Towards flexible short answer questions in the Moodle Quiz

Milena Frtunic*, Leonid Stoimenov*, Oliver Vojinovic*, Ivan Milentijevic*
* Faculty of Electrical Engineering, University of Niš, Niš, Serbia
milena.frtunic@elfak.ni.ac.rs, leonid.stoimenov@elfak.ni.ac.rs, oliver.vojinovic@elfak.ni.ac.rs, ivan.milentijevic@elfak.ni.ac.rs

***Abstract*— Assessment of student's knowledge and observation of their learning progress by using learning management systems is proven to be a very good solution. Main reason for that is provided possibility for testing a large number of students at the same time and automated evaluation of all quiz attempts. However, there are still a lot of limitations regarding automatic evaluations. In order to reduce one of those limitations in Moodle LMS, this paper will present an improvement of the automatic evaluation for questions with short answers in Moodle LMS system. Within this paper, an algorithm which is based on checking the similarity between two sentences will be introduced as a new solution for assessing short answer questions created in English language.**

I. INTRODUCTION

The advancement of technology had a great influence in all aspects of an average person's everyday life. Due to this, a number of new opportunities and possibilities for more efficient learning and informing were introduced in education. These changes have led to the development of different learning management systems (LMS) that offer numerous features in both the administrative and teaching field for both educational institutes and individual tutors.

Teaching field consists from different aspects, starting from storing teaching materials and giving lessons to testing students' knowledge. Learning management systems which offer possibility for testing knowledge provide teaching staff a quick and efficient way to evaluate knowledge of a large number of students at the same time. Such systems offer some form of automatic evaluation of tests, but that automatization usually applies only to questions with offered predefined answers where student have to choose one or more answers he thinks is correct. Some systems offer possibility for automatic evaluation of questions where students write free answer that they find to be correct. Unfortunately, in those situations in order for automatic evaluation to work and question to be evaluated as correct, written answer usually has to be identical to one that is defined to be correct by questions' creator.

In those cases teaching staff use questions with free answers only when answer can be written in a single word or said in one or two correct ways. In all other situations they usually use other types of questions, such as those where student needs to choose answers she/he thinks are correct or make some matching between facts. Although, this way of forming quiz keeps it 100% possible to be automatically evaluated, it leaves space for students to select correct response by choosing randomly. Because of that, it is questionable whether the quiz results are presentation of real students' knowledge. For that reason, teaching stuff is forced to define questions where they will expect longer answers knowing that automatic evaluation probably won't work correctly. In those situations they are obligated to go manually through every test and check whether it is necessary to reevaluate question response and change final points.

The goal of this paper is to present a solution for this problem in form of improving automatic evaluation by including algorithm that will check semantic similarity between written answer and the one defined as correct by question's creator. This research is implemented on Moodle (Modular Object Oriented Dynamic Learning Environment) learning management systems. This choice was made because Moodle is one of the most popular LMS in Europe and it is in everyday use at Faculty of Electrical Engineering, University of Niš. Further, it offers a very rich module for testing students' knowledge but still has previously mentioned limitations.

The proposed solution offers new possibilities within Moodle quiz module and gets this part of the Moodle system to a higher level. The idea is to offer an improvement of automatic evaluation of questions with short answers. The main goal is to create reliable solution for evaluating few word long answers written in English language. In the first stage the proposed solution will be attached to MoodleQuiz Android mobile application [1] and testing of the system will be done on students that use mobile application for attempting Moodle quiz.

In the next part of this paper Moodle system will be further described and special attention will be devoted to the module for attempting and evaluating quizzes. It will be explained how this module operates and what options are available. After that, special attention will be given to short answer questions as they are central part of this research. Later, computational lexicon WordNet will be discussed as it is the base for checking the semantic similarity of answers. Further, original MoodleQuiz Android application will be explained along with all updates that were developed for the sake of implementation of this research. Also, attention will be given to newly developed *MatchingService* service that is one of the core tasks of this project. Its task is to check similarity between two sentences, in this case, answer

that was written by a student and answer that is predefined to be correct.

II. MOODLE AND MOODLE QUIZ

Moodle (Modular Object-Oriented Dynamic Learning Environment) is one of the most popular open source learning management systems [2]. It is used in 220 countries and has 64041 registered sites of which the majority is in America and Europe [3]. Moodle is a very scalable learning management system and presents a suitable solution for both big universities and organizations with thousands of students and individual tutors with small number of participants. Core of the Moodle system consists of courses with their resources and activities. Besides that, Moodle supports over twenty activities and modules for different purposes and scope.

Moodle quiz module is very rich module and contains number of different possibilities when creating both quizzes and questions. It has functionalities for creating and publishing different types of quizzes that can be used for automatic evaluation of student's knowledge and monitoring student's progress during the course. Moodle quiz can be used not only for establishing students' grade, but also for giving students possibility to test their knowledge while studying material and preparing for exam.

When creating the quiz, creator defines overall setup for quiz like grading method and question behaviors like possibility for multiple attempts. At this point creator chooses individual questions he wants to be used or question groups from which the question can be selected randomly.

This module consists of a large variety of question types [4]. These questions are kept in the question bank when created and can be re-used in different quizzes. Some of the basic question types are: description, true/false, short answers, essay, matching questions, multiple-choice questions, numerical, calculated, embedded questions etc. For each question type there is number of options that can be set. Some of the options are common for all question types, like setting how many points correct answer is worth. Other options depend on the question type and its specifics.

Since Moodle quiz is proven to be very useful module for evaluating students' knowledge, a number of plugins what extend regular question types were created [5]. These plugins offer new question types that extend existing ones or present totally new question types that can be very useful, depending on the area quiz and its' questions are from.

This module supports automatic evaluation of quiz attempts which does not need to be used if teacher wants to do that job manually. Each question type has its' own rules for evaluating question and calculating points. How one question is going to be evaluated is set when question is created. Within one question type creator can choose different settings for different questions based on his wishes.

Results of every quiz attempt course administrator can review in the course administration panel on Moodle system. At that point teaching staff can reevaluate the question by changing points one has won on the question and automatically change total number of point earned on the attempt.

Short answer questions

Short answer questions are questions that are answered by entering free text that can contain different types of characters entered in form of one or more words, phrases or sentence. Depending on the setting, when answering the questions one needs to pay attention on small and capital letters. This option may or may not be set depending on the creators' wish. Creator can set one or more than one correct answer. Further, he can enter more answers and assign how worth in percentage each answer is from a maximal number of points for that question. Fig. 1 presents an example of question with short answer in Moodle system.

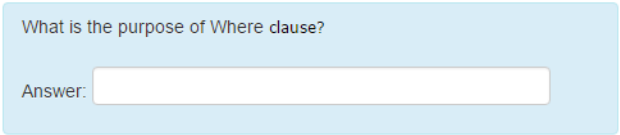

Figure 1. Example of question with short answer in Moodle system

Moodle quiz module offers automatic grading solution for quiz attempts. This works for all question types except essays which have to be evaluated manually and short answer questions in some situations. Automatic evaluation of questions with short answers checks whether answer written by student is identical to one defined by questions' creator. If it's the same, students gets the points, otherwise he doesn't. In situations where there are more than one answers defined for one question and they are worth different percentages of maximum number of points, student receives percentage that corresponds to the answer that is identical to his.

This system of evaluation leaves possibility for answers that are formulated in a different way not to be properly assessed because they are not identical with defined ones. Such situation can be very common since usually same thing can be said in a several ways and one sentence can be phrased differently. In those situations manual reevaluation of such questions is needed and this process can take a lot of time and energy if many students have attempted that quiz.

To avoid manual reevaluation of quiz attempts, question answers have to be specific so that they can be expressed in only one way. Other solution is answers to be very short so that possibility for evaluation errors is minimized. In order to overcome these limitations within this paper a solution for evaluating accuracy of the answers by semantic similarity is proposed. All details of the proposed solution will be explained more closely later in this paper.

III. WORDNET

For the purpose of this research it is necessary to include Natural Language Processing (NLP) in order to assure successful sentence comparison. Word Sense Disambiguation (WSD) is considered to be one of the core tasks in Natural Language Processing [6]. Its purpose is to assign for each word in the sentence appropriate sense(s) and for that purpose supervised and unsupervised methods can be used.

A majority of WSD methods use external knowledge sources as central component for performing WSD. There are different knowledge sources such as ontologies, glossaries, corpora of texts, computational lexicons,

thesauri etc. WordNet is one of the external knowledge sources that has been widely used for performing WSD [7]. It is computational lexicon for English language created in 1985 under the direction of Professor George Armitage Miller in the Cognitive Science Laboratory of Princeton University. Over time, many people gave their contribution to WordNet development and today the newest version (3.1) is available on the Internet and consists of 155287 words organized in 117659 *synsets* [8].

WordNet database consists of nouns, verbs, adjective and adverbs grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Each synset represents a structure, which consists of a term, its class, connections with all semantically related terms and brief illustration of the use of the synset members. Most frequently used semantic relations are: hypernymy (kind-of or is–a), hyponymy (the inverse relations of hypernymy), meronymy (part-of) and holonymy (the inverse of meronymy). In figure 2, one example of the hyponym taxonomy in WordNet is presented.

WordNet can be efficiently used in a number of unsupervised methods which introduce semantic similarity measures for performing word disambiguation. In such cases WordNet is used to determine the similarity between words. Rada et al. [9], Leacock and Chodorow [10], Wu and Palmer [11], have successfully used WordNet as a base for creating graph-like structure in order to perform word similarity measurement. Within this paper Wu and Palmer method will be used for measuring similarity between words.

IV. IMPROVEMENT OF AUTOMATIC ASSESSMENT OF QUESTIONS WITH SHORT ANSWERS

The main purpose of this paper is to propose the improvement of automated assessment of Moodle questions with short answers. The goal is this solution to be used on questions which answers are a few word long sentences. Because of that, this solution won't be used to automate the evaluation of essays in Moodle system. The solution applies only for answers written in English language. This goal is implemented in a Moodle quiz application that was developed for solving Moodle quizzes on Android mobile devices.

In order to implement this improvement, it was necessary to realize two things:

• Create a web service that compares two sentences and returns their similarity and

• Improve MoodleQuiz application in order to support communication with service and develop new mechanism for assessing questions with short answers.

It the next part of this paper both components will be discussed in detail.

A. *Service for sentence similarity measurement*

For the purpose of comparing two sentences, in this case student's answer and correct answer defined by professor, *MatchingService* was created. The service is designed for English language only and relies on WordNet as a resource for English words. As an input, service requires two sentences for which similarity should be calculated. After processing service returns similarity percentage expressed with value from 0 to 1. Algorithm implemented within MatchingService actually consists of two independent algorithms, one for creating similarity matrix for two sentences and the other for calculating sentence similarity.

Algorithm for creating similarity matrix is preformed first. At the beginning it receives two sentences and preforms tokenization. Tokenization presents partitioning sentences into lists of words ($List_a$ and $List_b$) with removing all stop words (frequently occurring, insignificant words). Further, an array with Part of speech (POS) words is created. This array contains all types of words that can be used for comparison: noun, verb, adjective, adverb, unknown and other.

At this point everything is set for creation of similarity matrix which represents the similarity of each pair of words from the arrays ($List_a$ and $List_b$). This means that if the arrays have length *m* and *n*, created similarity matrix will have dimensions *mxn*. For each pair of terms in the arrays, similarity is checked in two ways and final result is better of the two.

Within the first way, the similarity among the characters in the words is checked. In this way the result is 1 (100% similarity) if the words are identical.

Within the second way, WordNet computational lexicon is used as a resource for determination of word type and appropriate sense in the sentence, as well as similarity with other words in the sentence. In order to achieve this, a type from a POS array is assigned to every word in a pair (T_a, T_b) and for such combination a

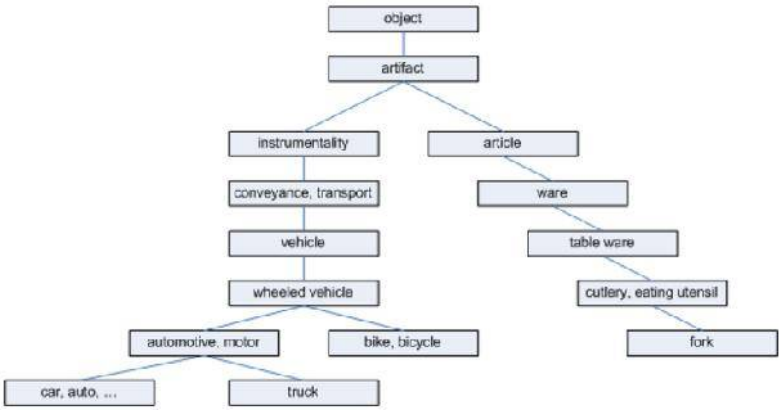

Figure 2. Example of the hyponym taxonomy in WordNet

semantic similarity is checked $sim(T_a, T_b)$. It is done by measuring the path length to the root node from the last common ancestor in the graph-like structure created with WordNet resource for information and relations among terms. After having path lengths calculated Wu and Palmer calculation is used for final determination of similarity for one POS combination for $sim(T_a, T_b)$. The calculation is done by scaling measured distance between the root node and the least common ancestor of the two concepts with the sum of the path lengths from the individual terms to the root.

Such calculation is performed for every combination of pair (T_a, T_b) with elements from POS array and final similarity is the highest result from all combinations.

After both algorithms for word similarity are performed, better result from both algorithms is inserted in similarity matrix. This procedure is repeated for every pair of words in $List_a$ and $List_b$.

After having similarity matrix created, algorithm for calculating similarity between entered sentences is performed. Within this prototype a heuristic method of calculation is used based on the similarity matrix. Following method is used:

$$score = (sumSim\_i + sumSim\_j)/(m+n),$$

where:

score – is result of final similarity of similarity matrix and can have value in range of [0,1];

m i n – are similarity matrix dimensions;

sumSim_i – sum of maximal elements in matrix per column:

$$sumSim\_i = \sum_{i=0}^{m-1} \max(i),$$

sumSim_j – sum of maximal elements in matrix per row:

$$sumSim\_j = \sum_{j=0}^{n-1} \max(j),$$

After final calculation of score is finished, the result is sent back to the client.

B. *MoodleQuiz application*

MoodleQuiz is an Android application developed for attempting Moodle quizzes on mobile devices [1]. Application is designed for devices with Android operating system version 2.2 or higher. Application is designed to support four basic types of Moodle questions that are commonly used at Faculty of Electrical Engineering in Niš: true/false questions, questions with short answers, matching questions and multichoice questions. Fig. 3 represents an example of question with short answer in MoodleQuiz application.

MoodleQuiz application is supported by Moodle Web service which is responsible for communication with Moodle system and access to all necessary data. Further, this service does all needed updates in Moodle database in order to assure consistency in Moodle system and assure that no difference is noticed between quizzes attempted in Moodle system and ones attempted on MoodleQuiz application. The whole system is designed to be compatible with Moodle version 2.5. Within this project an update is made, and now the system is compatible with Moodle version 2.8.2.

Within this project no changes were made in user interface of Android application, so users won't notice any changes in their user experience. Most changes were made in part of the application which is executed after the attempt is finished. In original application, quiz submission included assessment of all questions in the quiz and calculation of final score. This was done based on quiz and question setup, limitations set by quiz designer, question designer and official Moodle documentation. After that, final score along with other information is sent to the server and inserted into Moodle database, so that it can be available in administration panel in Moodle system.

Figure 3. Example of question with short answer in MoodleQuiz application

New version of the application made for this research has new actions added in order to support communication with MarchingService service. The service is called in POST method. When it is called, two sentences are forwarded in JSON format, correct answer and answer given by student in the attempt. Since the service returns percentage of the similarity between the send sentences, final score on the question is formed by multiplying the maximal number of points one can score on the question with similarity percentage received from the service.

This service is called during the preparation of the results for sending to Moodle server. When preparing the results, each time question with short answer appears, MatchingService is called and recalculation of point on the question and total score is done. Since each question with short answer can have more than one answer defined, service is called for each defined answer separately and for final result the highest similarity percentage between student's and correct answer is taken.

By testing this method for assessing question with short answers it was concluded that in cases where percentage was lower than 20%, student gave incorrect answer. For

that reason, in those cases correction of the points on that question is introduced and student receives 0 points. When percentage is higher than 20%, total score is calculated like described. After all questions have total marks, final score is calculated and update of the Moodle database is done. After that moment, course administrators can review the attempt normally in Moodle system.

Table 1 presents examples of answers defined by teachers and how some of the answers were evaluated by system and teachers. Given examples have results for two questions. Correct answers in rows with number 1a and 1b are correct answers for one question and row with number 2 for other question. Since number 1a and number 1b belong to the same question, one written answer is compared with both correct answers.

As it can be seen, results given by algorithm proposed in this paper are pretty accurate for question which answers is marked with number 2. For other answers the results were not that accurate. However, since both correct answers belong to the same question better percent will be taken for calculation of the points for that question. Having that in mind, final calculation is not too imprecise. Nevertheless, answer "*to filter database entries*" was badly evaluated in both combinations. This indicates that proposed semantic similarity evaluation can't be taken for granted and in order to work question's creator has to offer few combinations of word choices.

TABLE I.
ILLUSTRATION OF THE CORRECT AND WRITTEN ANSWERS AND RESULTS RETURNED FROM THE SYSTEM

No.	**Correct answer**	**Written answer**	**Systems result [%]**	**Teacher evaluation [%]**
1a	Where clause specifies conditions in the query	to filter database entries	43	100
		purpose is to specify conditions	67	90
		Where clause limits rows	86	100
		specifies conditions in the query	83	100
		Where clause specifies conditions in the query	100	100
1b	Where clause limits which rows will be returned	to filter database entries	51	100
		purpose is to specify conditions	54	90
		Where clause limits rows	95	100
		specifies conditions in the query	62	100
		Where clause specifies conditions in the query	83	100
2	Virtualization refers to the act of creating a virtual (rather than actual) version of something, including virtual computer hardware platforms, operating systems, storage devices, and computer network resources.	virtualization creates virtual version of something	69	70
		creates virtual version of something like computer resources	80	80
		creation of virtual computer platform, operating system, storage device or resources	89	90
		lorem ipsum lorem ipsum lorem ipsum lorem ipsum lorem ipsum	0	0
		virtualization refers to creating virtual version of hardware platform, operating system, storage devices, computer network resources	96	100

V. CONCLUSON AND FURTHER WORK

In this paper an improvement of automatic assessment of Moodle questions with short answer is presented. Offered solution provides much more flexibility while creating questions with short answers and testing students' knowledge. This primarily refers to the ability to define question with short answer whose answer can be a short sentence and have a confidence that it will be correctly automatically evaluated. Offered solution increases usability where this type of questions can be used and still assures that whole quiz can be totally automatically evaluated the moment it is submitted. Proposed solution reduces the necessity to use different types of question which offer answers in order to maintain 100% automatic evaluation of the whole quiz and save time.

Solution presented in this paper is currently in test phase and checking the reliability of comparing sentence similarity. Based on the results from table 1 it can be concluded that proposed algorithm made progress in comparison with current Moodle system's evaluation algorithm. However, at this point in order to expect better results question's creator should enter more correct answers with different word choices in the answers.

The plan is to test system's reliability on course Database at Faculty of Electrical Engineering, University of Niš. Based on the results of the mass testing, the algorithm will be further improved in order to provide better results. At this point, the goal is to assure correct evaluation of given answers that contain few words and minimize the possibility of error and the need to go manually through tests and evaluate each question separately. After that, the aim is to define the rules that should be followed when formulating questions and answers in order to get the best possible results.

If the system proves to be reliable in the next phase it will be transferred on Moodle system directly. With this step it will become available on Moodle system and dependence on MoodleQuiz application will no longer exist. In this way all Moodle system users will have the possibility of using this solution and it won't be limited on Android users only.

ACKNOWLEDGMENT

The research presented in this paper was funded by the Ministry of Education, Science and Technological Development of the Republic of Serbia as part of the project "Infrastructure for electronically supported learning in Serbia" number III47003.

REFERENCES

[1] M. Frtunić, L. Stoimenov, and D. Rančić, "Incremental Development of E-Learning Systems for Mobile Platforms," *ICEST 2014*, vol. 1, pp. 105–108, 2014

[2] J. Hant, "Top 100 Tools for Learning 2013," *7th Annual Learning Tools Survey*, September 2013

[3] "Moodle Statistics," 2015 [online] https://moodle.net/stats/

[4] "Moodle Questions," 2015. [online] https://docs.moodle.org/25/en/Question_types

[5] "Question Types, " 2015 [online] https://moodle.org/plugins/browse.php?list=category&id=29

[6] R. Navigli, "Word sense disambiguation: A survey," *ACM Computing Surveys (CSUR) Surveys*, vol. 41 issue 2, article no. 10, USA, 2009

[7] C. Fellbaum, "WordNet: An Electronic Lexical Database, Language, Speech, and Communication," *The MIT Press*, 1998.

[8] "WordNet statistics," 2015 [online] http://wordnet.princeton.edu/wordnet/man/wnstats.7WN.html

[9] R. Rada, H. Mili, E. Bicknell and M. Blettner, "Development and application of a metric on semantic nets," *IEEE Trans. Syst. Man Cybern.* 19 (1) (1989) 17–30.

[10] C. Leacock and M. Chodorow, "Combining local context and WordNet similarity for word sense identification," C. Fellbaum (Ed.), WordNet: An Electronic Lexical Database, *MIT Press*, pp. 305–332, 1998

[11] Z. Wu and M.S. Palmer, "Verb semantics and lexical selection," *Proceedings of the 32th Annual Meeting on Association for Computational Linguistics*, pp. 133–138, Las Cruces, New Mexico, 1994