2011 IEEE International Conference on Technology for Education

Auto-Assessor: Computerized Assessment System for Marking Student's Short-Answers Automatically

Laurie Cutrone
Information Systems Technology
Red River College
Winnipeg, Canada
lcutrone@rrc.mb.ca

Maiga Chang and Kinshuk
School of Computing and Information Systems
Athabasca University
Athabasca, Canada
maiga@ms2.hinet.net and kinshuk@athabascau.ca

***Abstract*—A number of Learning Management Systems (LMSs) exist on the market today. A subset of a LMS is the component in which student assessment is managed. In some forms of assessment, such as short-answer questions, the LMS is incapable of evaluating the students' responses and therefore human intervention is necessary. This study leverages the research conducted in recent Natural Language Processing studies to provide a fair, timely and accurate assessment of student short-answers based on the semantic meaning of those answers. A component-based system utilizing a Text Pre-Processing phase and a Word/Synonym Matching phase has been developed to automate the marking process. An evaluation plan is also made to verify the possibility of applying such computerized assessment system in practical situations as well as to reveal areas in which the system could be improved later.**

Keywords-natural language processing; WordNet; short-answer question; computerized grading; semantic meaning

I. INTRODUCTION

Learning Management Systems provide a number of advantages over face-to-face delivery. The most obvious advantage is the flexibility that the LMS provides. This flexibility can be realized by both the student and the teacher. No longer is it necessary for seats to be filled in a classroom during a specific time slot. This is especially important in situations in which the student and/or the teacher can replace their physical presence in a classroom with a virtual presence.

There are a number of commercial assessment tools on the market today; however these tools support objective-style questioning such as multiple-choice questions [13]. Multiple-choice questions will assess knowledge through the student's recall ability. However, multiple-choice questions will not assess the learner at the higher levels of Bloom's (1956) taxonomy of educational objectives [2][5][13]. In order to assess at a higher level, it is necessary to include open-style questions in which the student is given the task as well as the freedom to arrive at a response without the comfort of recall words and/or phrases. In assessing open questions written in the traditional paper and pen format, the assessor would provide feedback directly on the student responses in order to indicate areas of correctness or incorrectness in a way of justifying the final grade given. However providing feedback using LMS software is awkward and quite time-consuming compared to the paper and pen counterpart.

To provide a mechanism in which the LMS software would be capable of accurately assessing the students' responses to open questions would alleviate the shortcomings described above. The system would allow for students to be evaluated at the higher levels of Bloom's (1956) taxonomy in that the students would be asked to state, suggest, describe or explain an answer [27], rather than simply recall an answer. The system would be able to provide appropriate feedback which would be absent of biases influencing the overall grade of a question. Moreover, the student responses could be evaluated in a timely manner without the need for teacher intervention.

Automating the assessment process of open questions is an area of research that has been ongoing since the 1960s [8][13][17]. However recent advances in the areas of Information Extraction [6][31][35] and Information Retrieval [7][10][16][23] have allowed for alternative approaches to be explored. Earlier work in the area of Natural Language Processing, with respect to assessing responses to open questions, focused on a statistical or probabilistic approach [18][12][17][23].

These approaches, while successful, focused heavily on conceptual understanding. The semantic meaning of the text was never evaluated. Rather, the location of specific words and/or phrases, and the number of occurrences of such words was being evaluated. Recent gains in Natural Language Processing have resulted in a shift in the way in which free text can be evaluated. Work in the area of Information Extraction has made significant gains in actually determining the semantic meaning of natural language text [6][31][35]. This has allowed for a more linguistic approach which focuses heavily on factual understanding [3][33].

This study leverages the research conducted in recent Natural Language Processing studies to provide a fair, timely and accurate assessment of student responses to short-answer questions based on the semantic meaning of those responses.

Section 2 discusses the contributions that many researchers have made in the domains of Natural Language Processing as well as automated essay grading. Section 3 discusses the proposed methodology for this study. Section 4 focuses on the implementation of the proposed system – Auto-Assessor. Section 5 provides a system evaluation plan to verify the possibility of applying such computerized assessment system in practical situations as well as to reveal areas in which the system could be improved later. Section 6 makes conclusions at the end.

978-0-7695-4534-9/11 $26.00 © 2011 IEEE
DOI 10.1109/T4E.2011.21

II. AUTOMATIC ASSESSMENT AND NLP

A. Automatic Assessment Methods

There has been an interest in automatic assessment of open questions since the 1960s [8][13][17]. During this same era, there has been a great interest in Natural Language Processing. Natural Language Processing (NLP) involves using computers to identify semantic relations among human words [14]. It involves various dimensions of human language including grammar, usage and semantics [20]. Countless studies have attempted to decipher free text. [3] suggests work in the area of Natural Language Processing falls along a Text Technology Continuum in Natural Language Processing as shown in Figure 1 below.

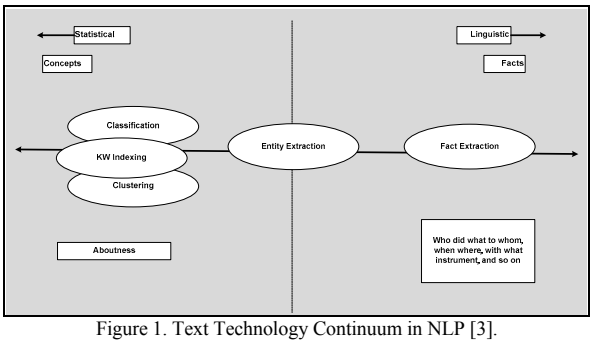

Figure 1. Text Technology Continuum in NLP [3].

An early notable contribution to automatic essay grading was that of Project Essay Grader (PEG). [13] indicates, however, that PEG was not widely accepted because it considered things such as the number of commas, or the number of uncommon words, yet it is notable as one of the early essay grading automated tools. PEG was an early attempt at the statistical approach to automatic essay grading.

E-Rater developed by Educational Testing Services (ETS) was a significant contribution to the assessment of open questions [13][27]. E-Rater included the structure of the text as part of the assessment process, and therefore incorporated some linguistic features. This system identified syntactic and speech features. The content of the text is compared to predefined content words. An essay that contains appropriate content words, reasonable speech features, and uses good vocabulary would receive a higher grade [27]. However, an equivalent response that failed to use the predefined content words would not receive a higher grade. This is acceptable when specific terminology must be used in the student responses. E-Rater has been successfully implemented to assess Graduate Management Admission Test (GMAT) exams since 1999 [30].

Much of the work that followed tended to continue along the statistical end of the Text Technology Continuum in Natural Language Processing (see Figure 1). Subsequent work was greatly influenced by an approach known as Latent Semantic Analysis (LSA) [12][32]. LSA uses a 'bag of words' approach in which similarity and co-location of words is evaluated [8]. LSA is a corpus-based text comparison approach and uses an algebraic technique to determine the level of similarity between the text and the corpus [12]. LSA uses word-document co-occurrences based on the corpus and presents these in a vector space [27]. LSA assumes a relationship between the meaning of text and the words used in that text. Therefore two texts that use similar words would be considered semantically similar using LSA. Texts with similar wording would be mapped closer together in the vector space [12]. This sort of approach requires a reasonable corpus to start with, and depending on the domain, the corpus may require regular updates. An additional problem inherent with LSA is that the order in which the words are presented is not considered important [8]. Therefore, the sentences: "*the boy stepped on a spider*" and "*the spider stepped on a boy*" would be considered equivalent.

A number of approaches to automatic grading of open questions have been developed with excellent success rates (80-90% agreement with a human-grader gold standard) [4][9][12][17]. However, when leaning toward a statistical and/or probabilistic approach, there are a couple of considerations to make. First, many of these approaches require a large corpus of a previously evaluated knowledge base (such as previously graded essays) [17][32]. In many domains, the content being assessed may evolve significantly from one year to the next. This would require that the corpus be updated on a regular basis thereby potentially negating some of the time-based benefits realized from the automatic assessment tool. Secondly, using the 'closest match' and probability techniques, it is possible for a student to 'beat the system' simply by providing a number of keywords in their response, yet not accurately answering the question, or conversely, a student could answer a question accurately, yet not provide the proper keywords which results in a less-than-perfect grade.

B. Natural Language Processing

Despite the success rates of the automatic grading systems developed thus far, there is still an underlying problem with the past approaches. These approaches failed to attempt to equate the meaning of the student response to an appropriate grade. Instead, these approaches used combinations of matching algorithms, statistics, and probabilities supported by corpus and the like to make a reasonable estimate at an appropriate grade. The trend that has remained thus far in much of the previous work in automatic essay grading is that most studies have remained on the statistical end of the Text Technology Continuum in Natural Language Processing (see Figure 1). This is despite the fact that great inroads have been made in Natural Language Processing which would support an approach closer to the linguistic end of the continuum.

A significant offering to Natural Language Processing in recent years has been the development of WordNet by George A. Miller of Princeton University. [26] describes WordNet as a database containing the lexical and conceptual meaning of more than 150,000 words. Words are arranged based on the relations among them. WordNet focuses on the semantic relationships between words much like a thesaurus. It allows for searching of concepts through other words that imply the same meaning. WordNet divides the words into four categories based on part of speech. These categories are nouns, verbs, adjectives and adverbs. WordNet's basic unit is the synonym set, known as the synset. Each synset is composed of synonymous words along with pointers to related synsets. WordNet maintains both lexical and

semantic relations among the synsets include Synonymy, Hypernymy/Hyponymy, Antonymy, Meronymy/Holonymy, Lexical Entailment, and Troponymy [24].

Part of Speech (POS) Tagging is a technique that has been widely used in Information Extraction Systems [11][15]. POS Tagging involves dividing documents into paragraphs, and then further dividing the paragraphs into sentences and phrases. Each word in each sentence is tagged with its corresponding part of speech element such as nouns, adjectives, adverbs, verbs and pronouns [11][15]. There are a number of POS Tagging tools available, some of which also perform sentence chunking to produce noun phrases and verb groups. One such tool is SharpNLP. SharpNLP, maintained by codeplex.com, provides a number of Natural Language Processing tools written in the C# programming language. Text is tagged using SharpNLP based on the Penn Treebank Tagset [21].

Much of the previous work in Natural Language Processing has incorporated a Text Pre-Processing phase in which the natural language text is prepared for the larger task of gaining a semantic understanding of the text [1][15]. Text Pre-Processing involves techniques such as chunking, stemming, removing stop words and tagging all in an effort to reduce sentence or phrase to its canonical form.

Stemming is a technique which removes suffixes in order to determine the root or stem of a word [23]. Chunking is the process of dividing sentences into noun phrases and verb groups [19]. For example the sentence: Encryption is a mathematical formula that is applied to electronic data would be chunked as follows: {Encryption} is a {mathematical formula} that is {applied to} {electronic data}. Each chunk of the sentence can then be further processed based on the part of speech that the individual words represent within each chunk. Stop words are words such as pronouns, adjectives, adverbs and prepositions such as the, are, and, of, and in [23]. Although these words make a sentence grammatically correct, they do not contribute to the semantic meaning of the text. Studies have shown that the accuracy is improved when stop words have been removed [23].

This study makes use of the recent advances in Natural Language Processing to develop a system capable of automatically assessing short-answers in a manner that assesses the student response based on its linguistic features. While this study focuses on linguistics, the tools utilized such as WordNet and SharpNLP do have a statistical undertone. The system reduces the supplied answer as well as the student response to their canonical form through a comprehensive Text Pre-Processing phase. All words in the canonical form are tagged based on their part of speech. The student response and the supplied answer are then compared. In this comparison, features encapsulated within WordNet are utilized to ensure that exact word matches are not necessary in determining the level of equivalency between the student response and the supplied answer.

III. Proposed Architecture

A. Objectives and Assumptions

The primary focus of this system is to produce software that focuses on the linguistic end of the Text Technology Continuum in Natural Language Processing (see Figure 1). In particular, the focus is on determining the semantic meaning of the student responses. This system has been developed with a goal in mind that there be a tremendous amount of flexibility in the way in which the student response is worded.

The proposed system therefore has seven assumptions/objectives:

1) The system has been developed for the English language, and will therefore be based on English language part of speech elements.

2) Although grammar and spelling are critical components in learning within any domain, grammar and spelling will not be the focus of this system. Therefore, a primary assumption in this system is that all student responses as well as supplied correct answers are entered by the end users using proper spelling and grammar using complete sentences.

3) The system requires that the supplied answers as well as student responses be written in a direct manner. That is, all answers must not use analogies, slang or examples.

4) The system is supported by the generic WordNet ontology.

5) This system focuses on single-sentence responses.

6) In this early version of the software, it is assumed that each word in the reduced correct answer holds equivalent weight in the overall grade value.

7) The system was developed using a component-based architecture. See System Architecture subsection below.

B. System Architecture

This system utilizes a component-based architecture. The components created in order to reduce the sentences to their canonical form are used in both the pre-processing of the supplied correct answer as well as the student response. The basic architecture of the system is shown in Figure 2 at the end of this paper, and is described in the following sections.

1) Assessor User Interface

a) Natural Language Question Text Editor: Editor in which the assessor enters the open question(s) in natural language for use in student evaluation.

b) Natural Language Correct Answer Text Editor: Editor in which the assessor enters the correct answer using natural language. Note: The system assumes that the assessor will respond in complete sentences, using proper grammar and spelling.

2) Student User Interface

a) Question Display: Interface in which the student is presented with the open question(s).

b) Natural Language Student Response Text Editor: Editor in which he student is able to use natural language to respond to the question(s) that appears in the Question Display.

3) Text Pre-Processing - The Text Pre-Processing component is comprised of a number of steps which run sequentially in an effort to reduce each sentence to its

canonical form [1][15]. These steps are applied to both the correct answer (CA) and the student response (SR). Additionally, a portion of these steps are applied to the Question.

a) Text Tagging: Text tagging involves applying POS tags to each word in the sentence. In addition, certain words within the sentence given an additional tag to indicate that thosee words are the beginning words in 'chunks'. This will help to identify noun phrases and verb groups as necessary during later phases in the Text Pre-Processing phase. In this study, these tasks will be accomplished using SharpNLP. For example, the sentence: Encryption is an algorithm applied to electronic data. would be processed using SharpNLP's POS Tagger as shown in Figure 3 below.

Encryption is an algorithm applied to electronic data.	
Encryption/NN is/VBZ an/DT algorithm/NN applied/VBN to/TO electronic/JJ data/NNS ./.	
NN	Noun, singular
VBZ	Verb, 3sg
DT	Determiner
VBN	Verb, past participle
TO	To
JJ	Adjective
NNS	Noun, plural
.	Sentence final

Figure 3. POS Tagging Example [22].

b) Remove Punctuation: The POS Tagger used in this project applies tags to punctuation using a different format than the tags applied to words. In order to alleviate problems associated with the punctuation tags, all punctuation is removed from the sentence.

c) Remove Question Words: In order to prevent a student from being credited for simply repeating the question words in their response, all question words are removed from the student response as well as the supplied answer. The question words to be removed are based on the canonical form of the question. As such, the question must also be subjected to a portion of the Text Pre-Processing phase prior to this step.

d) N-Gram Detection: It is important to recognize word groupings that connote a single meaning. These include compound words or proper nouns [29]. For example, the word groupings 'telephone directory' and 'National Hockey League' should not be split even though they are comprised of individual nouns that, in and of themselves, connote meaning. This pre-processing step will re-tag any identified n-grams so that the true meaning of the sentence is captured.

e) Reverse Context: Natural language text can have a variety of morpho-syntactic variations which are equivalent semantically [32]. In some cases, a sentence can be stated in a reverse form which is equivalent to a more direct approach. For example: "Encryption is a process in which a mathematical formula is applied to electronic data" and "Encryption is the process of modifying electronic data by applying a mathematical formula."

f) Stop Word Processing: In this step, the tagged text is examined to determine whether any stop words exist. If so, the stop words are removed from the text [19][29]. This causes most sentences to be grammatically incorrect. However, the semantic meaning of the sentence remains. A complete sentence is shown below followed by the same sentence with stop words removed: from "Encryption is an algorithm applied to electronic data" to "Encryption algorithm applied electronic data."

g) Stemming: In the stemming phase, individual words are reduced to their canonical form or stem. The canonical form of a word is the base or lemma of that word [15][19]. For example the canonical form of the words artist and artisan is art. In order to reduce a sentence to its canonical form, the individual words within the sentence must be examined to ensure that they are also in their canonical form. Stemming simplifies the process of locating synonyms which takes place following the pre-processing phase.

4) WordNet Processing - Following Text Pre-Processing, the actual evaluation of the student responses based on the correct answer takes place. Each word in the Correct Answer in Canonical Form (CFCA) is compared to the corresponding word(s) in the Student Response in Canonical Form (CFSR). This process makes use of WordNet.NET, a .NET version of WordNet developed by Troy Simpson and maintained by Ebswift [34]. The words are first compared for an exact match. An exact match is determined based on:

a) A matching part-of-speech tag

b) A word match

c) The words that have been matched have an equivalent relative position in the sentence with respect to the sentence verb(s) (if any exist).

IV. The System – Auto-Assessor

A number of factors were considered when deciding on a development methodology. Cost for development is another consideration. Freeware and open source solutions were utilized whenever possible as long as they provide an acceptable solution.

A variation of the Agile development methodology was used. Agile is an iterative approach which supports the rapid evolution of solutions. The system is comprised of a number of independent components. These components were developed individually and as the project progressed, the components were integrated with one another when appropriate. This allowed each component to be individually developed, tested and implemented without affecting the development of the other components.

A. Development Tools

The development tools selected for this project were selected based on preliminary analysis of the proposed system. As the Agile development process continued, evolutions of the system emerged. During these evolutions, some development tools emerged as appropriate for the project at that particular time. Any costs associated with the development of this system were borne by the

developer. The tools listed below were selected for this system. The justification of these selections is provided below.

WordNet.NET was selected as the tool used to decipher whether appropriate synonyms exist in the student response. Other WordNet-like tools exist such as the Information Content tool created by [28], as well as WordNet-like tools created for languages other than the English Language [24][25]. WordNet was selected because it is one of the largest lexical databases for the English language. Additionally, WordNet provides well-documented open source [26]. WordNet has been widely utilized in the Natural Language Processing domain, and as such, many projects were referred to while pursuing this project in an effort to best utilize the tool. As well, WordNet has a .NET version available. Finally, WordNet is a free download which satisfied budgetary concerns.

SharpNLP was selected as the POS Tagger for this project. SharpNLP was selected because it has a .NET version which will allow for integration with the .NET version of WordNet. Additionally, SharpNLP is encapsulated with a chunker which support some of the tasks required within the Text Pre-Processing phase. SharpNLP is also a free download which satisfied budgetary concerns. As well, SharpNLP is accompanied by extensive documentation as well as an active forum.

As a result of the Natural Language Processing tools mentioned above, the development tool selected for this project is C#. C# was selected based on the .NET versions of WordNet and SharpNLP. C# allows for the system to be developed as a Windows- or Web-Based application offering greater flexibility to the end-user.

B. The Prototype System

The experiment system has been developed as a Windows application. The system is presented in a Multi Document Interface style and depending on the logon credentials supplied, a different set of windows are supplied to the user. Three different user types will use the system. These types include the Assessor, the Student and Operations personnel.

1) The Assessor

The Assessor is responsible for creating the test. This includes creating and/or selecting the questions that will be included in the test. Figure 4 shows the Test Designer interface.

When creating new questions, the Assessor must provide the question as well as the correct answer as shown in Figure 5. The question and the correct answer must be free of any spelling or grammar errors. In addition, the correct answer must be comprised of a single sentence.

2) The Student

The Student has the ability to take a test and review test scores. When taking a test, the student is presented with an interface that allows for navigation among the test questions. The Student is presented with the test question and an editor in which a response can be composed as shown in Figure 6. The student response must be free of spelling or grammatical errors, and must be formulated in a single sentence.

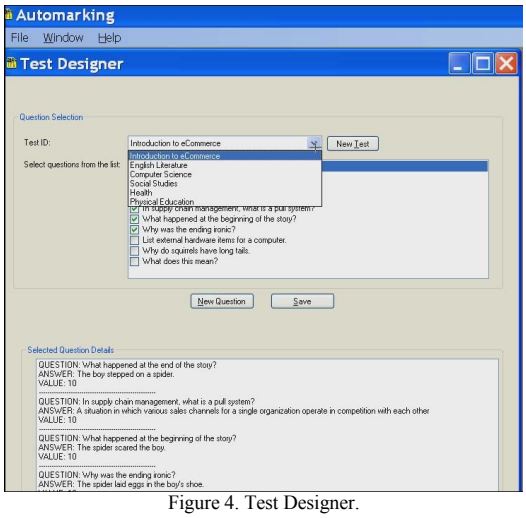

Figure 4. Test Designer.

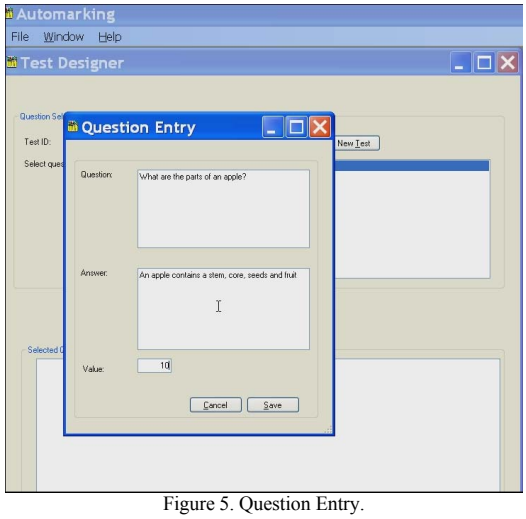

Figure 5. Question Entry.

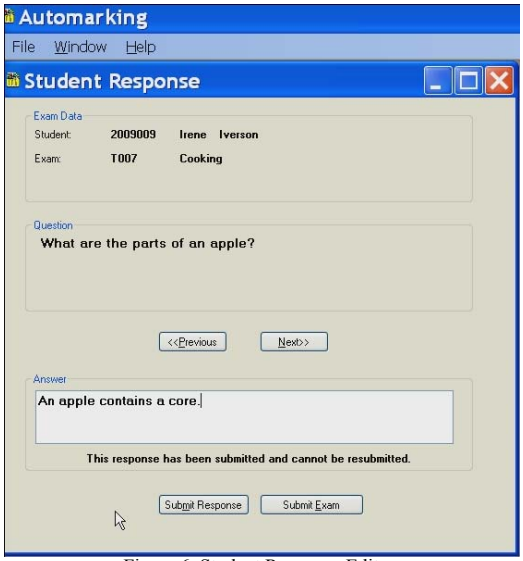

Figure 6. Student Response Editor.

When reviewing test scores, the Student is provided with the list of the questions that had appeared on the test,

the student responses to those questions as well as the calculated score for each question as shown in Figure 7.

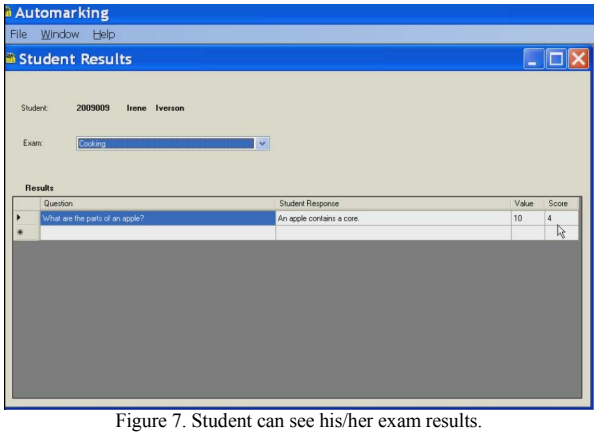

Figure 7. Student can see his/her exam results.

3) The Operator

In the experiment system, an Operator role was established to perform the operation of grading the tests. While this process would eventually be set up as a regular batch job, the experiment system was designed such that an end user would initiate this process. The operator selects the exam to be evaluated and manually initiates the grading process as shown in Figure 8.

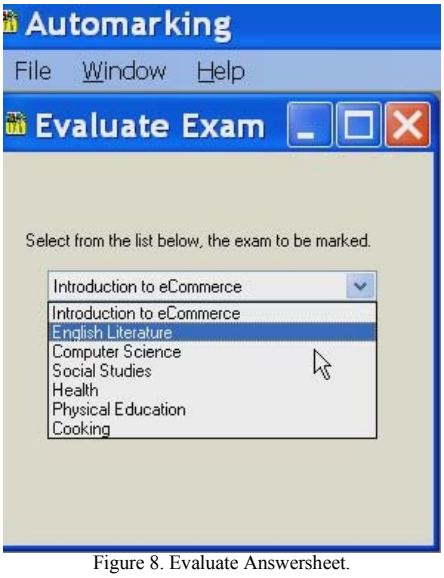

Figure 8. Evaluate Answersheet.

V. EVALUATION PLAN

A prototype system has been developed using the architecture described above and shown in Figure 2. The prototype system contains the user interfaces described in the System Architecture section above. Both the correct answer and the student response are processed using the text preprocessing steps in order to reduce these sentences to their canonical forms. The canonical forms are then compared to one another to determine their level of equivalency. This comparison incorporates all synonyms of the canonical words in order to allow for more flexibility in terms of the choice of words in the correct answer as well as the student response. The degree to which similarity of words is determined is based on the distance between words within the WordNet hierarchy. Essentially, the greater the distance, the less related two words are.

Early testing of the prototype system has produced some encouraging results. For example, the sentence "the boy stepped on a spider" and "a spider was stepped on by the boy" are considered equivalent. While a third sentence: "a spider stepped on the boy" is not considered equivalent to the other two. This comparison is successful as a result of the reverse context component in the text pre-processing phase which looks for 'was/by' phrasing, and simplifies the sentence by reversing the text and removing the 'was/by' phrasing. Additionally, the third sentence is considered incorrect based on the placement of the nouns (spider and boy) with respect to the verb (stepped) even though it contains all of the words found in the first sentence.

Much of the previous work in the area of open question assessment used a benchmark standard, which is regarded as a definitive point from which to make comparisons, often referred to as the gold standard. The gold standard in previous work has been based on a comparison between the system test results and the test results utilizing one or more Human Graders [22][33].

This study is going to invite two independent Human Graders to grade all student responses for all of the short-answer questions. The Human Graders will be provided with an answer key containing a single correct answer. Both Human Graders need to use the single correct answer for each question as a benchmark as is common for multiple teachers to follow the same answer key when administering the same test to different groups of students. The Human Graders will also be given some guidelines in terms of the range of acceptable answers. The time spent in grading will be recorded as well. Each Human Grader is not aware of the grading strategies and the results of the other Human Grader aside from knowing that the other Human Grader is provided with the same answer key and grading guidelines.

One of the criticisms of human grading of open questions is that biases and opinions can influence the overall grade provided [13]. In an effort to prove the effectiveness of a non-biased assessment tool such as the one developed in this project, the Human Grader results are compared to determine the level of agreement among the graders.

The system will then automatically assess the same student responses as were assessed by each of the independent Human Graders. The system works based on the same answer key as was provided to the Human Graders. The system, however, determines automatically the range of acceptable answers and appropriate deductions for answers that are at the far ends of the range. The time spent by the system is also calculated in an effort to draw a comparison between the automatic assessment process versus the Human Grader (manual) assessment process.

At the end, the evaluation of the proposed system and its methodology is calculated based on comparisons between the each of the Human Graders and the automatic assessment system to determine the level of agreement among the two assessment methods. Additionally, the

Human Graders are compared to one another to determine the level of agreement between two humans. Results are compared based on the level of agreement, as well as the time spent among each of the grading strategies.

VI. Conclusion

Auto-Assessor is a system that leverages Natural Language Processing tools including WordNet.NET and SharpNLP in order to evaluate student responses to short-answer questions. This system differs from much of the previous work in open question assessment in that it focuses on the linguistic end of the Text Technology Continuum in Natural Language Processing (see Figure 1). The goal of the system was to produce accurate, consistent grades for student responses to open questions by deciphering the semantic meaning of the response. In addition, the system allows for a great deal of latitude when composing the responses rather than requiring specific keywords.

The development of the system used a variation of an Agile methodology. This methodology complemented the component-based architecture. Each component was initially created using the 'just enough' standard supported by the Agile methodology. This allowed for frequent incremental tests of the system in which problems could be identified early. The component-based architecture selected seems appropriate as the ultimate goal of the system would be to support existing Learning Management Systems. As such, only those components required could be 'plugged into' an existing LMS to allow for open-ended question assessment.

This study was developed under very strict constraints. The system in its current format is capable of processing answers containing a single sentence that is free of grammar and spelling mistakes. Future work is encouraged which would allow for multiple sentences to be graded based on their collective meaning. Additionally, future work could incorporate a spell checker and grammar checker. Future work should also allow for a more flexible marking algorithm in which the canonical words could be given varying weight values in the grading scheme depending on their level of importance in the student response.

Acknowledgment

The authors wish to thank the support of Athabasca University, the Mission Critical Research funding, NSERC, iCORE, Xerox and the research related gift funding provided to the Learning Communities Project by Mr. Allan Markin. Additional acknowledgements go out to the Natural Language Processing community. This growing community provides a great deal of support to developers working on leading edge applications of Natural Language Processing. This support takes on many forms such as active forums, robust and well-supported freeware, along with the growing number of Natural Language Processing experts.

References

[1] R. A. Abdul Seoud, A-B.M. Youssef, and Y. M. Kadah, "Extraction of protein interaction information from unstructured text using a link grammar parser," Proceedings International Conference on Computer Engineering & Systems (ICCES 2007), November 27-29, 2007, pp. 70-75.

[2] D. R. Bacon, "Assessing learning outcomes: A comparison of multiple-choice and short-answer questions in a marketing context," Journal of Marketing Education, vol. 25, no. 1, April 2003, pp. 31-36.

[3] D. Bean, "How advances in search combine databases, sentence, diagramming and 'Just the Facts'," IT Professional, vol. 9, no. 1, January/February 2007, pp. 14-19.

[4] J. Burstein, K. Kukich, S. Wolf, C. Lu, M. Chodorow, L. Bradenharder, and M. D. Harris, "Automated scoring using a hybrid feature identification technique," Proceedings Annual Meeting of the Association of Computational Linguistics (ACL 1998), August 10-14, 1998, pp. 206-210.

[5] D. Callear, J. Jerrams-Smith, and V. Soh, "CAA of short non-MCQ answers," Proceedings International Computer Assisted Assessment Conference (CAA 2001), July 2-3, 2001, Loughborough. Retrieved on Feb. 23, 2011, from http://www.caaconference.com/pastConferences/2001/proceedings/k3.pdf

[6] H.-H. Chang, Y.-H. Ko, and J.-P. Hsu, "An event-driven and ontology-based approach for the delivery and information extraction of e-mails," Proceedings International Workshop on Multimedia Software Engineering (MSE 2000), December 11-13, 2000, pp. 103-109.

[7] B.-C. Chien, C.-H. Hu, and M.-Y. Ju, "Intelligent information retrieval applying automatic constructed fuzzy ontology," Proceedings International Conference on Machine Learning and Cybernetics (ICMLC 2007), vol. 4, August 19-22, 2007, pp. 2239-2244.

[8] A. Datar, N. Doddapaneni, S. Khanna, V. Kodali, and A. Yadav, "EGAL - Essay Grading and Analysis Logic," 2004. Retrieved on Feb. 23, 2011, from http://www.d.umn.edu/~tpederse/Courses/CS8761-FALL04/Project/Readme-Boca.html

[9] P. Dessus, B. Lemaire, and A. Vernier, "Free-text assessment in a virtual campus," Proceedings International Conference on Human-Learning Systems (CAPS 2000), December 13-14, 2000, pp. 61-75.

[10] O. Dridi, "Ontology-based information retrieval: Overview and new proposition," Proceedings International Conference on Research Challenges in Information Science (RCIS 2008), June 3-6, 2008, pp. 421-426.

[11] T. Q. Dung and W. Kameyama, "A proposal of ontology-based health care information extraction system: VnHIES," Proceedings International Conference on Research, Innovation and Vision for the Future (RIVF 2007), March 5-9, 2007, pp. 1-7.

[12] P. W. Foltz, D. Laham, and T. K. Landauer, "The intelligent essay assessor: applications to educational technology," Interactive Multimedia Electronic Journal of Computer Enhanced Learning, vol. 1, no. 2, February 1999. Retrieved on Feb. 23, 2011, from http://imej.wfu.edu/articles/1999/2/04/index.asp

[13] S. Ghosh and S. S. Fatima, "Design of an Automatic Essay Grading (AEG) system in Indian context," Proceedings IEEE Region 10 Conference (TENCON 2008), November 19-21, 2008, pp. 1-6.

[14] R. Girju, A. Badulescut, and D. Moldovan, "Automatic discovery of part-whole relations," Computational Linguistics, vol. 32, no. 1, March 2006, pp. 83-135.

[15] M. Hwang, S. Baek, J. Choi, J. Park, and P. Kim, "Grasping related words of unknown word for automatic extension of lexical dictionary," Proceedings International Workshop on Knowledge Discovery and Data Mining (WKDD 2008), January 23-24, 2008, pp. 31-35.

[16] K. Kang, K. Lin, C. Zhou, and F. Guo, "Domain-specific information retrieval based on improved language model," Proceedings International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007), vol. 2, August 24-27, 2007, pp. 374-378.

[17] L. S. Larkey, "Automatic essay grading using text categorization techniques," Proceedings of ACM International Conference on Research and Development in Information Retrieval (SIGIR 1998), August 24-28, 1998, pp. 90-95.

[18] B. Li, J. Lu, J.-M. Yao, and Q.-M. Zhu, "Automated essay scoring using the KNN algorithm," Proceedings International Conference

on Computer Science and Software Engineering, vol. 1, December 12-14, 2008, pp. 735-738.

[19] P. Liao, Y. Liu, and L. Chen, "Hybrid Chinese text chunking," Proceedings IEEE International Conference on Information Reuse and Integration (IRI 2006), September 16-18, 2006, pp. 561-566.

[20] Link Grammar Parser, Computer Software, Abiword, available online at http://www.abisource.com/projects/link-grammar/

[21] M. P. Marcus, M. A. Marcinkiewicz, and B. Santorini, "Building a large annotated corpus of English: The Penn treebank," Computational Linguistics, vol. 19, no. 2, June 1993, pp. 313-330.

[22] T. Mitchell, T. Russell, P. Broomhead, and N. Aldridge, "Towards robust computerised marking of free-text responses," Proceedings Computer Assisted Assessment Conference (CAA 2002), 2002. Retrieved on Feb. 23, 2011, from http://www.caaconference.com/pastConferences/2002/proceedings/Mitchell_t1.pdf

[23] L. Rudner and T. Liang, "Automated essay scoring using Bayes' theorem," Journal of Technology, Learning, and Assessment, vol. 1, no. 2, June 2002. Retrieved on Feb. 23, 2011, from http://escholarship.bc.edu/cgi/viewcontent.cgi?article=1011&context=jtla

[24] K. Sahoo and V. E. Vidyasagar, (2003). "Kannada WordNet -A lexical database," Proceedings IEEE Region 10 Conference (TENCON 2003), vol. 4, October 14-17, 2003, pp. 1352-1356.

[25] K.-S. Shim, C.-Y. Ock, D.-M. Kim, H.-S. Choe, and C.-H. Kim, "Finding similar texts using U-Win," Proceedings International Conference on Advanced Language Processing and Web Information Technology (ALPIT 2008), July 23-25, 2008, pp. 43-48.

[26] E. Sosa, A. Lozano-Tello, and A. E. Prieto, "Semantic comparison of ontologies based on WordNet," Proceedings International Conference on Complex Intelligent and Software Intensive Systems (CISIS 2008), March 4-7, 2008, pp. 899-904.

[27] J. Z. Sukkarieh, S. G. Pulman, and N. Raikes, "Auto-marking: using computational linguistics to score short, free text responses," Proceedings of International Association for Educational Assessment (IAEA 2003), October 5-10, 2003. Retrieved on Feb. 23, 2011, from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.108.7417&rep=rep1&type=pdf

[28] D. Temperley, D. D. K. Sleator, and J. Lafferty, Link Grammar, 2008, available online at http://www.link.cs.cmu.edu/link/index.html

[29] M. Vallez, and R. Pedraza-Jimenez, "Natural language processing in textual information retrieval and related topics," Hipertext.Net, vol. 5, 2007. Retrieved on Feb. 23, 2011, from http://www.upf.edu/hipertextnet/en/numero-5/pln.html

[30] J. Wang, and M. S. Brown, "Automated essay scoring versus human scoring: A correlational study," Contemporary Issues in Technology and Teacher Education, vol. 8, no. 4, 2008. Retrieved on Feb. 23, 2011, from http://www.citejournal.org/vol8/iss4/languagearts/article1.cfm

[31] H. Wang, L. Yuan, and H. Shao, "Text information extraction based on OWL ontologies," Proceedings International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2008), vol. 4, October 18-20, 2008, pp. 217-222.

[32] P. Wiemer-Hastings, D. Allbritton, and E. Arnott, "RMT: A dialog-based research methods tutor with or without a head," Proceedings International Conference Intelligent Tutoring Systems (ITS 2004), Springer, LNCS 3220, August 30-September 3, 2004, pp. 614-623.

[33] R. Williams and H. Dreher, "Automatically grading essays with MarkIT," Issues in Informing Science and Information Technology, vol. 1, pp. 693-700. Retreived on Feb. 23, 2011, from http://espace.library.curtin.edu.au/R/?func=dbin-jump-full&object_id=20483&local_base=GEN01-ERA02

[34] WordNet.NET, Computer Software, Ebswift, available online at http://opensource.ebswift.com/WordNet.Net/

[35] Q. Zhu and X. Cheng, "The opportunities and challenges of information extraction," Proceedings International Symposium on Intelligent Information Technology Application Workshops (IITAW 2008), December 21-22, 2008, pp. 597-600.

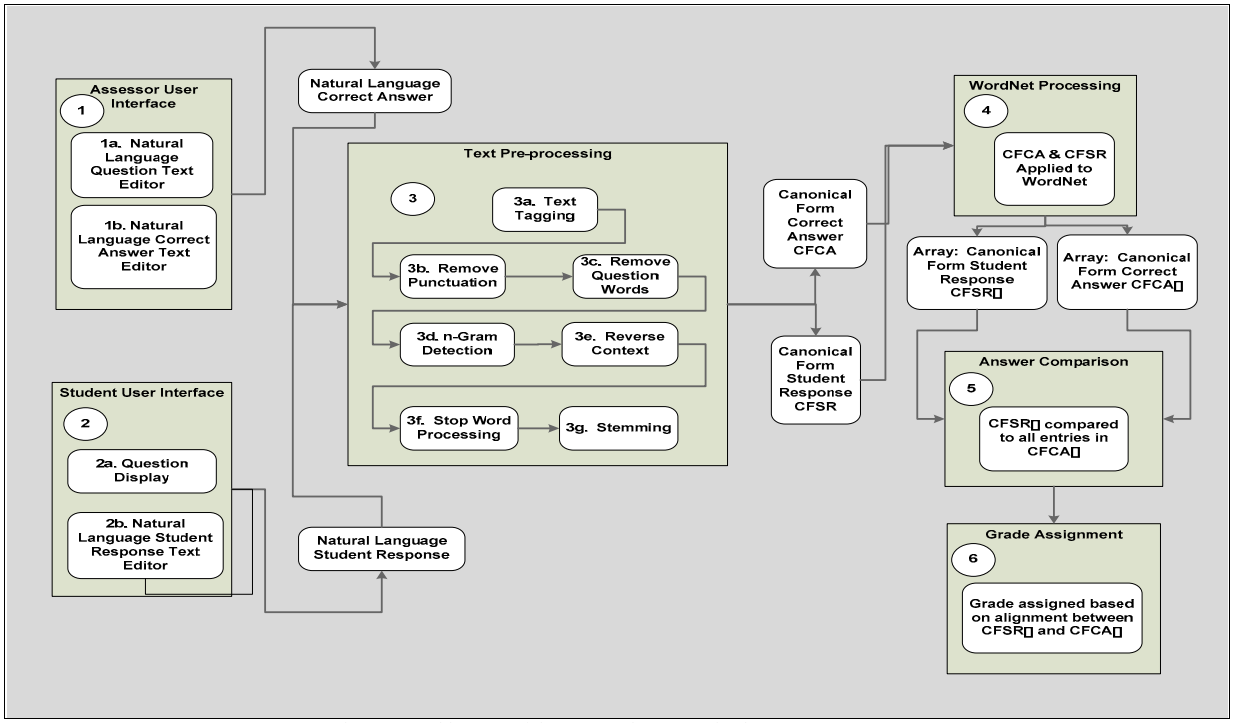

Figure 2. Architecture of the proposed computerized assessment system