word before tagging to find the verbs rather than applying stemming techniques.

5) Tagging: Part-of-speech tagging is the process of converting a sentence, in the form of a list of words, into a list of tuples, where each tuple is of the form (word, tag). Many taggers were tested for appropriateness. A unigram tagger only uses a single word as its context for determining the part-of-speech tag. After the training process is over unigram tagger was tested with the tree bank corpus for the evaluation purposes. It has given 0.85888 as the accuracy value. Then Brill tagger was evaluated with another initial tagger and training sentences. Brill tagger was used with backoff taggers such as unigram tagger, trigram tagger and with a default tagger. Accuracy was reported as up to 0.8829. Then in the next phase TnT tagger, a statistical tagger based on second order Markov models, was used. TnT tagger was trained with the treebank corpus and once it was tested the evaluation accuracy was given as 0.8753. Then TnT tagger was tested with the usage of default tagger (with NN for all words); with tree bank corpus, accuracy was given as 0.8924. Then experiment was done with TnT tagger by further increasing the number of solutions the tagger maintained. It was increased to 2000, (usually it is 1000); the accuracy was given as 0.8765. WordnetTagger was used with the back-off taggers of unigram, bigram and trigram and with the default tagger (NN); accuracy was increased to 0.8848. Classifier based tagger was used with the tree bank corpus sentences, which resulted in accuracy of 0.9309. Since the classifier based tagger with the usage of ClassifierBasedPOSTagger has given the highest accuracy it was selected for the tagging. Table II summarises the tagger accuracies.

TABLE II. ACCURACY OF NLP TAGGERS TESTED WITH TREE BANK CORPUS

Tagger	Accuracy
[a] unigram tagger	**0.7757**
[b]unigram tagger with Default tagger(NN)	**0.8588**
[c]Brill tagger	**0.8829**
[d]Tnt tagger	**0.8756**
[e]Tnt tagger with Default tagger(NN)	**0.8924**
[f]Tnt tagger with N=2000	**0.8765**
[g]Wordnet tagger with backofftaggers	**0.8848**
[h]Classifier based tagger	**0.9309**

6) Verb extraxtion: After the completion of tokenization, word correction, Lemmatization and tagging, verbs were extracted for each question and stored in MySQL database. The tag starting is matched with letters ‘V’, and ‘W’, selected as related words for Anderson taxonomy and stored in the database. Therefore VB, VBD, VBG, VBN, VBP, VBZ, WDT, WP, WP$ and WRP types of tag words were extracted.
What, when, where, which, why, who, how words are considered as Bloom’s taxonomy knowledge category words. Since majority of the words that was extracted falls under verb types all the words that was extracted in this research was named as verbs.

7) Verb net similarity comparison: NLTK comes with a simple interface for looking up words in WordNet. Synset instances are groupings of synonymous words that express the same concept. Many words have only one synset, but some have several synsets. Algorithm in Fig. 4 was used with many semantic similarity algorithms to identify the most suitable algorithm. According to the step 3 all the verbs of a question were extracted. In step 4 taxonomy word list was taken from MySQL database for each category. After that in steps 7 and 8 wordnet synsets lists were taken for each word of taxonomy category and for each word. Then in step 12 wordnet similarity algorithm was used to derive the similarity features. If a similarity feature was given, then it was appended in to an initial list. Once the verblist and the taxonomy word list for one category was completed the maximum value of similarity and the average similarity were calculated as part of the feature set used to build the rules. Fig. 4 algorithm was extended in Fig. 5 to identify the lemma similarity.

```
For question in QuestionList
    Wordlist= Tagger(Lemmatizer(Spellcorrector(Tokenizer(question))))
    Verblist = ExtractVerb(Wordlist)
    for taxonomywordlist1 in Taxonomywordcategory
    for    taxonomyword in Taxonomywordlist1
        for verb in Verblist
            Synsettaxonomylist =  wordnet.Synsets(taxonomyword))
            Synsetverblist          =  wordnet.Synsets(verb))
            if len(Synsettaxonomylist)>0 and len(Synsetverblist)>0:
                for listA in Synsettaxonomylist:
                    for listB in Synsetverblist:
                        s = listA.WordnetSemanticSimilarity(listB)
                            if str(s) == 'None':
                                s = 0
                            initialList.append(s)
                            if len(initialList)>0:
    generate sum(initiallist),Max(initiallist),
                  average = sum(initialList)/len(initialList)
```

Fig. 4. Algorithm for verb extraction and to find the maximum, average synset similarity value for each word in Bloom's taxonomy

The algorithm in Fig. 5 is used to find out number of lemmas for each verb in the question and the number of lemmas of each category of the taxonomy; it is used to test the equality of the lemmas. Finally all the similarity lemmas are added for each category and the database was updated. Path similarity, Leacock-Chodorow-Similarity, Wu-Palmer-Similarity and Leacock-Chodorow-Similarity, Jiang-Conrath Similarity, and Lin Similarity were used to identify the appropriate similarity algorithm. Information Content (IC) file is used with the similarity algorithms.

```
For question in QuestionList
    Wordlist=Tagger(Lemmatizer(Spellcorrector(Tokenizer(question))
    Verblist = ExtractVerb(Wordlist)
        for taxonomywordlist1 in Taxonomywordcategory
            for    taxonomyword in Taxonomywordlist1
            for verb in Verblist
                Synsettaxonomylist = wordnet.Synsets(taxonomyword)
                Synsetverblist          =  wordnet.Synsets(verb))
                    if len(Synsettaxonomylist)>0 and len(Synsetverblist)>0:
                    for listA in Synsettaxonomylist
                     for lemma in syn.lemmas():
                    for listB in Synsetverblist:
                        for lemma1 in syn1.lemmas():
                        if(lemma.name()==lemma1.name()):
                        counter = counter +1
```

Fig. 5. Algorithm for verb extraction and to find the number of lemma similarity value for each Bloom's taxonomy category

According to Table III jcn_similarity (brown), jcn_similarity (semcor), res_similarity (semcor) and res_similarity (brown) algorithms and lch_similarity have given high range value distribution whereas lin_similarity (brown), lin_similarity (semcor), wup_simlarity, path_similarity have given value distribution within 0 to 1 range except for lch_similarity. Therefore for the first four values log scale was used to compare the usefulness to identify the verb similarity for the given application. Sum value of semantic similarity algorithms for every verb of the question with all the verbs of Bloom's categories were used to identify the main category of a question and it was validated by a domain expert. The path similarity and the Wu-Palmer-Similarity algorithms gave the best accuracy in our research as shown in Fig 6. 26 questions were tested with each category in order to identify the best Wordnet similarity algorithm. Out of 26 questions 22 questions were accurately identified with the path similarity algorithm. Fig. 7 shows an executing instance of the implemented system.

TABLE III. MAXIMUM WORDNET SIMILARITY OF BLOOM'S (ANDERSON REVISED) TAXONOMY FOR IDENTIFIED VERBS

Algorithm	IC	Wordnet similarity value Range	
		Mimimum value	***Maximum Value***
jcn_similarity	brown	1e-300	1e300
jcn _similarity	semcor	5e-301	1e300
res _similarity	semcor	0	1e300
res _similarity	brown	0	9.9732
lin_ similarity	brown	0	1.0
lin _similarity	semcor	0	1.0
lch_similarity		0.485508	3.258097
path_similarity		0	1.0
wup_similarity		0	1.0

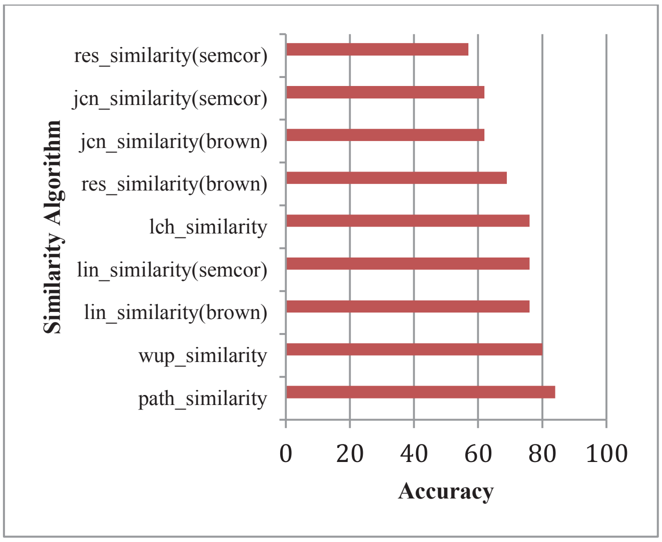
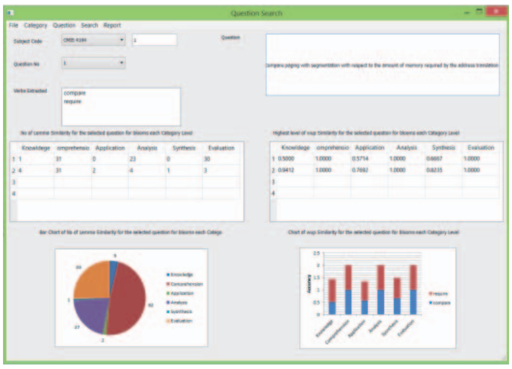


Fig. 6. Wordnet semantic similarity algorithm comparison with human anotated computer science questions.



Subject Code CMB 4184 1

Question No 1

Verbs Extracted
compare
require

Question
compare paging with segmentation with respect to the amount of memory required by the address translation

No of Lemma Similarity for the selected question for blooms each Category Level

	Knowledge	omprehensio	Application	Analysis	Synthesis	Evaluation
1	1	31	0	23	0	30
2	4	31	2	4	1	3
3						
4						

Highest level of wup Similarity for the selected question for blooms each Category Level

	Knowledge	omprehensio	Application	Analysis	Synthesis	Evaluation
1	0.5000	1.0000	0.5714	1.0000	0.6667	1.0000
2	0.9412	1.0000	0.7692	1.0000	0.8235	1.0000
3						
4						

Bar Chart of No of Lemma Similarity for the selected question for blooms each Catego
5
33
1
62
27
2
Knowledge
Comprehension
Application
Analysis
Synthesis
Evaluation
Chart of wup Similarity for the selected question for blooms each Category Level
2.5
2
1.5
1
0.5
0
Accuracy
require
compare
Knowledge
Comprehension
Application
Analysis
Synthesis
Evaluation

Fig. 7. An excution instance of the implemented applicaiton.

IV. Result and Dataset Analysis

The test questions are a collection of examination questions of MSc in Computer Science courses, which were obtained from the Faculty of Engineering, University of Moratuwa. The training set consists of 85 examination questions and the test dataset comprises of 62 questions. According to the output most of the questions were solely belong to one category; therefore weights were assigned for each category based on the path similarity and lemma similarity values. Mere reliance on a keyword found in the question often does not necessarily mean that an accurate category or cognitive level can be determined automatically. Based on the questions that were categorized manually a question may fall into more than one category. Weightage for each question was done automatically according to the following analysis. An execution instance of the developed application is shown in Fig. 7. Following analysis shows three different questions that were analyzed.

A. *Q1: What is meant by BI governance? what are its main components? what is the most important role played by BI solution?*

This question belongs to a lower level of Bloom's taxonomy; a level where students remember or memorize facts or recall the knowledge they have learnt before. As the results of verb extraction process, 'what' and 'play' words were extracted. Even though the word 'play' does not give a big impact to the question categorization it has given values above 0.87 in each category level. But the word 'what' appears under the first Bloom's category, max similarity was given as 1. Therefore the total was well above the other Bloom's category levels (Fig. 8). Hence, the gap between the first level and the other level were comparatively high. If the category value is very high compared to other categories, those do not get weights. Moreover 'what' was found three times in the processing and was used to clarify further the category of similar type of questions to assign weights and generate rules.

B. *Q2: Build a case as to why an agile approach is appropiate for the development of a BI solution?*

The above question was categorized, semi-automatically, as Application and Synthesis level of the Bloom's taxonomy. The words 'build' and 'why' were identified after the verb extraction process, which are also found in the taxonomy. Levels L1 to L6 were identified with Bloom's taxonomy categories such as knowledge, comprehension, etc.

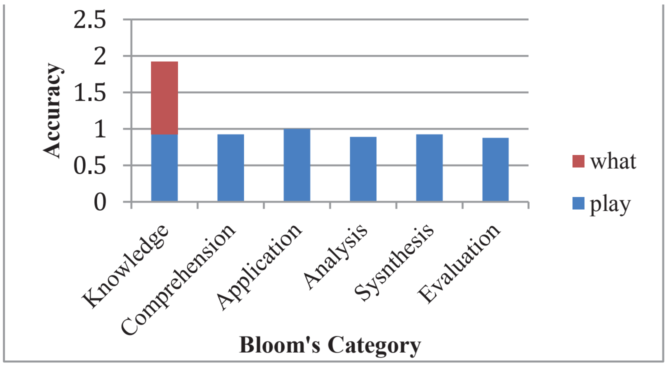


Fig. 8. Maximum synset wordnet similarity of Bloom's (Anderson Revised) Taxonomy for identified verbs in Q 1

TABLE IV. Maximum wordnet similarity of Bloom's (Anderson Revised) Taxonomy for identified verbs in Q 2

Verbs	Bloom category level					
	L 1	L2	L 3	L 4	L 5	L 6
	1.7692	1.3334	2	2	2	1.7273
build	0.7692	0.6667	1	1	1	0.7273
why	1	0.6667	1	1	1	1

According to the path similarity values given in Table IV question 2 was categorized as a question of *Application*, *Analysis* and *Synthesis* levels. Since the total of the two words in those two levels equals to 2 according to Table IV, we will not be able to categorize above question exactly into all three categories. Therefore in cases like this lemma similarity was observed further to get an idea about the question to group it and assign weight for each category. According to the lemma similarity, Application was identified with 202 lemmas [45%], Synthesis was identified with 205 lemmas [46%], and Analysis was with 23 lemmas [5%] (Fig. 9). As per the number of lemmas found the question can be identified as one belongs to Application and Synthesis categories. Weights were assigned to each category based on the number of lemmas found. In this case it was equally divided among *application* and *synthesis*.

C. *Q3: Compare paging with segmentation with respect to the amount of memory required by the address translation?*

This was categorized as a Comprehension level question. Bloom [1] describes this level as grasping the meaning of information. Even though this question can be categorized as Comprehension type, after the path similarity algorithm values with Wordnet the highest values were given for the levels Comprehension, Analysis and Evaluation categories, which is equivalent to 2.0 as in Fig.10. Therefore when assigning weights according to path similarity, 0.33 was given to each category. To analyse the questions further lemma accuracy was checked. According to Fig. 11, 62 was returned as the highest number in Comprehension, 33 in Evaluation and 27 in Analysis. Hence the weight assigned for this type of question is 0.50 for comprehension 0.25 for analysis and 0.25 for evaluation.

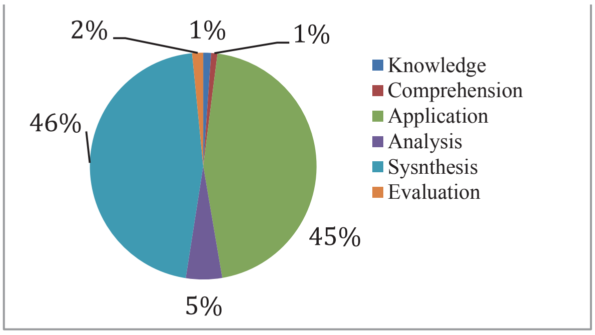


Fig. 9. Number of lemmas in synset wordnet similarity of Bloom's (Anderson Revised) Taxonomy for identified verbs in Q 2

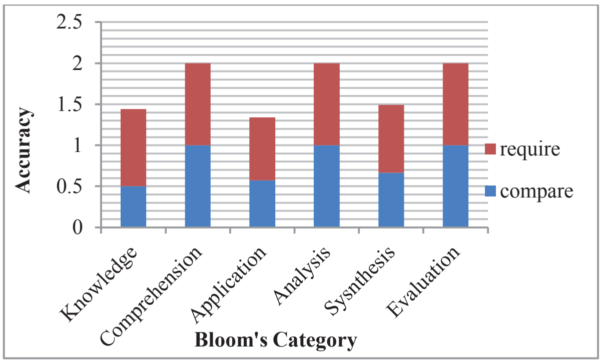


Fig. 10. Maximum synset wordnet similarity of Bloom's (Anderson Revised) Taxonomy for identified verbs in Q 3

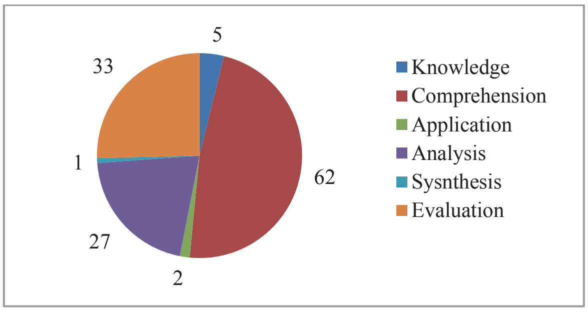


Fig. 11. Number of lemmas in synset wordnet similarity of Bloom's (Anderson Revised) Taxonomy for identified verbs in Q 3

V. Rule Generation and Weight assignment

As we mentioned previously the highest path similarity value and the lemma similarity values have been used to generate and identify the rules. We have used the following assumption in order to generate the rules. Usually the proportion that each question belongs to a category can be different. The values of the highest path similarity algorithm and equal number of lemma values have been normalized by scaling between 0 and 1. Once it is normalized if the gap between the highest and the lowest value is less than 0.25 then those categories were omitted in the first rule and categorized that question under the highest value category. Then full weight was assigned to that category. In this instance considering lemma similarities is not needed. If there are more than one category, which gives the same highest value then we moved into lemma similarity normalized values to generate the rule. Based on that several rules were generated to identify the main question category and assign the weights of the question. In such scenarios this tool support with the rule set provide an important insight to the question being analysed as manual categorization and weighting can be subjective and often fail to include all the possible categories.

VI. Disscussion

Based on above analysis rules were identified to assign the weight for category of the question and assign the question to the top most category as well. Since there are no accepted weights for each category of questions by academics the automatic weight assigning was given partially accurate results. According to the generated rule set, 51 questions out of 62 were identified with correct category (accuracy is 82%). Average synset similarity was also used to identify question categorization but it does not give reliable results to differentiate the questions into different categories since it gave similar results for all questions (Table V). Since human experts were able to identify only the main category of the question, our weight assignment for each category can be used to effectively build and assess the quality of the teaching, learning and assessment in engineering course modules.

TABLE V. Average Wordnet Similarity of Bloom's (Anderson Revised) Taxonomy for Questons

	Bloom category level					
	L 1	L2	L 3	L 4	L 5	L 6
Q1	0.1970	0.1927	0.2012	0.2100	0.203	0.2102
Q2	0.1964	0.1893	0.2035	0.2026	0.206	0.2043
Q3	0.1882	0.1817	0.1933	0.2038	0.191	0.2009

A. Study Limitations

Some end semester exam questions were having a description before the question. That was not evaluated to categorize the question and assign the weights. Another limitation is when there are images as part of the question description the information given in those images are not considered for the process. Apart from that certain question verbs were extracted and given high lemma similarity values, which are not suitable to classify the question according to the taxonomy. Such as in question 1, 'play' was identified as a verb in the question, although not relevant given a high accuracy for a number of similar lemmas. Moreover, some questions are not used with verbs and are difficult to categorize under this process.

VII. Conclusion

We have presented the need for rule-based automatic question classification in-line with one of the widely used teaching and supporting learning taxonomies to reap the benefits of effective and efficient assessment that is aligned with learning outcomes. In the pursuit of this research an NLP based automatic question categorization technique with the development of a rule set was practiced; comparatively the results indicate the success of our method and its accuracy, given varying levels of challenging question wordings and keywords within the subject domain of computer science and engineering.

While the research efforts presented in this paper have been fruitful in achieving the objectives there are few notable future research extensions to be mentioned. The rules must be extended and examined for wider spectrum of exam questions before using in credit-bearing, mainstream academic needs, which can be suggested for extension of this work with future research. Another improvement for similarity algorithm and lemma similarity algorithm can be to exploit parallelism for fast execution. Further, different annotated POS tag patterns of questions can be identified to improve the identification of question category and for weight assignment. With the generic nature of the proposed rule-based classification method, it is fair to say that the process can be extended into different engineering domain specialisations, which we believe will be another valuable contribution to the teaching learning and assessment thereof.

REFERENCES

[1] Bloom, Benjamin S. Taxonomy of Educational Objectives: The Classification of Education Goals. Cognitive Domain. Handbook 1. Longman, 1956.

[2] Fuller, Ursula, et al. "Developing a computer science-specific learning taxonomy." ACM SIGCSE Bulletin. Vol. 39. No. 4, 2007.

[3] Swart, Arthur James. "Evaluation of final examination papers in engineering: A case study using Bloom's Taxonomy." Education, IEEE Transactions on 53, no. 2, 2010,p.257-264.

[4] Anderson, Lorin W., David R. Krathwohl, Peter W. Airasian, Kathleen A. Cruikshank, Richard E. Mayer, Paul R. Pintrich, James Raths, and Merlin C. Wittrock. "A Taxonomy for Learning,"Teaching, and Assessing-A Revision of Bloom's Taxonomy of Educational Objectives",(Eds.) Addison Wesley Longman.", 2001.

[5] Biggs, John B., and Kevin F. Collis. Evaluating the quality of learning: The SOLO taxonomy (Structure of the Observed Learning Outcome). Academic Press, 2014.

[6] Lister, R., and Leaney, J. Introductory programming,criterion-referencing, and Bloom. Proceedings of the 34th SIGCSE technical symposium on Computer science education, Reno, Nevada, USA, ACM Press, 2003.

[7] Meyers, Noel M., and Duncan D. Nulty. "How to use (five) curriculum design principles to align authentic learning environments, assessment, students' approaches to thinking and learning outcomes." Assessment & Evaluation in Higher Education 34, no. 5,2009,p.565-577.

[8] Forehand, Mary. "Bloom's taxonomy." Emerging perspectives on learning, teaching, and technology,2010,pp.41-47.

[9] Rutkowski, Jerzy, et al. "Application of Bloom's taxonomy for increasing teaching efficiency–case study." In Proc. of ICEE2010, 2010.

[10] Thompson, Errol, Andrew Luxton-Reilly, Jacqueline L. Whalley, Minjie Hu, and Phil Robbins. "Bloom's taxonomy for CS assessment." In Proceedings of the tenth conference on Australasian computing education-Volume 78,2008, pp. 155-161.

[11] Swart, A.J. (2010). Evaluation of Final Examination Papers in Engineering: A Case Study Using Bloom's Taxonomy, IEEE Transactions on Education,Vol. 53, No.2,pp.257-264, May 2010.

[12] Smrž, Pavel. "Integrating natural language processing into e-learning: a case of Czech." In Proceedings of the Workshop on eLearning for Computational Linguistics and Computational Linguistics for eLearning, 2004,pp. 1-10.

[13] Christopher Manning,Hinrich Schütze,Foundation of Statistical Natural Language Processing.

[14] Cutrone, Laurie, and Maiga Chang. "Automarking: automatic assessment of open questions." In Advanced Learning Technologies (ICALT), 2010 IEEE 10th International Conf. on,2010, pp. 143-147.

[15] Omar, Nazlia, et al. "Automated analysis of exam questions according to Bloom's taxonomy." Procedia-Social and Behavioral Sciences 59 ,2012,pp. 297-303.

[16] Yusof, Norazah, and Chai Jing Hui. "Determination of Bloom's cognitive level of question items using artificial neural network." In Intelligent Systems Design and Applications (ISDA), 2010 10th International Conference on,2010, pp. 866-870.

[17] Willett, Peter. "The Porter stemming algorithm: then and now." Program 40, no. 3,2006,pp. 219-223.

[18] Hogenboom, Frederik, Flavius Frasincar, and Uzay Kaymak. "An Overview of Approaches to Extract Information from Natural Language Corpora." Information Foraging Lab,2010.

[19] Miller, George A. "WordNet: a lexical database for English." Communications of the ACM 38, no. 11 ,1995,pp. 39-41.

[20] Miller, George A. "Nouns in WordNet: a lexical inheritance system." International journal of Lexicography 3, no. 4,1990,pp. 245-264.

[21] Fellbaum, Christiane. "English verbs as a semantic net." International Journal of Lexicography 3, no. 4 ,1990,pp. 278-301.

[22] Gross, Derek, and Katherine J. Miller. "Adjectives in wordnet." International Journal of Lexicography 3, no. 4 ,1990,pp. 265-277.

[23] ZHANG, Shu-yu, Guo-ning DU, and Zhong-ying ZHU. "Study of semi-structured information retrival technology based on Web [J]." Systems Engineering and Electronics 5,2004.

[24] Somers, Harold. "Review article: Example-based machine translation." Machine Translation 14, no. 2,1999,pp. 113-157.

[25] Bird, Steven, Ewan Klein, and Edward Loper. Natural language processing with Python. " O'Reilly Media, Inc.", 2009.

[26] Yang, Dongqiang, and David MW Powers. Verb similarity on the taxonomy of WordNet. Masaryk University, 2006.

[27] Huang, Zhiheng, Marcus Thint, and Zengchang Qin. "Question classification using head words and their hypernyms." In Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2008,pp. 927-936.

[28] Leacock, Claudia, and Martin Chodorow. "Combining local context and WordNet similarity for word sense identification." WordNet: An electronic lexical database 49, no. 2,1998,pp. 265-283.

[29] Wu, Zhibiao, and Martha Palmer. "Verbs semantics and lexical selection." In Proceedings of the 32nd annual meeting on Association for Computational Linguistics,1994, pp. 133-138.

[30] Jiang, J. J., and David W. Conrath. "Semantic similarity based on corpus statistics and lexical taxonomy." arXiv preprint cmp-lg/9709008 ,1997.

[31] Lin, Dekang. "An information-theoretic definition of similarity." In ICML, vol. 98,1998, pp. 296-304.

[32] Yahya, A. Ali, and Addin Osama. "Automatic classification of questions into Bloom's cognitive levels using support vector machines." ,2011.

[33] Shaw, M. L. G., and B. R. Gaines. "Question classification in rule-based systems." In Proceedings of Expert Systems' 86, The 6Th Annual Technical Conference on Research and development in expert systems III,1987, pp. 123-131.

[34] Huang, Zhiheng, Marcus Thint, and Zengchang Qin. "Question classification using head words and their hypernyms." In Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2008,pp. 927-936.

[35] Fei, Ting, et al. "Question classification for e-learning by artificial neural network." Information, Communications and Signal Processing, 2003 and Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint Conference of the Fourth International Conference on. Vol. 3. IEEE, 2003.

[36] Zhang, Dell, and Wee Sun Lee. "Question classification using support vector machines." In Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval, 2003,pp. 26-32.

[37] Webster, Jonathan J., and Chunyu Kit. "Tokenization as the initial phase in NLP." In Proceedings of the 14th conference on Computational linguistics-Volume 4, 1992,pp. 1106-1110.