نموذج رقم (١)

إقــــــرار

أنا الموقع أدناه مقدم الرسالة التي تحمل العنوان:

An Ontology-Based Automated Scoring System for Short Questions

أقر بأن ما اشتملت عليه هذه الرسالة إنما هو نتاج جهدي الخاص، باستثناء ما تمت الإشارة إليه حيثما ورد، وإن هذه الرسالة ككل أو أي جزء منها لم يقدم من قبل لنيل درجة أو لقب علمي أو بحثي لدى أي مؤسسة تعليمية أو بحثية أخرى.

DECLARATION

The work provided in this thesis, unless otherwise referenced, is the researcher's own work, and has not been submitted elsewhere for any other degree or qualification

Student's name:

اسم الطالبة: **مريم حسن احمد أبو مغصيب**

Signature:

التوقيع:

Date:

التاريخ: ٢٠١٥/١٢/٩

بسم الله الرحمن الرحيم

Islamic University of Gaza
Deanery of Post-Graduation Studies
Information Technology Program

الجامعة الإسلامية – غزة
عمادة الدراسات العليا
برنامج تكنولوجيا المعلومات

An Ontology-Based Automated Scoring System for Short Answer Questions

A Thesis Submitted as Partial Fulfillment of the Requirements for the Degree of Master in Information Technology

Submitted by:
Mariam H. Abu Mugasib

Supervised by:
Dr. Rebhi S. Baraka

December 2015

بسم الله الرحمن الرحيم

الجامعة الإسلامية – غزة
The Islamic University - Gaza

مكتب نائب الرئيس للبحث العلمي والدراسات العليا هاتف داخلي: 1150

الرقم ج س غ/35/ Ref

التاريخ 2015/07/15م Date

نتيجة الحكم على أطروحة ماجستير

بناءً على موافقة شئون البحث العلمي والدراسات العليا بالجامعة الإسلامية بغزة على تشكيل لجنة الحكم على أطروحة الباحثة/ مريم حسن أحمد أبو مغصيب لنيل درجة الماجستير في كلية *تكنولوجيا المعلومات* برنامج تكنولوجيا المعلومات وموضوعها:

نظام تقييم إجابات الأسئلة القصيرة استنادًا إلى الأنتولوجي

An Ontology-Based Automated Scoring System for Short Questions

وبعد المناقشة التي تمت اليوم الأربعاء 28 رمضان 1436هـ، الموافق 2015/07/15م الساعة الثانية مساءً، اجتمعت لجنة الحكم على الأطروحة والمكونة من:

د. ربحي سليمان بركة	مشرفاً و رئيساً	....
أ.د. علاء مصطفى الهليس	مناقشاً داخلياً	....
د. سناء وفا الصايغ	مناقشاً خارجياً	....

وبعد المداولة أوصت اللجنة بمنح الباحثة درجة الماجستير في كلية *تكنولوجيا المعلومات*/ برنامج تكنولوجيا المعلومات.

واللجنة إذ تمنحها هذه الدرجة فإنها توصيها بتقوى الله ولزوم طاعته وأن تسخر علمها في خدمة دينها ووطنها.

والله ولي التوفيق ،،،

مساعد نائب الرئيس للبحث العلمي والدراسات العليا

أ.د. فؤاد علي العاجز

شئون البحث العلمي والدراسات العليا
الجامعة الإسلامية غزة
The Islamic University
Research & Graduate Affairs

Abstract

Short answer questions are open-ended questions that require students to create an answer. They are commonly used in examinations to assess the basic knowledge and understanding and are an important expression of academic achievement. Unfortunately, they are expensive and time consuming to be graded by hand. Therefore, teachers are frequently limited to multiple-choice or true-false standardized tests. In this field, automated scoring systems is developing technology. It is used to overcome time and cost difficulties found in paper passed exams. The search for excellence in machine scoring of short questions is continuing and numerous studies are being conducted to improve the effectiveness and reliability of these systems.

We propose a hybrid approach for measuring the semantic similarity of text, to overcome the problems found in similar systems that adopted single approach only. The proposed approach rely on WordNet ontology for measuring the similarity between two texts. It also uses traditional string matching to get over the shortage of WordNet as an upper ontology. Besides that, the proposed system uses some natural language processing tools such as Parser, Word Segmenter, and Part of Speech Tagger for text preprocessing operations.

The results was modest and still need improvement to make the system scoring as closer as possible to the human specialist scoring.

Keywords: short question grading, question types, automatic grading, electronic evaluation, text relatedness, relatedness measure.

عنوان البحث

طريقة مؤتمتة تعتمد الانتولوجي لتقييم اسئلة الاجابات القصيرة

الملخص

تعتبر الاسئلة ذات الاجابات القصيرة من اهم انواع الاسئلة المستخدمة في التقييم الاكاديمي، و لكن لسوء الحظ فإنها من اعلى انواع الاسئلة تكلفة و استنفاذا للجهد و الوقت اذا ما قُيّمت بطريقة يدوية . لهذا السبب فان معظم المدرسين يحصرون اسئلتهم على اسئلة الاختيار من متعدد او اسئلة الصح و الخطأ. وفي هذا المجال تعتبر انظمة التقييم الالكتروني تكنولوجيا قيد التطوير وهي تهدف الى التغلب على مشكلة الوقت و الجهد التي يعاني منها المدرسين عند تقييم اسئلة الاجابات الحرة القصيرة , ولازال البحث عن نظام تقييم الكتروني فعّال لأسئلة الاجابات القصيرة مستمر و هناك الكثير من الجهود لتحسين كفاءة مثل هذه الانظمة.

هذا البحث يقدم طريقة لقياس مدى التشابه بين النصوص وبالتالي يمكن استخدامه في تقييم الاجابات القصيرة. تعتمد هذه الطريقة على خلط عدة اساليب لقياس التشابه بين النصوص و ذلك لتجاوز القصور الموجود في كل طريقة على حدة .

يستخدم هذا النظام WordNet كمصدر للبيانات لقياس مدى التشابه بين العبارات تزامنا مع استخدام طريقة تطابق النصوص التقليدية المعروفة للتغلب على القصور الموجود في بعض عمليات البحث في WordNet. استخدم النظام ايضا اداة لتحليل النصوص امن اجل تحليل النص الى اجزاء الكلام المعروفة(الفعل , الفاعل , المفعول به.. الخ) و من ثم قياس التشابه بين العبارات حسب اجزاء الكلام التي وردت فيها.

النتائج كانت جيدة، و لكنها لا تزال بحاجة الى التطوير لجعل الدرجات التي يعطيها النظام مقاربة الى حد كبير لتلك التي يعطيها المقيمون المختصون.

الكلمات المفتاحية

التعليم الالكتروني , تقييم اسئلة الاجابات القصيرة , الموودل , انواع الاسئلة , التصحيح الالي , التقييم الالكتروني , تشابه النصوص , قياس مدى التشابه بين النصوص

Acknowledgement

Thanks to Allah for his mercy and help to complete this thesis.

Thanks to my supervisor Dr. Rebhi Baraka for his time, patience, and understanding.

Thanks to my father for his encouragement.

Thanks to my mother for her Prayer and Doaa.

Thanks to the staff who prepare the data set.

Thanks to Mr. Ibrahim H. Abu Shighiba for his advice and assistant.

Thanks to my colleagues and friends for their support.

Mariam H. Abu Mugasib

Table of Contents

Abstract ……………………………………………………………………………..…. I

الملخص.................................…………………………………………………………… II

List of Tables ……………………………………………………………...…………… VI

List of Figures ……………………………………………………………….………... VIII

Chapter 1. Introduction ………………………………………………………..……. 1

1.1. Statement of the Problem ………………………………………………….. 3

1.2. Objectives ………………………………..……………..…………. 3

1.3. Importance of the Research …………………………………………………. 4

1.4. Scope and Limitations of the Research ……………………..………………… 4

1.5. Methodology ………………………………………………………….……… 5

1.6. Thesis Structure ……………………………………..……………………… 6

Chapter 2. State of the Art ……………………………………………………… 7

2.1. Ontology ……………………………………………………………………... 7

2.2. WordNet ……………………………………………………………..……….. 8

2.3. Similarity Measure Approaches ……………………………………..…….. 10

2.4. Semantic Role (SR) ………………………………………………………... 12

2.5. Grammatical Relations (GR) ……………………………………………… 13

Chapter 3. Related Work ……………………………………………………………… 17

Chapter 4. The Automated Short Answers Scoring System ………………………….. 27

4.1. The Approach For Measuring Semantic Similarity Short Answer Texts……… 27

4.2. The Automated Scoring System ……………………………………………… 33

4.3. Implementation of The Scoring System ……………………………............... 38

Chapter 5. Experimental Results and Evaluation ……………………………….... 40

5.1. Experiments ……………………………………………………………….... 40

5.2. Evaluation ………………………………………………………………….. 49

5.3. Discussion ………………………………………………………………….. 50

Chapter 6. Conclusion and Future Work …………………………………………… 52

References ………………………………………………………………….…………… 53

Appendix A………………………………………………………..…………………… 56

Appendix B…………………………………………………...……………………… 58

List of Tables

Table 2.1: Number of Words and Synsets in WordNet ……………………… 8

Table 4.1 : Vector Space For T1 And T2 ……………………..………………. 29

Table 4.2: Sample Data Set …………………………………………………… 39

Table 5.1 : System Results for Question_1 Using Wordnet Ontology ……….. 41

Table 5.2: System Results for Question_1 Using Cosine Similarity ………... 41

Table 5.3: System Average Results (s1, s2) for Question_1…………………. 42

Table 5.4: : System Results for Question_2 Using Wordnet Ontology……… 42

Table 5.5: System Results for Question_2 Using Cosine Similarity ………… 43

Table 5.6: System Average Results (S1, S2) for Question_2 ………………… 44

Table 5.7: :T-test for question 1……………………………………………….. 45

Table 5.7: :T-test for question 1……………………………………………….. 45

Table:5.9 : T-test summary ……………………………………………………. 46

Table 5.10: Accracy for Question_1 To Question_10 ………………………… 50

List of Figures

Figure 2.1: An Example of Wordnet Lexecal Database from Abstract to Specific... 10

Figure 2.2: Semantic Role List ………………………………………………………. 13

Figure 2.3: Difference Between Semantic Roles and Grammatical Relations ……… 14

Figure 2.4: Semantic Roles Example……………………………………………..…. 16

Figure 4.1: Cosine Similariy Formula ………………………………………………. 30

Figure 4.2: Steps for Calculating Cosine Similarity ……………………………… ... 30

Figure 4.3: Word to Word Similarity ……………………………………………... 31

Figure 4.4 : Code for Retrieving Word Synonyms ………………………………… 31

Figure 4.5: Screen Shoot of Synonyms for The Word "Drink" ……………… 31

Figure 4.6: Screen Shoot of Synonyms for The Word "Swallow" …………… 32

Figure 4.7: Screen Shoot of System Results ………………………………………… 34

Figure 4.8: Grammatical Semantic Similarity Method ……………………………… 36

Figure 5.1: System Results for Question_1 ………………………………………... 47

Figure 5.2. : System Results for Question_2 ……………………………………….. 47

Figure 5.3: System Results for Question_9……………………………………….. 48

Figure 5.4: System Results for Question_10……………………………………….. 48

List of Abbreviations

Semantic Role	SR
Information Content	IC
Grammatical Relations	GR

Chapter 1

Introduction

One of the most important aspects of the learning process is the assessment of the knowledge acquired by the learner. In a typical examination (e.g., exam, assignment or quiz), teachers usually create different types of questions, true/false, essay, multiple choice, short answer, numerical, matching are some of questions types.

Short answer questions are directed for answers of one sentence or two. It has the advantage of requiring students to construct an answer for themselves, rather than selecting from a number of predetermined options [22].

Objective questions such as multiple choice or true/false questions are criticized for only being able to assess lower order cognitive skills. For this reason objective tests are often used within an overall assessment strategy that would include free text questions [23].

Free-text questions such as short answers and essay have traditionally been absent from computerized tests because they were considered to be very difficult to mark automatically. With the advent of new technology, such as advances in the field of natural language processing and information extraction, it is possible to include certain types of free- text such as short answer questions in computerized tests [47].

Grading in education is the process of applying standardized measurements of varying levels of achievement for a student in a course [36].

Of the benefits of automating marking include time and cost savings, and the reduction in errors and unfairness due to human bias, exhaustion or lack of consistency [1].

This research concentrates on short question type; this is because short answer questions suffer from less attention in the field of automatic grading. An automated grading system is proposed. It will read student's answer, evaluate it and returns an approximate grade based on a prior assessment rules.

For such system, to give accurate score for the student answer, semantic similarity is considered during the process of evaluation. This means when the system starts assessing what the student writes, it is not enough to search the inputted answer for the existence of specific words that the teacher provided in advance (string matching only). However, we have to put in mind all words and sentences that carry the same meaning regardless of the sentence structure.

Semantic similarity can be measured at different levels, ranging from word and sentence to paragraphs and documents. We focus on quantifying semantic similarity at the sentence or paragraph level, i.e. compute the semantic similarity between two given paragraphs (student answer and teacher reference answer).

To measure or quantify the semantic similarity between two texts; several approaches can be used, some of these approaches are ontology-based measures, the others are either information content (corpus) based measures or feature based measures.

Each of the preceding measure approaches has its own disadvantages. Ontology based measures approach depends on the ontology construction and its accuracy is tied to how well the ontology is designed. Furthermore it assumes that the relation connects different concepts is only "is-a" type relation while actually many other relations can connect between concepts.

In the information content approach a probability is associated to each concept in the ontology, these probabilities are based on the word occurrences in a given corpus. So, the short length of some text segments do not provide enough context for the information content measure to be effective [24].

We are proposing a system that uses ontology-based and other similarity measure approach to overcome the weakness of each approach alone. In the context of the research, the role of the ontology is to provide the system with definition of vocabularies to use them in the semantic similarity measure between the concepts that appears in teacher answer and those appear in student answer.

The system supposed to be a helpful tool for teachers as it will free them from the exhausting manual grading and automatically score essays accurately.

1.1. Statement of the Problem

Teachers can prepare quizzes with different types of questions such as true/false, essay, matching, multiple choice and short-answer. All these types can be graded automatically except essay and short answer questions; they still need manual assessment, which is time consuming and slow releasing of students records process. Therefore, until the moment of writing this thesis the essay and short answer questions are the most difficult and time consuming question type for grading. Because of this, developing a system that can make accurate assessment for these types of questions is an argent need.

1.2. Objectives

1.2.1. Main Objectives

The main objective of this research is to build an ontology-based grading system especially designed for short answer questions. This system is supposed to be capable of assessing students' answers with a mark approximately near that given manually by the teacher.

1.2.2. Specific Objectives

The specific objectives of this research are:

1. Analyze similar systems used for automatic grading to determine the most frequent approach used for grading.
2. Select the best semantic similarity approach that exists so far through carrying out experiments.
3. Determine the best ontology to be used as information source (WordNet, Idilia or other domain specific ontology) to get the best result.
4. Implement a system prototype based on the proposed approach.
5. Evaluate the system's accuracy based on a chosen evaluation strategy.

1.3. Importance of the Research

Several factors have contributed to a growing interest in automated grading among them are time, cost, accountability, standards, and technology.

- Students need to receive feedback in order to increase their knowledge. However, responding to student answers can be a burden for teachers, particularly if they have large number of students and if they frequently assign writing assignments, providing individual feedback to student answers might be quite time consuming. Automated Grading systems can be very useful as they can provide the student with a score as well as feedback within seconds [1].
- Some educational institutes and universities used to pay for teaching assistants to correct exams or assignment, this usually occur when the number of students are very large, Automated Grading systems can free us from this unnecessary cost .
- Scoring short answer questions has traditionally relied on human raters, who pass through different modes (happiness, anger, and stress). These modes affect the way they assess students' answers by somehow. In other cases, the rater’s impression from one characteristic of the answer is generalized to the answer as a whole, and this will surly affect the overall assessment of the answer. However, using Automated Grading systems release assessment from these human factors. In other words, we can say that Automated Grading systems standardize the methodology of assessment.
- The advance in information technology especially in the field of semantics and NLP promises to measure educational achievement with high accuracy, so, it is important to utilize this advance.

1.4. Scope and Limitations of the Research

- This research assumes the existence of a reference answer to compare the student answer with; therefor it is not suitable for grading essay in general (without reference answer).
- The research will adopt upper domain ontology, which makes it a general system, and not specific to a certain domain, this option is justified since the system performs words to word similarity for predetermined domain specific keywords provided by

the teacher when evaluating the question. These keywords can partially replace the domain ontology.

- The system do not supports Arabic language and deals with English.
- The system assumes that the answer text is written via computer not and hand written.

1.5. Methodology

To accomplish the objectives of the research, the following methodology phases has been accomplished:

- *Analyzing similar systems phase*

  In this phase, a number of automated scoring systems were studied and analyzed carefully to identify their methodology in marking.

- *Information source selection phase*

  In this phase, we decided which ontology to use, upper ontology or domain ontology. The choice was to use WordNet (upper domain ontology) because it is freely available and contains a huge number of vocabularies and their meaning (about 155287 term) [43].

- *Data set preparation phase*

  In this phase a collection of questions with their correct answers along with students' answer samples were prepared. The question bank prepared by Prof.Aly Aly Fahmy and Eng. Wael Hassan Gomaa is considered [25].

- *System development phase*

  In this phase, we develop a prototype of the proposed approach using one of supported development language, the development includes the following:

  - Specify the requirements of the system.
  - Retrieve data concepts and their relations using a suitable API functions.
  - Specify a scoring scale to evaluate the answers according to it.

- *System evaluation phase*

In this phase, we evaluate the implementation and verify that it achieves acceptable accuracy. The verification was done by referring to professional teachers and compare their judgment with the system result.

Our methodology in measuring the similarity between two given texts consists of two methods and then combine these methods together to get the overall score.
The first method measures the similarity between two texts through measuring the similarity between their component words (word to word similarity.

The other method measures the similarity between two texts through determining to how far their semantic roles are similar (grammatical semantic role similarity).

1.6. Thesis Structure

The thesis is divided into 6 main chapters. Chapter 1: Introduction, Chapter 2: State of the Art, Chapter 3: Related Works, Chapter 4: The Automated short Answers Scoring System**,** Chapter 5: Experimental Results and Evaluation, Chapter 6: Conclusion and Future Work.

Chapter 2

State of the Art

This chapter presents the background of a group of theoretical concepts that were mentioned and adopted in the research. These concepts are Automatic scoring systems, ontology, WordNet, similarity measure approaches, semantic roles and grammatical relations.

2.1. Automatic scoring systems

Computers usually grade questions' answers by simply matching them to a key answer. The system assigns a grade to the student answer based on its similarity to a model answer provided by the instructor.

Systems for automating the assessment of textual answers have been available since the mid 1990’s and some progress has been made in their application to assessing short answer questions. However, progress in the field is delayed by a lack of qualitative information regarding the effectiveness of such systems.

Short answer marking engines work best with questions producing convergent answers, that is, where there are a limited set of answers that the examiner is looking for, and they do not cope well with questions where there is an unpredictable range of acceptable answers[xxx1].

Various techniques such as Ontology, Semantic similarity matching and Statistical methods are used to these systems.

2.2. Ontology

In the context of knowledge engineering, the definition of ontology is rather confusing because ontologies can be explained from three different aspects: the content of an ontology, the form of an ontology and the purpose of an ontology, however we will give a simple definition of ontology. Ontology is a finite list of terms and Relationships between these terms. Many reasons can stand behind the development of ontologies, some of them are to share common understanding of the structure of information among people or software agents and enabling reuse of domain knowledge [38].

Sharing common understanding of the structure of information among people or software agents is one of the more common goals in developing ontologies. For example, suppose several different Web sites contain medical information or provide medical e-commerce services. If these Web sites share and publish the same underlying ontology of the terms they all use, then computer agents can extract and aggregate information from these different sites. The agents can use this aggregated information to answer user queries or as input data to other applications [39].

Regarding the methodology for developing ontologies, it is important to say that there is no one correct way or methodology for developing ontologies. But, there is general issues to consider when start developing an ontology [38]. For example, it is good to use iterative approach by starting with a rough first pass at the ontology, then revise and refine the progressing ontology and fill in the details. Another consideration is trying to make concepts as close as possible to objects and relationships in the domain of interest. There are many methodologies for building an ontology, all of them share the following common steps

Step 1. Determine the domain and scope of the ontology

Step 2. Consider reusing existing ontologies

Step 3. Enumerate important terms in the ontology

Step 4 Define the classes and the class hierarchy

Step 5. Define the properties of classes—slots

Step 6. Define the facets of the slots (value type, allowed values, the number of the values (cardinality), and other features of the values the slot can take)

Step 7. Create instances

2.2. WordNet

WordNet is a lexical database for the English language[43]. It groups English words into sets and provides short definitions and usage examples. WordNet can thus be seen as a dictionary. While it is accessible to human users via a web browser, its primary use is in automatic text analysis and artificial intelligence applications [43]. Table 2.1 shows the number of words and synsets in wordnet.

Table 2.1: Number of Words and Synsets in WordNet

POS	Unique Strings	Synsets
Noun	117798	82115
Verb	11529	13767
Adjective	21479	18156
Adverb	4481	3621
Totals	155287	117659

WordNet was designed to establish the connections between four types of Parts of Speech (POS) - noun, verb, adjective, and adverb. The smallest unit in a WordNet is synset, which represents a specific meaning of a word. It includes the word, its explanation, and its synonyms. The specific meaning of one word under one type of POS is called a sense. Each sense of a word is in a different synset. Each synset has a gloss that defines the concept it represents.

For example, the words night, nighttime, and dark constitute a single synset that has the following gloss: the time after sunset and before sunrise while it is dark outside. Synsets are connected to one another through explicit semantic relations. Some of these relations (hypernym, hyponym for nouns, and hypernym and troponym for verbs). For example, tree is a kind of plant, tree is a hyponym of plant, and plant is a hypernym of tree. WordNet organizes them in the order of the most frequently used to the least frequently used [44].

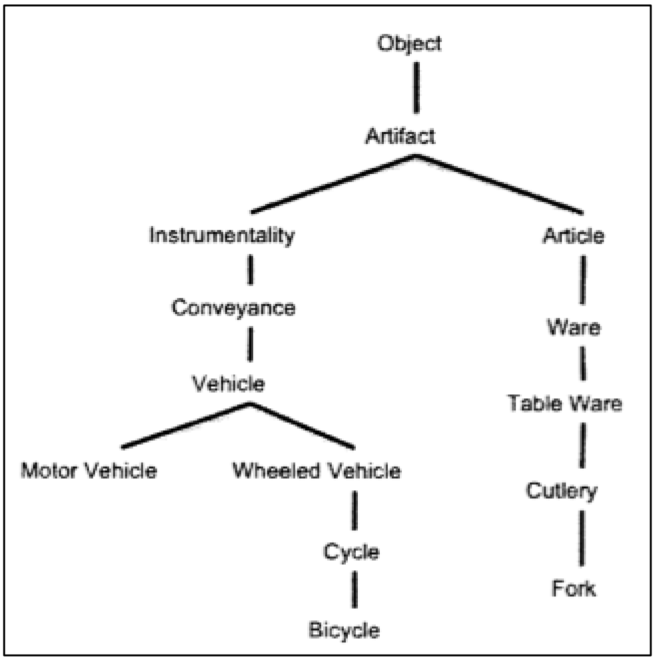


Figure 2.1: An Example of Wordnet Lexecal Database from Most Abstract to Most Specific

2.3. Similarity Measure Approaches

Semantic similarity measure is the process of comparing two different objects to determine how well they agree or match with each other. It is used to quantify the common essential features shared by two concepts and it plays a key role in information Retrieval and integration and other applications involving comparison between concepts. Following is a brief description for the most common similarity measure approaches.

- **Path Distance Measure:** It is based on the ontology structure, and assumes that there is "is-a" relation that connect different concepts, computation of similarity is done in terms of the shortest path between the target synsets (the groups that each concept belongs to). The degree of similarity corresponds inversely with the path length (if the path is short then the similarity degree is strong and vice versa) [24].
- **Depth Relative Measures Approach***:* The depth relative approach is the shortest path approach, but it consider the depth of the edges connecting the two concepts in the overall structure of the ontology to quantify similarity. It calculates the depth from the root of the taxonomy to the target concept [24].

- **Information Content (IC) Based Measures (Corpus):** This approach associate probabilities to each concept in the ontology. These probabilities are based on the word occurrences in a given corpus. The IC value of the root concept (the most abstract concept) is 0 and the IC value of a leaf concept is 1. Hence, the information content values of the intermediate concepts in the taxonomy range from 0 to 1. Resnik Measure, Lin Measure, Jiang and Conrath measure are all purely based on corpus statistics and use the IC as a basis for computing similarity between concepts [24].
- **Information Extraction Approach:** Information extraction is a natural language processing technique that turns the unstructured information embedded in text into structured data. Most information extraction approaches manually constructed patterns, which if matched, indicate that some text is similar to another or in our thesis context the question has been answered correctly. Some information extraction approaches uses information extraction technique to extract significant features from answers and match them with hand-made patterns. These approaches require skill and familiarity with domain [31].
- **Machine Learning Approach:** As mentioned before, information extractions approach requires manually crafted patterns, and constructing these patterns is a laborious process. To save time and labor, researcher have investigates machine learning techniques like Nearest Neighbor Classification techniques to learn information extraction patterns and automate this process though annotating corpus and indicating which sentences in the text contains the relevant information for a particular pattern then more patterns can be learned by bootstrapping those annotating patterns [31].
- **Hybrid Measures:** It is the combination of the above-mentioned methods with the available knowledge resources. The major advantage of this approach is if the knowledge of an information source is insufficient then it may be derived from alternate information sources. Hence the quality of similarity assessment would be improved [24].

Almost all of the approaches mentioned above has their own weaknesses, some of them, e.g. information extraction require the grader to exert a hard effort to preprocess the text and construct the answer patterns, and in most cases it needs an expert in the domain of the exam and also in computational linguistics.

Others, like Information Content based (Corpus based), measures the similarity according to the co-occurrence words in the text, so they work well for long text because they have sufficient ad adequate information for computational methods operations. In short text cases they found to be less effective.

for machine learning approach, some studies that make comparisons between several machine learning technique such as decision tree, Bayesian and the information extraction techniques concluded that machine learning methods are not accurate enough to replace hand crafted pattern matching approach [31].

2.4. Semantic Role (SR)

A semantic role is a task in natural language processing consisting of the detection of the semantic arguments associated with the verb of a sentence and their classification into their specific roles. For example, given a sentence like "Mary sold the book to John", the task would be to recognize the verb "to sell", "Mary" as representing the seller (agent), "the book" as representing the goods (theme), and "John" as representing the recipient. This is an important step towards making sense of the meaning of a sentence [40].

The goal of semantic role theories is to obtain a set of semantic roles that can apply to any argument of any verb. Their function is to make possible the unique identification of the arguments of the verb.

There have been a large number of proposals with regard to the number and nature of the list of semantic roles needed. Below is one of these lists

Agent: The ‘doer’ of the action denoted by the predicate.

Patient: The ‘undergoer’ of the action or event denoted by the predicate.

Theme: The entity that is moved by the action or event denoted by the predicate.

Experiencer: The living entity that experiences the action denoted by the predicate.

Goal: The location or entity in the direction of which something moves.

Benefactive: The entity that benefits from the action or event denoted by the predicate.

Source: The location or entity from which something moves

Instrument: The medium by which the action is carried out.

Locative: The specification of the place where the action denoted by the predicate in situated.

Figure 2.2: Semantic Role List[36]

2.5. Grammatical Relations (GR)

Grammatical Relations (GRs) are relations between words in sentences. These are much more clearly defined than semantic roles. GRs vary from one language to another. Therefore there are no universal definition of grammatical relations. The given definition here is applied only to English.

Subject. The subject is the nominal element (noun, noun phrase or pronoun) that the verb agrees with. It comes right before the verb in a clause, and when pronominalized, employs subjective pronouns (I, she, we, they, etc.). "Omar" is the subject in each of the following sentences

- Omar ate all the apples.
- Omar heard a train coming.
- Omar is tall.

(Direct) object: A nominal element that comes right after the verb in a sentence, and is not preceded by a preposition. Omar is the direct object in each of the following sentences.

- Ahmed saw Omar.
- Ahmed kicked Omar.
- Ahmed sent Omar a letter.

(Indirect) object: A nominal element preceded by "to" or "for" that can be paraphrased as a direct object. Omar is the indirect object in each of the following sentences.

- Ahmed sent a letter to Omar.
- Ahmed made a sandwich for Omar.

It is possible that a single word appears in two sentences with the same semantic role, but different grammatical relations. Consider the following two sentences.

Luci sent a letter to Omar.

Luci sent Omar a letter.

The scene referred to is the same for both sentences, therefore the semantic roles do not change. However, there is a grammatical difference between them. In the first sentence Omar appears at the end of the clause, and is preceded by "to". Therefore it is grammatically defined as indirect object (see above).

In the second sentence the word Omar appears right after the verb, and does not follow a preposition. Therefore it is grammatically defined as a direct object (see above) [45].

The following example illustrate the difference between SR and GRs. Lucretia is considered as agent in SR and as subject in GR, the room is considered as location in SR and as an object in GR.

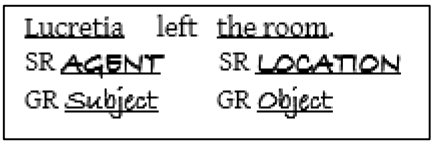


Figure 2.3: Difference Between Semantic Roles and Grammatical Relations [41]

When talking about semantic role tools there are a number of them available such as, Meta tools, BioKIT which is dedicated for biomedical text, SEMAFOR and SENNA and others, following is a brief introduction to some of these

- **Meta Tools**

This tools provide part-of-speech tagging, dependency parsing, and semantic role labeling of a sentence. The system has two main components are dependency parser and semantic role labeler. The tools are language independent, provide a high accuracy. The dependency parser had the top score for German and English languages [41].

- **SEMAFOR**

This is a tool for automatic analysis of the frame-semantic structure of English text. It uses FrameNet which is a lexical resource that groups concepts as "frames". Each frame in the lexicon defines several "roles" corresponding to parts of that concept.

This tool attempts to find which words in text evoke which semantic frames, and to find and label each frame's arguments. It takes as input a file with English sentences, one per line, and produces an XML file containing the text of the input sentences, augmented with the frame-semantic information.

- **SENNA**

SENNA is a software distributed under a non-commercial license, and provides Natural Language Processing (NLP) services like: part-of-speech (POS) tags, semantic role labeling (SRL) and syntactic parsing.
SENNA is accurate and fast because it uses a simple architecture, self-contained as it does not rely on the output of existing NLP system.
SENNA is written in ANSI C, with about 3500 lines of code. It requires about 2002MB of RAM and should run on any computer [42].

In this research we choose to use SENNA because it is open source and fast. It comes with a binary for Windows OS and can be called from java programs (the programming language we use in implementation)

Regardless of the tool used in semantic role labeling, if we try to label the sentence "He wouldn't accept anything of value from those he was writing about", the output will be like the following.

[A0 He] [AM-MOD would] [AM-NEG n't] [V **accept**] [A1 anything of value] from [A2 those he was writing about] .

Figure 2.4: Semantic Roles Example

Where V, A0, A1, A2 refer to verb, subject, object, second object as illustrated in the table 2.2.

Table 2.2.: Roles for the Accept Fram Shown In Figure2.4

V	Verb
A0	Acceptor –subject
A1	thing accepted- object
A2	Accepted from- second object

Chapter 3.

Related Work

There are a number of approaches that have been proposed in the past for automatic short answer grading. This chapter presents works related to automated short answers grading systems that have been developed by researchers and companies either as commercial products or as research prototypes.

Before start discovering the related works, it is appropriate to mention that each of them belongs to one of the similarity measure approaches mentioned in Section 2.3. In the following pages we are considering a package of studies that forms a preface and contributed in understanding how the automated systems works.

- **Clustering Approach to Amplify Human Effort for Short Answer Grading**

  In [29] Sumit Basu introduce an approach to grade short answer questions. The approach first train a similarity metric between student responses, then, use this metric to group responses into clusters and sub-clusters. This make teachers able to grade multiple answers as package. This amplification is referred to as “power-grading.

  Another important point in this study is trying to grade answers in a partially automatic way, it attempts to mix the abilities of both the human and the machine. In particular, it does not classify individual answers as being right or wrong, instead, the automatic part of the approach finds groupings and subgroupings of similar answers from a large set of answers to the same question, and then the human or manual part depends on the teacher to apply his mark to the groups. This means that the teacher can mark the answers as right or wrong and give feedback to a whole group at the same time once.
  The power-grading approach seems good for the teachers as it ease the assessment process them, but it seems not for the students because although there is a little variation between the answers it gives the same mark for all the answers in the same group and deals with them as one block.

To evaluate the benefit of the approach, the researcher examine how far a grader can get with a given amount of effort. Throughout the two contexts, “grading on a budget,” and the context of “effort left for perfection,” which means maximizing the progress from a fixed number of actions and the number of additional user actions required to grade all items correctly respectively, Under these criteria, the results find that using clusters formed via the learned similarity metric leads to substantially better results than using those formed via individually classifying items.

- **Arabic Short Answer Scoring with Effective Feedback for Students**

  In [25] Fahmy and Gomaa compare a number of string-based and corpus-based similarity measures and then evaluate the effect of combining these measures. They run experiments over fourteen String-Based and two Corpus-Based similarity algorithms through two models. The first model (Holistic Model) measures the similarity between the complete form of student answer and model answer without dividing the student answer. The second model (Partitioning Model) automatically divides student answer into set of sentences using sentences boundary detection templates based on regular expression, then it maps each sentence to the highest similarity element of model answers [25]. First, the similarity between the student and model answers is measured using the text similarity measures. Second, the obtained similarity values (0-1) are scaled onto the original scale (0-10).

  By applying String-based measures to map each sentence in student answer to each element in model answer the elapsed time was reduced to the sixth which is considered real achievement. From the other side, the combination paved the way to multithreading approach which accordingly decreased the elapsed time.

  The results showed that applying stop word removing task separately or merged with the stemming task is better than applying the stemming task separately. Also partitioning model achieved better results than holistic model in all cases.

- **On the Automated Assessment of Short Free-Text Responses**

Raheel Siddiqi and Christopher J. Harrison evaluate C-rater to identify its capabilities and limitations. They call to create a common repository of standardized data sets and made it available to researchers and system developers. The creation of this standardized data sets will help quantifying the progress in the field [47].

- **Scaling Short-answer Grading by Combining Peer Assessment with Algorithmic Scoring**

Chinmay Kulkarni et.al tries to integrate peer and machine grading to preserve the robustness of peer assessment and lower grading burden. Before peer assessment begins, a machine-learning algorithm predicts the grade for each answer, to do this they built text classifier with the predicted grade as the output Teaching assistants provided numeric scores and correct/incorrect attributes for about 500 student responses per question. The numeric grades were used as labels to train the classifier [26].

The classifier outputs the most likely grade (the prediction), as well as the probabilities of all possible grades (e.g., an answer may have a grade of 1 with probability of 0.2, and a grade of 0 with probability 0.8).

For the rest of the grading process, they use the probability of the most likely grade (in the example 0.8) as the algorithm's confidence in the grade. The algorithm's confidence determines the initial number of peer raters assigned to each answer. Next the peers identify correct/incorrect attributes in student answers independently.

Staff associated a score with the presence of each attribute. Finally, other peers verify whether these feature labels were accurately applied.

This approach adjusts the number of peers needed to evaluate an answer based on algorithmic confidence and peer agreement. i.e. its main objective is to predict the number of the peers raters needed not the score itself. This may cost a lot of peers when the system is in a false positive case (too many raters than needed).

- **TakeLab: Systems for Measuring Semantic Text Similarity**

  ˇSari et al developed a system for determining the semantic similarity of short texts. They predict the human ratings of sentence similarity using supervised machine learning support vector regression model with multiple features measuring, word-overlap similarity and syntax similarity. The system was submitted to the SemEval 2012 Task 6, and out of 89 systems submitted, it was ranked in the top 5 [2].

- **Measuring Semantic Similarity in Short Texts through Greedy Pairing and Word Semantics**

  Mihai Lintean and Vasile Rus propose a greedy method to the problem of measuring semantic similarity between short texts. Their method is based on the principle of compositionality, which states that the overall meaning of a sentence can be captured by summing up the meaning of its parts, i.e. the meanings of words. Based on this principle, they extend word- to-word semantic similarity metrics to quantify the semantic similarity at sentence level [3].

- **Similarity Measures for Short Segments of Text**

  Donald Metzler et.al study the problem of data sparseness and the lack of context in short segments of text from an information retrieval perspective, focusing on text representations and similarity measures [5]. In their work, they describe a set of similarity measures that can be used to tackle the problem. These measures include simple lexical matching, stemming, and text representations using web search results. They showed how web search results can be used to form expanded representations of short text segments to overcome the data sparseness problem. Their evaluation of the measures depend on query-query similarity task using a collection of 363,822 popular web queries.

- ***C-rater***

  C-rater [47] is one of the automated short-answer scoring systems produces by Education Testing Service (ETS). Its approach in marking student answers is to create a model of the correct answer and then map the student's answer on to this model.

  It generates tuples, for each sentence of the student's response. A tuple consists of verb in each clause of a sentence together with its arguments (such as subject and object).

  But first the students' responses have to be normalized to detect to which each pronoun refers (pronoun references), and also to detect Syntactic variation use of synonyms(e.g. decrease, lessen, minimize) and to identify morphological variations (e.g. hide, hides, hided, hidden)[47].

  After a student's response has been converted to a normalized representation, it is then compared with the model answer.

  C-rater uses a word similarity matrix for this purpose. The word similarity matrix has entries for a very large number of English words and with each word there is an associated list of similar word items.

  When a student's response is evaluated, C-rater tries to match each base form in the student's response with the base forms of the model answer and all the associated similar word lists. If a match is found, then the base form in the response is replaced with the word in the model answer.

  To evaluate C-rater it was used to assess students' response in the National Assessment of Educational Progress (NAEP) Math Online Project, the average student response was around 15 words. Each student response was scored by C-rater and by two human judges.

  The agreement percentage between C-rater and the first human judge was 84.4% and between C-rater and the second human judge 83.6%. This means that C-rater's performance was excellent in the case of the NAEP assessment.

Limitations of C-rater are that it was unable to recognize some correct concept in a response and assigns too much credit for a response than it deserve.

- **PEG (Project Essay Grade)**

  Project Essay Grader (PEG) was developed by Ellis Page upon the request of the College Board, which wanted to make the large-scale essay scoring process more practical and effective. Page uses variables such as fluency, diction, grammar, punctuation, etc. to generate a score.

  The scoring methodology of PEG is divided into two stages, a training stage and a scoring stage. PEG is trained on a sample of essays in the training stage, in the scoring stage there are qualities of the writing style that need to be measured, these qualities called trins.

  PEG uses approximations of these variables, called proxes, to measure these trins. Specific attributes of writing style, such as average word length, number of semicolons, and word count are examples of proxes that can be measured directly by PEG to generate a grade [37].

  For a given sample of training essays, human raters grade these essays, and determine values for up to 30 proxes. The grades are then entered as the criterion variable in a regression equation with all of the proxes as predictors, then, coefficients are computed for each predictor. For the remaining unscored essays, the values of the proxes are found, and those values are used to calculate a score for the unscored essay [37].

  Of the strength points of PEG is that the predicted scores are close to those of human raters. Furthermore, the system can computationally track the writing errors made by the users. However, PEG has been criticized for ignoring the semantic aspect of essays and focusing more on the surface structures.

- **Intelligent Essay Assessor**

  Intelligent Essay Assessor (IEA) is an essay grading software produced by the Pearson Knowledge Analysis Technologies. It analyzes and scores an essay using a semantic text-analysis method called Latent Semantic Analysis (LSA).

  IEA focus more on the content features rather than the form ones (grammar and punctuation); however, this does not mean that IEA provides no feedback on formal features in an essay. In other words, even though the system uses an LSA-based approach to evaluate the quality of the content of an essay, it also includes scoring and feedback on grammar, style and punctuation [20].

To be trained on a set of domain texts, it uses a pre-scored essays of other students, expert model essays as data source materials in order to measure the overall quality of an essay.
This approach allows IEA to compare each essay with similar texts in terms of the content quality. First, IEA compares content similarity between a student's essay and other essays on the same topic scored by human raters to determine how closely they match. It then predicts the overall score.

- **E-Rater**

E-Rater was initially used for scoring the Graduate Management Admissions Test (GMAT). It uses the NLP tool for parsing all sentences in the essay and uses a combination of NLP techniques to extract features from the essays to be graded.

Essays are evaluated against a set of human graded essays, an essay that stays on the topic of the question and displays a variety of word use and syntactic structure will receive a score at the higher end of a six point scale.

E-Rater adopts a corpus-based approach by using actual essay data to analyze the features of essay samples.
The application is designed to identify features in the text that reflect writing qualities specified in human reader scoring criteria and is currently composed by a set of modules. One of these modules identify features that may be used as scoring guide criteria for the syntactic variety, the other module identify the organization of ideas and the vocabulary usage of an essay. Finally, there is a module used to compute the final score. E-Rater is currently embedded in criterion a web-based real-time version of the system developed by ETS Technologies.

A feedback component with advisory features has been added to the system. The advisories are completely independent from the generated score. E-Rater is trained on a collection of 270 essays that have been manually scored by trained human raters. It is far more complex and requires more training than many other available systems. Over 750000 GMAT essays have been scored, with agreement rates between human expert and system consistently above 97% [21].

- **IntelliMetric**

IntelliMetric is an AES system developed by Vantage Learning, and is the first essay-scoring tool that was based on artificial intelligence.

Like e-rater, IntelliMetric relies on NLP and needs to be trained with a set of pre-scored essays with known scores assigned by human raters. These essays are then used to extract the scoring scale.

The system has multiple steps to analyze essays. First, the system infers the essay features associated with each score. The second step includes testing the scoring model against a smaller set of essays with known scores for validation purposes. Finally, once the model scores the essays as desired, it is applied to new essays with unknown scores.

There are key principles underlying the IntelliMetric system. First, IntelliMetric is considered to be a learning engine that obtains the necessary information by learning ways to examine the sample pre-scored essays by expert raters. Second, its error reduction function' allows it to increase its accuracy over time by detecting and learning from its mistakes. Finally, one of the best attributes of IntelliMetric is that it is capable of evaluating essay responses in multiple languages including English, Spanish, Hebrew, Dutch, French, Portuguese, German, Italian, Arabic, and Japanese [20].

- **Betsy (Bayesian Essay Test Scoring System)**

Betsy is an essay scoring system that was developed as a research tool that includes the best features of PEG, LSA, and e-rater along with its own essential characteristics.

It is used to classify text based on trained material, in addition, it can be applied to short essays in various content areas, the goal of this system is to determine the most likely classification of an essay into a four point scale (extensive, essential, partial, unsatisfactory) using a large set of features including both content and style features.

The models used to classify text are the Multivariate Bernoulli Model (MBM) and the Bernoulli Model (BM). With the MBM each essay is viewed as a special case of all features, and the probability of each score for an essay is computed as the probabilities product of the essay features. With the BM the conditional probability of presence of each feature is estimated by the proportion of essays within each category that contain the feature.

This model is criticized because it requires a long time to compute since every term in the vocabulary needs to be examined. According to its authors, Betsy relies on an approach that may incorporate the best features of PEG, LSA and E-rater plus it has several advantages of its own. It can be employed on short essays, it is simple to implement, can be applied to a wide range of content areas.

Summary

From the previous studies we can see that they adapt either the content-based approach, information extraction approach or machine learning approach. For details of these approaches see Section 2.3, each of these approaches has shortage and limitation, information extraction for example requires the grader to exert a hard effort to preprocess the text and construct the answer patterns, and in most cases it needs an expert in the domain of the exam and also in computational linguistics.

Information Content based approach is criticized because it measures the similarity according to the co-occurrence words in the text, so they work well for long text because they have sufficient ad adequate information for computational methods operations. In short text cases they found to be less effective.

For machine learning approach, some studies that make comparisons between several machine learning technique such as decision tree, Bayesian and the information extraction techniques concluded that machine learning methods are not accurate enough to replace hand crafted pattern matching approach

Little studies such as "Text-to-text Semantic Similarity for Automatic Short Answer Grading" by Michael Mohler and Rada Mihalcea , "An Automated Grader for Short Answer Responses" by Sami Saqer and "Automatic Short Answer Grading System " by 1P.Selvi depends on the hybrid approach. This gave us a hit to use the hybrid approach in our thesis and compare it with the result of each approach alone.

Chapter 4.

The Automated Short Answers Scoring System

The problem of short answer grading can simply be viewed as text similarity task, two text are similar if they carry out the same information regardless of the word order. In this chapter we will give an overview of our scoring system, its approach and structure. It explains the methods and lists the tools and programs used to implement the model. Beside that it gives an overview about the data set used in the evaluation.

4.1. The Approach for Measuring Semantic Similarity Short Answer Texts

Our system measures the similarity between two given texts by applying two methods and then combine them together to get the overall score.

The first method measures the similarity between two texts through determining to how far their component words are similar in meaning (word to word similarity), this approach is not enough to give acceptable accuracy because even if two sentences consist of the same words, they may have different meanings.

For example, the sentence "Omar bought the car from Ahmed" consists of the same words as the sentence "Ahmed bought the car from Omar". However, they are completely different in meaning. In the first, Ahmed is the seller and Omar is the one who gave the money to him, but in the last sentence, Omar is the seller and Ahmed is the one who gave the money to him.

To overcome this obstacle, the first method was supported with another one that measures the similarity between two texts through determining to how far their semantic roles are similar (grammatical semantic role similarity).

The following two Sections (4.1.1 and 4.1.2) explain these two methods

4.1.1. Word to Word Similarity

Sentences are made up of words, so it is reasonable to represent a sentence using the words in it. To measure the similarity between two sentences, we calculate the similarity between their components words using two ways, first using cosine similarity and second using enhanced traditional string matching through WordNet ontology

First: calculating word to word similarity using cosine similarity

This method computes the similarity between two texts through computing the cosine, i.e. normalized dot product between their corresponding vectors.

Cosine similarity is a very popular and mathematically technique to derive the semantic similarity based on analyzing word-to-word co-occurrence in a collections of texts. Its advantage is that a similarity measure can be computed between any two words or sentences that are being found in the analyzed texts.

Using Cosine similarity

makes it possible to compute similarity measures for adjectives and adverbs too, not only for nouns and verbs, as in the WordNet-based metrics.

Cosine similarity is exactly the angular difference between two vectors. It can be expressed by the formula shown in Equation (4.1)

$$\text{similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}}$$

Equation 4.1: Cosine Similariy Formula

To calculate the cosine similarity, the steps shown below have to be followed. .

- Take the dot product of vectors A and B.
- Calculate the magnitude of Vector A.
- Calculate the magnitude of Vector B.
- Multiple the magnitudes of A and B.
- Divide the dot product of A and B by the product of the magnitudes of A and B.

These steps in Figure 4.1 can be demonstrated through the following example:

Text 1: Ahmed reads more books than Ali do

Text 2: Ali reads less books than Ahmed

We want to know how similar these texts are, purely in terms of word counts. A list of the words from both texts is constructed without redundancy, which will be as "Ahmed reads more books than Ali do less"

Then, the number of times each of these words in the constructed list appears in each text is counted. This count can be summarized in a table like the following

Table 4.1 : Vector Space for T1 And T2

	Ahmed	reads	more	books	than	Ali	do	less
T1	1	1	1	1	1	1	1	0
T2	1	1	0	1	1	1	0	1

We are not interested in the words themselves. We are interested only in those two vertical vectors of counts. By applying the steps shown before in this chapter we get:

$$\text{Similarity} = \cos(\theta) = \frac{1*1+1*1+1*0+1*1+1*1+1*1+1*0+0*1}{\sqrt{1^2+1^2+1^2+1^2+1^2+1^2+1^2+0^2}.\sqrt{1^2+1^2+0^2+1^2+1^2+1^2+0^2+1^2}}$$

$$= \frac{5}{\sqrt{7}.\sqrt{6}}$$

$$= \frac{5}{6.5}$$

$$= .76$$

The result of this calculation will always be a value between 0 and 1, where 0 means 0% similar, and the 1 means 100% similar.

Second: calculating word to word using enhanced traditional string matching through WordNet ontology

Given two words: *w1* and *w2*, we need to find the similarity of (*w1*, *w2*).

In WordNet, words are organized into synonym sets (synsets), so if two words occur in the same synset we can say they are similar or related to each other.

For *w1* and *w2* , the proposed approach executes as shown in Figure (4.1)

Step 1: prepare a set *s1* that contains all the equivalent words for *w1* in the same synset

Step 2 : prepare a set *s2* that contains all the equivalent words for *w2* in the same synset

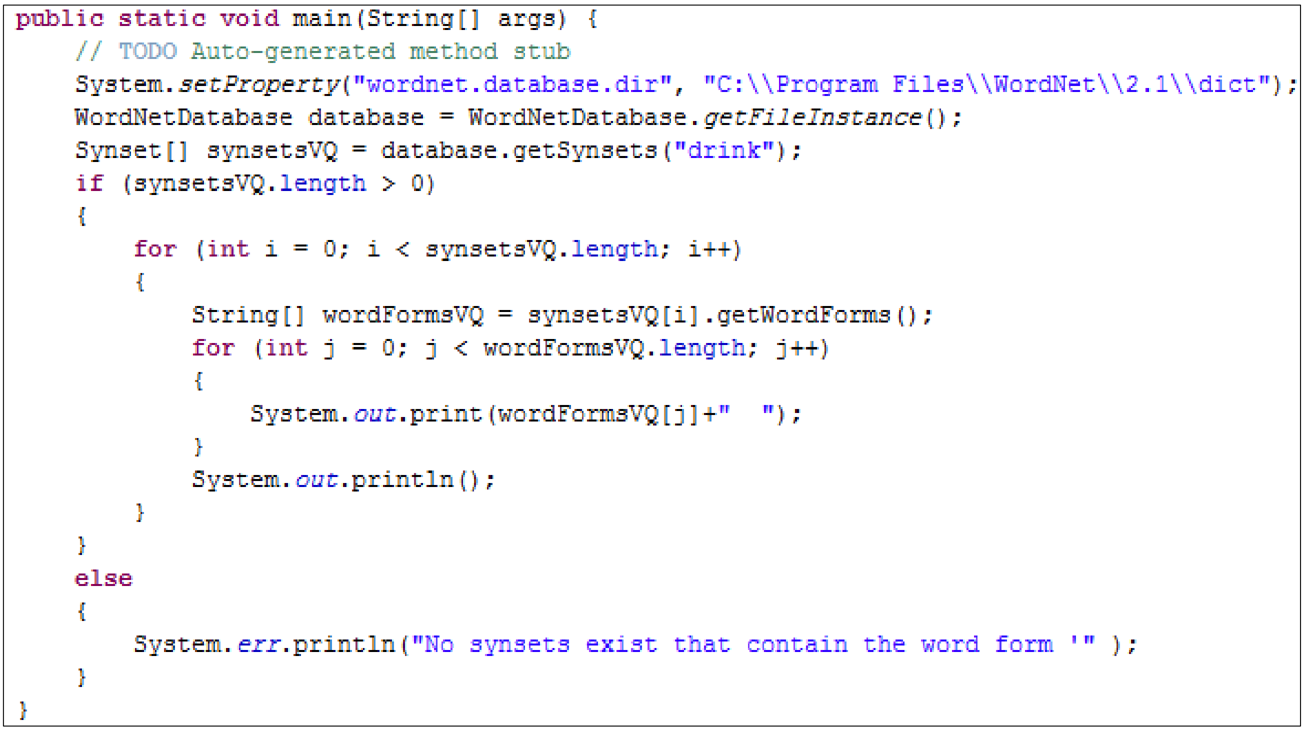
Step 3: for each word *w* in *s1*, find out if there is an intersection with *s2* and vice versa

Step 4: if there is intersection then w1 and w2 are similar

Figure 4.1: Word to Word Similarity

To make it more clear consider the following example.

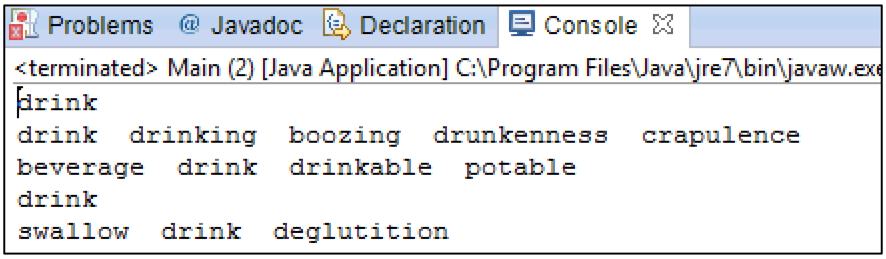
Let *w1* be "drink" and *w2* be "swallow", to see if they are similar in meaning we use WordNet dictionary to see if there is an intersection between the meanings of the two word using this piece of code.



```
public static void main(String[] args) {
    // TODO Auto-generated method stub
    System.setProperty("wordnet.database.dir", "C:\\Program Files\\WordNet\\2.1\\dict");
    WordNetDatabase database = WordNetDatabase.getFileInstance();
    Synset[] synsetsVQ = database.getSynsets("drink");
    if (synsetsVQ.length > 0)
    {
        for (int i = 0; i < synsetsVQ.length; i++)
        {
            String[] wordFormsVQ = synsetsVQ[i].getWordForms();
            for (int j = 0; j < wordFormsVQ.length; j++)
            {
                System.out.print(wordFormsVQ[j]+"  ");
            }
            System.out.println();
        }
    }
    else
    {
        System.err.println("No synsets exist that contain the word form '" );
    }
}
```

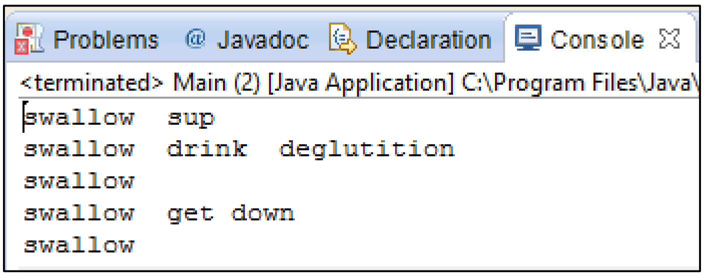
Figure 4.2 : Code for Retrieving Word Synonyms

Considering the steps in Figure 4.1 s1 and s2 which are the sets containing the synonyms of w1 and w2 will be as shown in Figure 4.3 and Figure 4.4 respectively.



```
<terminated> Main (2) [Java Application] C:\Program Files\Java\jre7\bin\javaw.exe
drink
drink  drinking  boozing  drunkenness  crapulence
beverage  drink  drinkable  potable
drink
swallow  drink  deglutition
```

Figure 4.3: Screen Shoot of Synonyms for The Word "Drink"

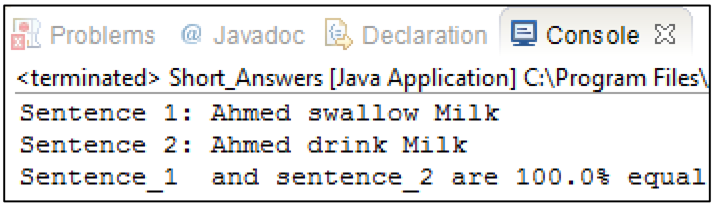


```
<terminated> Main (2) [Java Application] C:\Program Files\Java\
swallow  sup
swallow  drink  deglutition
swallow
swallow  get down
swallow
```

Figure 4.4: Screen Shoot of Synonyms for The Word "Swallow"

Comparing Figure 4.3 and Figure 4.4, there is an overlapping in the meaning of the word "drink" and the word "swallow". In Figure 4.3 which represents the alternate meanings of the word "drink", the synonym "swallow" appears and the same thing applies to the word "swallow" as the synonym "drink" appear. So we can say "drink" and "swallow" are similar words.

In Section 4.1.1 we mentioned that sentences are made up of words, so it is reasonable to represent a sentence using the words in it. To measure the similarity between two sentences, we calculate the similarity between the sentence "Ahmed drink Milk" and the sentence "Ahmed swallow Milk" the system should evaluate them to be 100% equals and this is really what our system do as shown in the result screen in Figure 4.5.



```
<terminated> Short_Answers [Java Application] C:\Program Files\
Sentence 1: Ahmed swallow Milk
Sentence 2: Ahmed drink Milk
Sentence_1  and sentence_2 are 100.0% equal
```

Figure 4.5: Screen Shoot of System Results

4.1.2. Grammatical Semantic Similarity

The second method used in this system is a grammatical similarity measure. As mentioned earlier in Section 4.1 the word to word similarity measure is not enough to judge how far the similarity between two short texts is. There can be some sentences that use the same vocabulary but carry completely different meanings.

Because of this, the grammatical similarity measure method is used beside the former word-to-word method, the grammatical similarity method finds out how far two sentences are similar from a grammatical view.

For example, if a sentence contains the verb "circles" another sentence contains the verb "rotates", we can say that the two sentences are similar regarding to the verb terms.

The same can be applied to the subject and object of the sentence. It is beneficial to clarify that the verb, subject, object or any part-of-speech terms can be detected using software tools called "Part-Of-Speech" Taggers (POS).

The grammatical semantic method works as shown in Figure 4.6

1. Using POS tagging tools, analyze the text and determine which term is the verb, which is the subject and which is the object if it exists. (Some sentences don't contain object like the sentence "the cat died") .
2. Using WordNet, compare the verb in the key answer with the verb in the student answer to see if they are equal in meaning.
3. Find the subject in the key answer and compare it with the subject in the student answer to see if they are equal in meaning.
4. If the key answer contains an object, then compare it with the object in the student answer if any.
5. According to the similarity between the 3 main component of the sentence (verb, subject, object), the approach can approximately calculate the similarity between the key answer and the student answer)

Figure 4.6: Grammatical Semantic Similarity Method

In our implementation of the scoring system we combine the word_to_word similarity approach and the grammatical similarity approach by taking the average of them. We have done that because depending on only on word_to_word approach will result in scoring mistakes as it consider the two sentences are similar if they just contain the same word while they may be completely different in meaning as we declare in Section 2.5

The next section presents the system realization, showing its components and how they were implemented.

4.2. The Automated Scoring System

The proposed system consists of a number of procedures for computing the similarity between two texts, before start computing the similarity between two given sentences. These sentences have to be preprocessed in order to be suitable for computation.

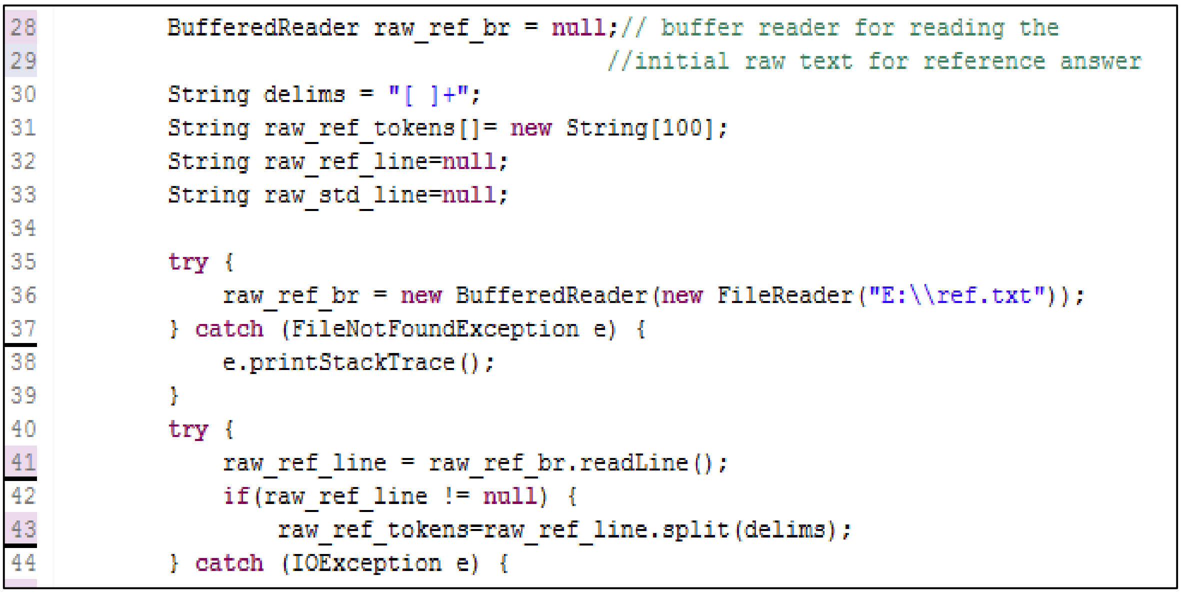
4.2.1. Preprocessing Stage

The step of preprocessing include a tokenization stage and then part-of-speech tagging stage

Tokenization: Tokenization is the process of breaking a stream of text up into words, phrases, or other meaningful elements called tokens. The list of tokens becomes input for further processing such as parsing or text mining. Tokens are separated by whitespace characters, line break, or by punctuation characters [45].

The above paragraph talks about word tokenization, which breaks the text into words. There is another type of tokenization "Sentence tokenization" which is the problem of dividing a string into its component sentences. In English and some other languages, using punctuation, particularly the full stop character is a reasonable approximation. However even in English this problem is not trivial due to the use of the full stop character for abbreviations, which may or may not also terminate a sentence. For example in "*Mr. Smith went to the shops in Jones Street."* *Mr*. is not a complete sentence to stop at. When processing plain text, tables of abbreviations that contain periods can help prevent incorrect assignment of sentence boundaries [46].

In our research we use word tokenization, actually we consider the whitespace characters as shown in the following code snippet in Figure 4.7. Line 3 declares the string *delims* as [ ]+ which means one or more spaces, then line 43 uses the java command *.split(delims)* which splits a string consisting of a group of words into separated words according to the *delims* and then stores them in an array of strings.



```
28          BufferedReader raw_ref_br = null;// buffer reader for reading the
29                                          //initial raw text for reference answer
30          String delims = "[ ]+";
31          String raw_ref_tokens[]= new String[100];
32          String raw_ref_line=null;
33          String raw_std_line=null;
34
35          try {
36              raw_ref_br = new BufferedReader(new FileReader("E:\\ref.txt"));
37          } catch (FileNotFoundException e) {
38              e.printStackTrace();
39          }
40          try {
41              raw_ref_line = raw_ref_br.readLine();
42              if(raw_ref_line != null) {
43                  raw_ref_tokens=raw_ref_line.split(delims);
44          } catch (IOException e) {
```

Figure 4.7: code snippet for tokenization

Part-of-Speech Tagging: In linguistics, part-of-speech tagging (POS or POST), also called grammatical tagging, is the process of marking up a word in a text as corresponding to a particular part of speech (nouns, verbs, adjectives, adverbs, etc.), this process is done based on both word definition, as well as its context, i.e. relationship with adjacent and related words in the sentence.

POS was performed by hand, it is now done using algorithms which associate terms with a set of descriptive tags.

Part-of-speech tagging is harder than just having a list of words and their parts of speech, because some words can represent more than one part of speech at different times. This is not rare in natural languages, a large percentage of word forms are ambiguous.

Commonly there are 9 parts of speech in English: noun, verb, article, adjective, preposition, pronoun, adverb, conjunction, and interjection. However, there are many more sub-categories. For nouns, the plural and singular forms can be distinguished. Words are also marked for their "case" (role as subject, object, etc.); while verbs are marked for tense.

In POS tagging by computer using tagging tools such as SENNA and SEMAFOR , NN is the tag used for singular common nouns, NNS for plural common nouns, PRP for personal pronouns. See Appendix A for more information about other Tags.

4.2.2. Computation Stage:

After the preprocessing stage, a list of words that represents the reference answer and student answer is produced.

Using word_to_word similarity, the text similarity is calculated through applying two approaches, cosine approach and ontology matching approach. See Section 4.1

In order to make the system able to detect the grammatical similarity between two texts. A semantic grammatical approach was applied, this approach compare the verb, subject, and object of the first text to the verb, subject, and object of the second one. See Section 4.1.2 for grammatical similarity details.

The computation process takes the reference and student answer as input, then it applied cosine similarity approach and WordNet word-to-word similarity approach to calculate the word-to-word similarity.

After computing the word-to-word similarity, the computation method applies a semantic grammatical method to support and strengthen the word-to-word approach results. The overall result of the system is the average of the mentioned approaches.

The steps of the computation process is conducted as follows based on the system structure shown in Figure 4.8

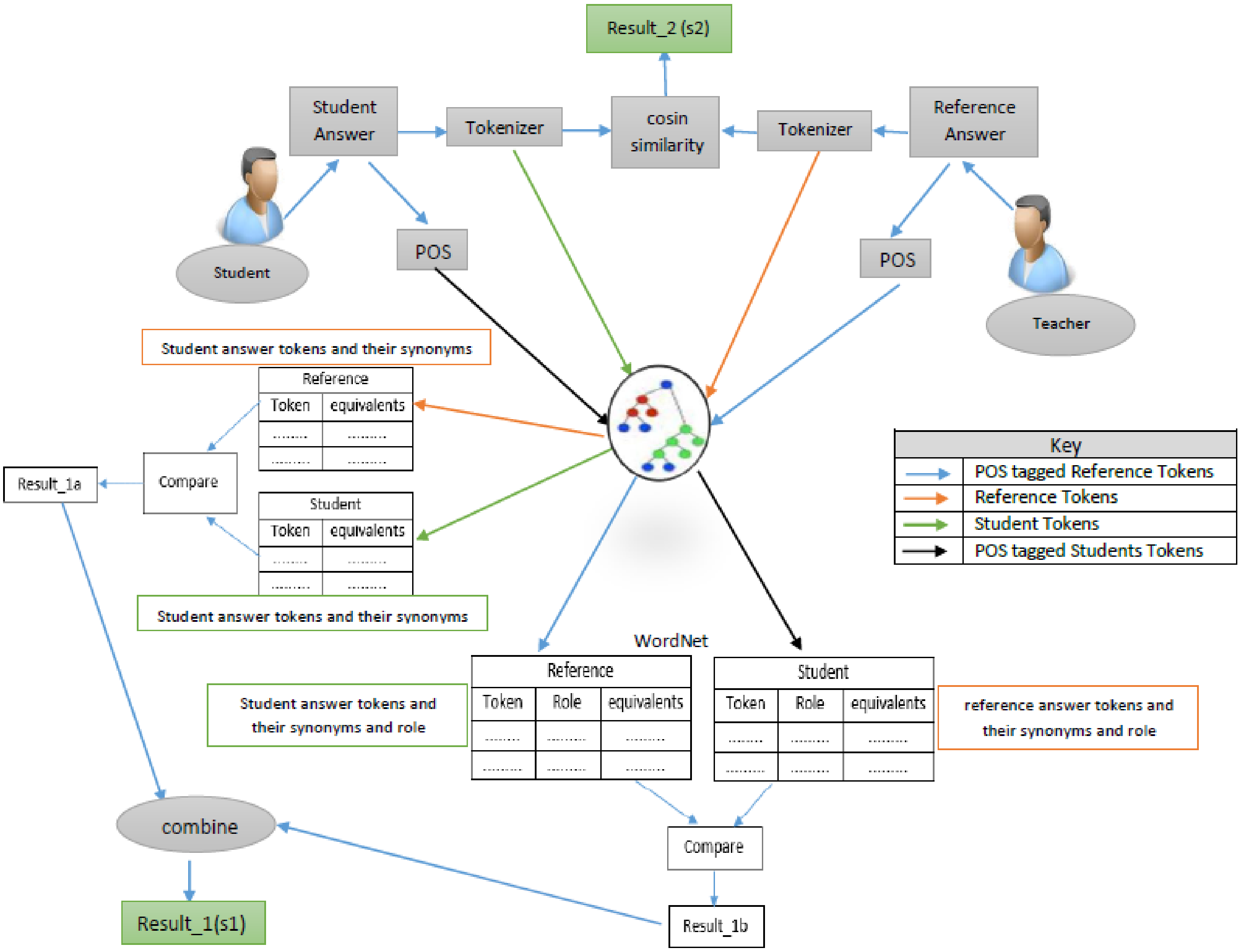


Figure 4.8: Scoring System Structur

- The student's answer as well as the reference answer is broken into tokens, each token consists of a single word
- Connect to WordNet through JAWS or other available API.
- Obtain the equivalents in meaning for the reference answer tokens and equivalents for the students answer tokens using WordNet ontology.
- Use the tokens outputted from the previous step to calculate the percentage of reference tokens found in the student tokens (result_1.a) then transform this percentage to 5 marks to be in line with the marks given by humans in the data set, see Section 4.3.2. the result obtained (result_1.a) is a part of the overall score

- Use one of the available semantic role tool (see Section 2.5) for labeling each token in the student's answer with its suitable role (verb, subject, object.etc.), the same thing goes out for reference answer.
- Determine if the reference answer verb(s) is equal (in spelling or meaning) to the student's answer verb(s)
- Determine if the reference answer subject is equal (in spelling or meaning) to the student's answer subject
- Determine if the reference answer object(s) is equal (in spelling or meaning) to the student's answer object(s)
- Calculate the percentage of verb, subject, object intersection between the reference and the student answer (result_1.b) and transform this percentage to 5 to be in line with the marks given by humans in the data set.
- Combing (result_1.a) and (result_1.b) together to get result_1
- Apply cosine approach to calculate the similarity between the student and reference answers (result_2), then transform this percentage to 5 marks to be in line with the marks given by humans in the data set.
- Return the overall score which is average of the mid results

  We can summarize the overall result from the following equation:

$$\text{Overall score=average (result\_1 : result\_2)}$$

4.3. Implementation of the Scoring System

In this section we will pass through the implementation process of our scoring system, first we will list the tools used in the implementation and talk about the functionality of each tool in the scoring system. Then we will give an orientation about the data set and its structure, and finally we will discuss some code snippets.

4.3.1. Tools and Programs

To construct our system, we utilize the following tools and programs.

1. Eclipse: is an integrated development environment (IDE). It contains a base workspace and an extensible plug-in system for customizing the environment. It is written mostly in Java, and can be used to develop applications.
2. WordNet: is a lexical database for the English language. It groups English words into sets of synonyms and provides short definitions and usage examples. WordNet can thus be seen as a combination of dictionary and thesaurus
3. JAWS: is an API that provides Java applications with the ability to retrieve data from the WordNet database. It is a simple and fast API that is compatible with both the 2.1 and 3.0 versions of the WordNet database files and can be used with Java 1.4 and later.
4. SENNA: is a software distributed under a non-commercial license, it can be used for part-of-speech (POS) tags, semantic role labeling (SRL) and syntactic parsing (PSG).

4.3.2. Data Set

The data set used in this research was prepared by Prof.Aly Aly Fahmy and Eng. Wael Hassan Gomaa from Faculty of Computers and Information at Cairo University, Egypt [25].

This data set is used as a benchmark for automatic questions grading as it contains a collection of students' answers and grades.

The data set questions cover only one chapter of the official Egyptian curriculum for Environmental Science subject. It consists of 61 questions and 610 answers, 10 answers per each question.

The number of students who answered the questions was about 25 student, and the questions were distributed randomly among them.

The data set supports 4 types of questions:

- Type_1: Define the scientific terms.
- Type _2: Explain.
- Type _3: What are the consequences of ?
- Type _4: Why.

The data set was divided according to the questions' types into 4 files, type_1 file contains 18 questions, type_2 file contains 6 questions, type_3 file contains 13 question and finally type_4 file contains 24 questions.

Each file is formatted as follows: for each question, first line contains the question, second line contains the model answer and finally 10 lines of students' answers.

Each student answer is represented in a line that has two marks each enclosed in square brackets [ ].

Answers were scored by two specialists who marks each question with values between 0 and 5.

Table 4.2 shows a sample question from the above data set

Table 4.2: Sample Data Set

Question	Define the term Environment		
Model Answer	All around man of the components of living or non-living affects and is affected by it.		
Students Answers		Human_1 grade	Human_2 grade
Student1 Answer	A word that means the space or range that is living organism and, of course, is adapted to the environment.	[3]	[3]
Student2 Answer	Space, center, who live by the organism and affects and is affected by it**.**	[4]	[4]
Student3 Answer	Surroundings of the components of human beings affect and be affected by it	[4]	[4]

Chapter 5.

Experimental Results and Evaluation

This chapter presents the experiments that was carried out to verify the ability of our system to score students answers with a fair mark. It discusses the results of the experiments and mentions the factors that have contributed to existence of some differences between human mark and system mark. It also calculate the accuracy of the system to judge if it can be used in the education process for scoring student answers.

5.1. Experiments

To evaluate the accuracy of the system a series of experiments were carried out. In each experiment a single question and a group of students' answers for that question were tested.

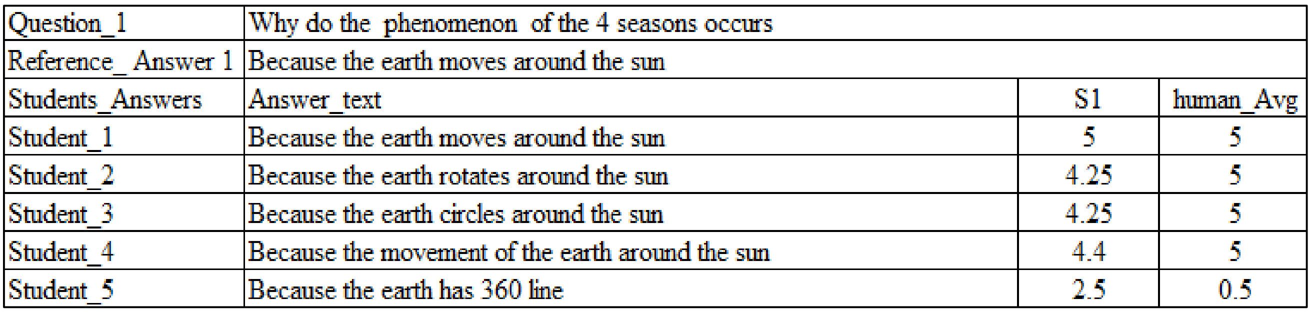
The system passes the reference answer to a tokenizer which divides the reference answer and the related students' answer into tokens. By using WordNet and the semantic labeler (SENNA) the system finds a list of tokens equivalents for the reference answer tokens and for each student answer tokens. Then the system calculates the percentage of the similarity between reference answer and student answer according to the percentage of intersection between the two answers tokens equivalents. This sub result is represented as (s1) in the results table 5.1. i.e s1 is the system result when using WordNet ontology for calculating text similarity.

The second sub result (s2) is the result obtained by applying cosine similarity on reference answer and the related student answer.

Table 5.1 illustrates the results for question_1. It shows the question text, the reference answer text, and the answers of 5 students.

In the same table s1 represents the score when applying only the ontology based method.

Table 5.1 : System Results for Question_1 Using Wordnet Ontology



Question_1	Why do the phenomenon of the 4 seasons occurs		
Reference_ Answer 1	Because the earth moves around the sun		
Students_Answers	Answer_text	S1	human_Avg
Student_1	Because the earth moves around the sun	5	5
Student_2	Because the earth rotates around the sun	4.25	5
Student_3	Because the earth circles around the sun	4.25	5
Student_4	Because the movement of the earth around the sun	4.4	5
Student_5	Because the earth has 360 line	2.5	0.5

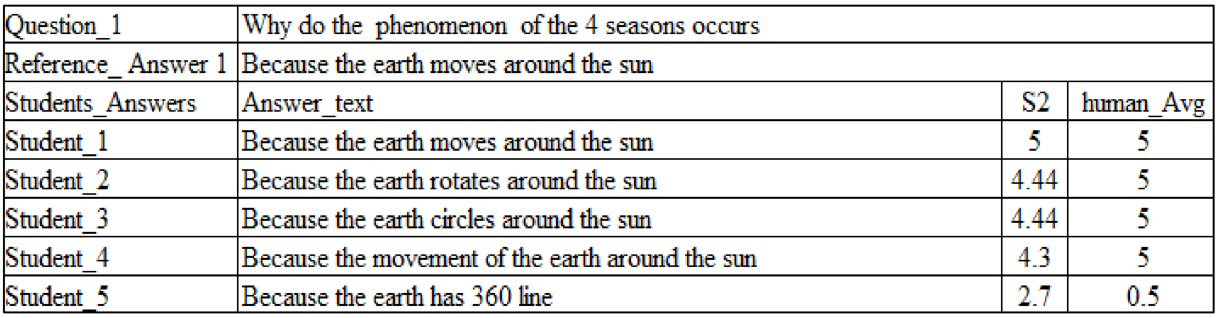
In Table 5.1 we can see the system results hits the target in scoring the answers for first student, and it was nearby in the case of student_2, student_3, student_4 and go far in the case of student_5

The system gives sudents_1 the full mark for his answer, and this is what it should do because the student answer is typically the same as the reference answer.

In case of student_5 the system gives more marks than it deserve, actually it gives the answer 2.5/5 whereas it just deserve 0.5/5. In my opinion this occurred because there is some word matching between the reference answer and student answers like the word "because", "the", "earth".

Table 5.2 shows the results obtained when applying only the cosine method. Comparing the results in table 5.1 and table 5.2 we can see that the results are very close. A reasonable explanation is that in the case of question_1 most of the answers use the almost the same vocabulary in the reference, this makes the WordNet similarity approach performs a word-to-word matching like the cosine method which adopts the word-to-word similarity also.

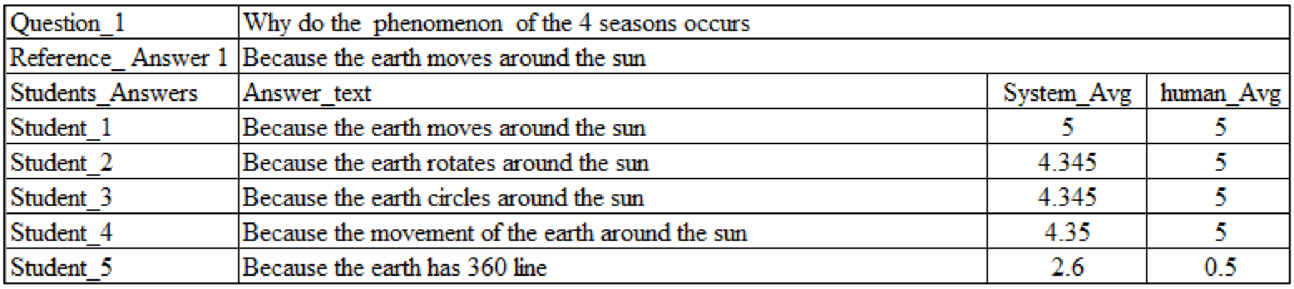
Table 5.2: system results for question_1 using cosine similarity



Question_1	Why do the phenomenon of the 4 seasons occurs		
Reference_ Answer 1	Because the earth moves around the sun		
Students_Answers	Answer_text	S2	human_Avg
Student_1	Because the earth moves around the sun	5	5
Student_2	Because the earth rotates around the sun	4.44	5
Student_3	Because the earth circles around the sun	4.44	5
Student_4	Because the movement of the earth around the sun	4.3	5
Student_5	Because the earth has 360 line	2.7	0.5

Table 5.3 shows the results of the system average for question_1 which expresses a very good results in average.

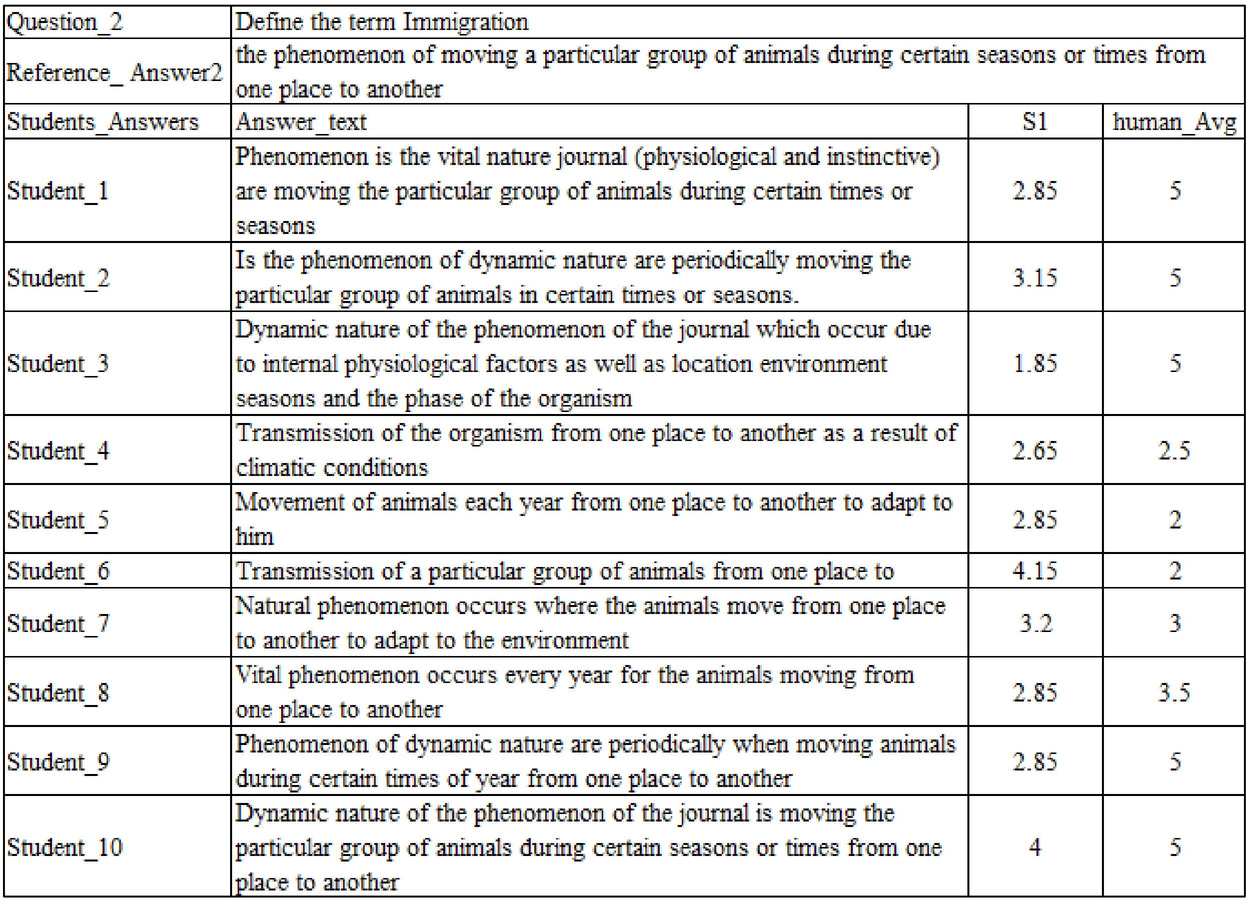
Table 5.3: System Average results (s1, s2) for question_1



Question_1	Why do the phenomenon of the 4 seasons occurs		
Reference_ Answer 1	Because the earth moves around the sun		
Students_Answers	Answer_text	System_Avg	human_Avg
Student_1	Because the earth moves around the sun	5	5
Student_2	Because the earth rotates around the sun	4.345	5
Student_3	Because the earth circles around the sun	4.345	5
Student_4	Because the movement of the earth around the sun	4.35	5
Student_5	Because the earth has 360 line	2.6	0.5

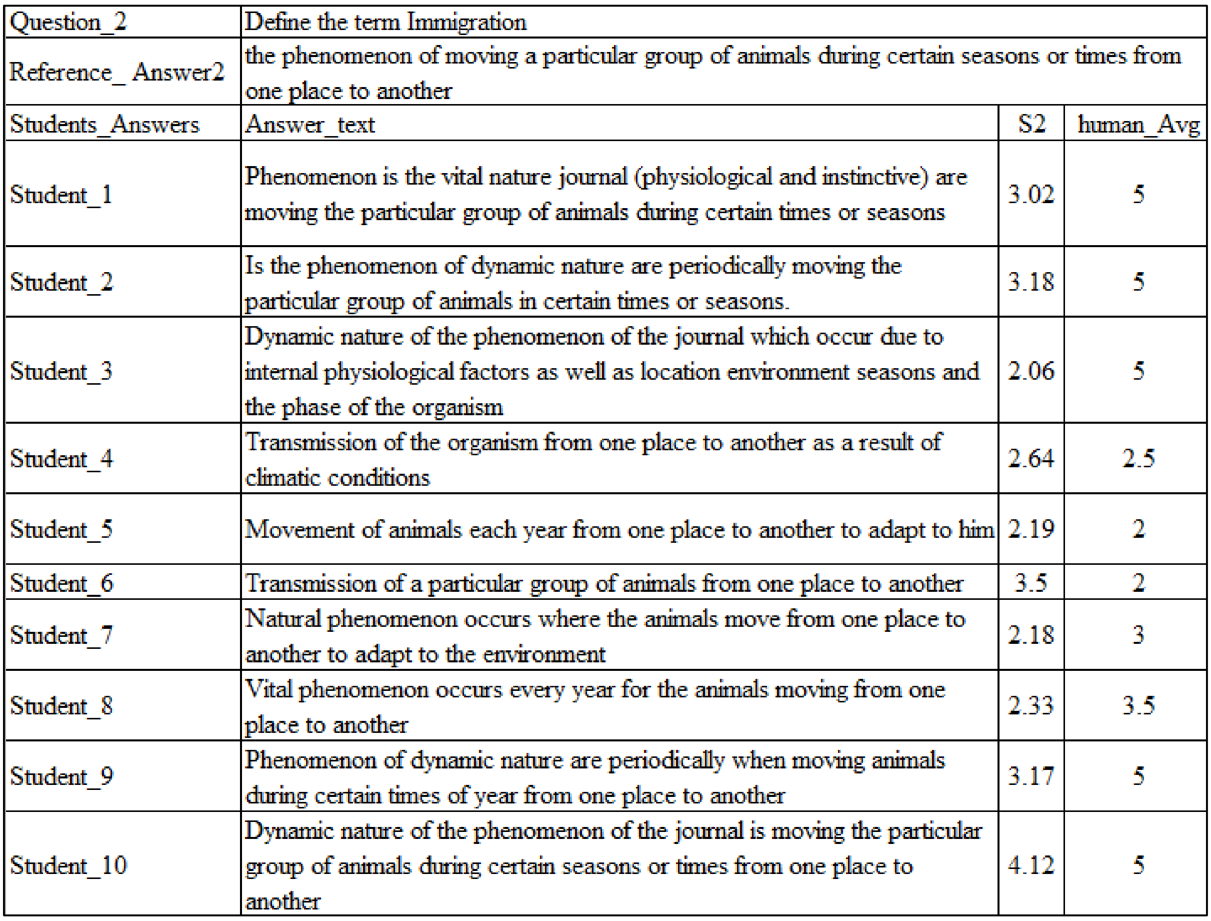
Table 5.4, Table 5.5 and Table 5.6 shows the results obtained from applying Wordnet similarity, cosine similarity and the average result of the two methods on question_2 respectively.

Table 5.4: : System Results for Question_2 Using Wordnet Ontology



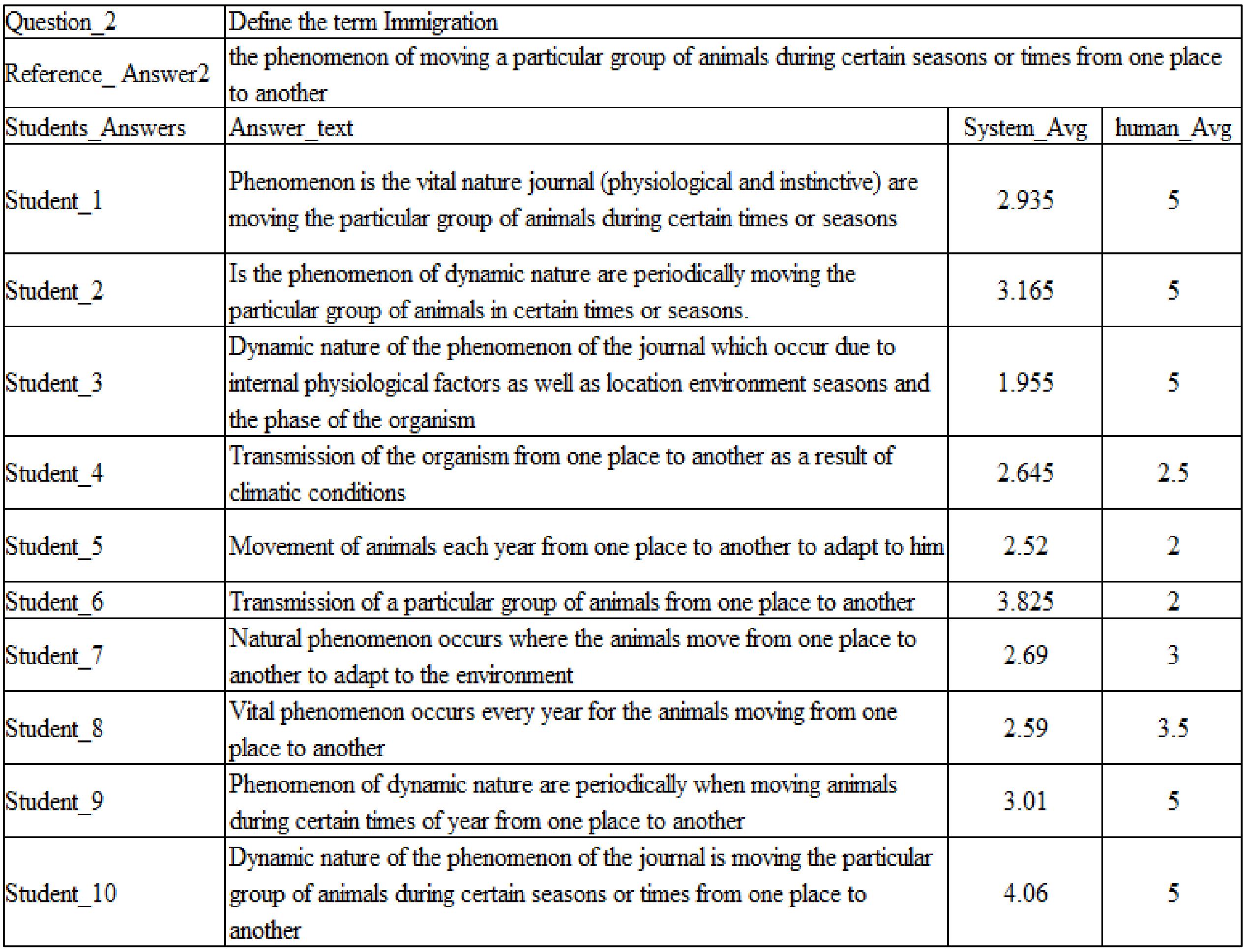
Question_2	Define the term Immigration		
Reference_ Answer2	the phenomenon of moving a particular group of animals during certain seasons or times from one place to another		
Students_Answers	Answer_text	S1	human_Avg
Student_1	Phenomenon is the vital nature journal (physiological and instinctive) are moving the particular group of animals during certain times or seasons	2.85	5
Student_2	Is the phenomenon of dynamic nature are periodically moving the particular group of animals in certain times or seasons.	3.15	5
Student_3	Dynamic nature of the phenomenon of the journal which occur due to internal physiological factors as well as location environment seasons and the phase of the organism	1.85	5
Student_4	Transmission of the organism from one place to another as a result of climatic conditions	2.65	2.5
Student_5	Movement of animals each year from one place to another to adapt to him	2.85	2
Student_6	Transmission of a particular group of animals from one place to	4.15	2
Student_7	Natural phenomenon occurs where the animals move from one place to another to adapt to the environment	3.2	3
Student_8	Vital phenomenon occurs every year for the animals moving from one place to another	2.85	3.5
Student_9	Phenomenon of dynamic nature are periodically when moving animals during certain times of year from one place to another	2.85	5
Student_10	Dynamic nature of the phenomenon of the journal is moving the particular group of animals during certain seasons or times from one place to another	4	5

Table 5.5: System Results for Question_2 Using Cosine Similarity



Question_2	Define the term Immigration		
Reference_ Answer2	the phenomenon of moving a particular group of animals during certain seasons or times from one place to another		
Students_Answers	Answer_text	S2	human_Avg
Student_1	Phenomenon is the vital nature journal (physiological and instinctive) are moving the particular group of animals during certain times or seasons	3.02	5
Student_2	Is the phenomenon of dynamic nature are periodically moving the particular group of animals in certain times or seasons.	3.18	5
Student_3	Dynamic nature of the phenomenon of the journal which occur due to internal physiological factors as well as location environment seasons and the phase of the organism	2.06	5
Student_4	Transmission of the organism from one place to another as a result of climatic conditions	2.64	2.5
Student_5	Movement of animals each year from one place to another to adapt to him	2.19	2
Student_6	Transmission of a particular group of animals from one place to another	3.5	2
Student_7	Natural phenomenon occurs where the animals move from one place to another to adapt to the environment	2.18	3
Student_8	Vital phenomenon occurs every year for the animals moving from one place to another	2.33	3.5
Student_9	Phenomenon of dynamic nature are periodically when moving animals during certain times of year from one place to another	3.17	5
Student_10	Dynamic nature of the phenomenon of the journal is moving the particular group of animals during certain seasons or times from one place to another	4.12	5

Table 5.6: System Average Results (S1, S2) for Question_2

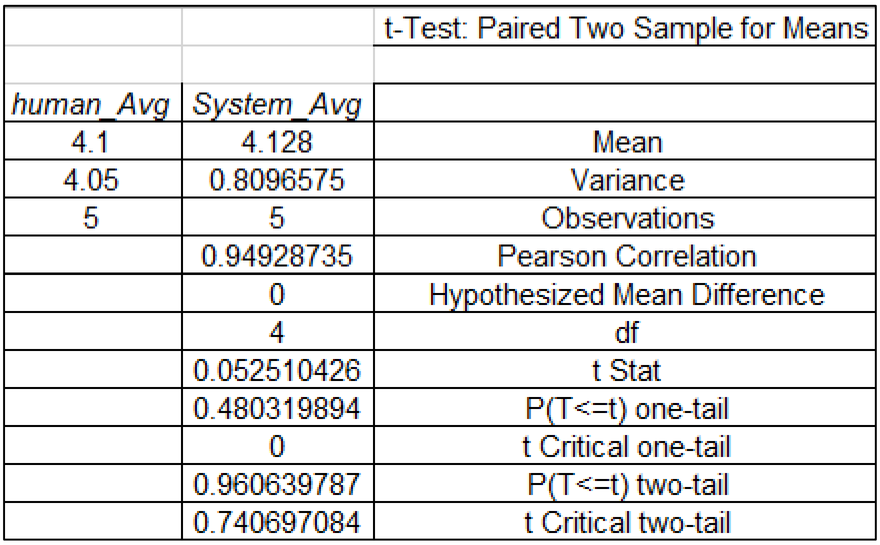


Question_2	Define the term Immigration		
Reference_ Answer2	the phenomenon of moving a particular group of animals during certain seasons or times from one place to another		
Students_Answers	Answer_text	System_Avg	human_Avg
Student_1	Phenomenon is the vital nature journal (physiological and instinctive) are moving the particular group of animals during certain times or seasons	2.935	5
Student_2	Is the phenomenon of dynamic nature are periodically moving the particular group of animals in certain times or seasons.	3.165	5
Student_3	Dynamic nature of the phenomenon of the journal which occur due to internal physiological factors as well as location environment seasons and the phase of the organism	1.955	5
Student_4	Transmission of the organism from one place to another as a result of climatic conditions	2.645	2.5
Student_5	Movement of animals each year from one place to another to adapt to him	2.52	2
Student_6	Transmission of a particular group of animals from one place to another	3.825	2
Student_7	Natural phenomenon occurs where the animals move from one place to another to adapt to the environment	2.69	3
Student_8	Vital phenomenon occurs every year for the animals moving from one place to another	2.59	3.5
Student_9	Phenomenon of dynamic nature are periodically when moving animals during certain times of year from one place to another	3.01	5
Student_10	Dynamic nature of the phenomenon of the journal is moving the particular group of animals during certain seasons or times from one place to another	4.06	5

As we mention in Chapter 1, the objectives of this research is to build an automated scoring system that can assess the students' responses with a mark that is close as possible from the mark given by human.

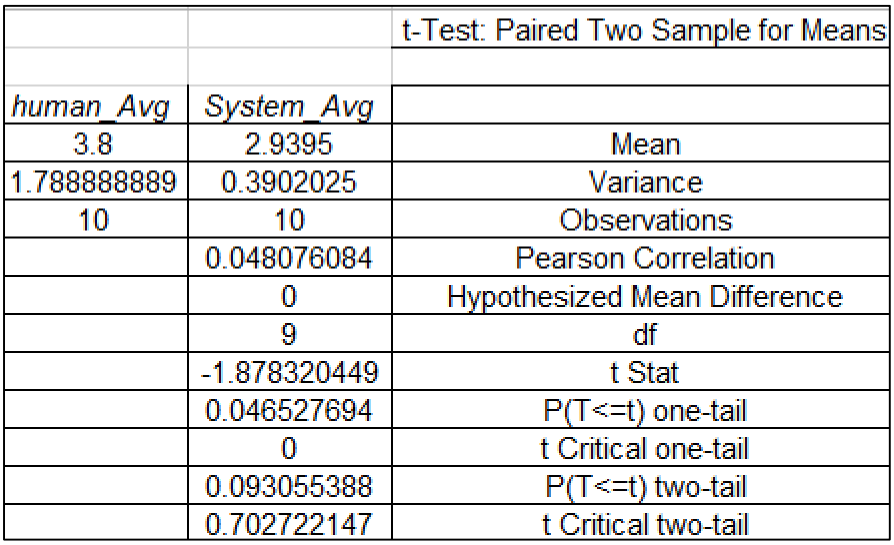
By applying T-test on the results obtained from the system and the those given by human, the results comes as shown in table 5.7 and table 5.8

Table 5.7: :T-test for question 1



		t-Test: Paired Two Sample for Means
human_Avg	*System_Avg*	
4.1	4.128	Mean
4.05	0.8096575	Variance
5	5	Observations
	0.94928735	Pearson Correlation
	0	Hypothesized Mean Difference
	4	df
	0.052510426	t Stat
	0.480319894	P(T<=t) one-tail
	0	t Critical one-tail
	0.960639787	P(T<=t) two-tail
	0.740697084	t Critical two-tail

Table 5.8 :T-test for question 2



		t-Test: Paired Two Sample for Means
human_Avg	*System_Avg*	
3.8	2.9395	Mean
1.788888889	0.3902025	Variance
10	10	Observations
	0.048076084	Pearson Correlation
	0	Hypothesized Mean Difference
	9	df
	-1.878320449	t Stat
	0.046527694	P(T<=t) one-tail
	0	t Critical one-tail
	0.093055388	P(T<=t) two-tail
	0.702722147	t Critical two-tail

Table:5.9 : T-test summary

Question_no	Pearson Correlation	Alpha
Question_1	0.94928735	0.5
Question_2	0.048076084	0.5
Question_3	0.448282	0.5
Question_4	0.386969	0.5
Question_5	0.845862	0.5
Question_6	0.915222	0.5
Question_7	0.500193	0.5
Question_8	0.556459	0.5
Question_9	0.710992	0.5
Question_10	0.931474	0.5

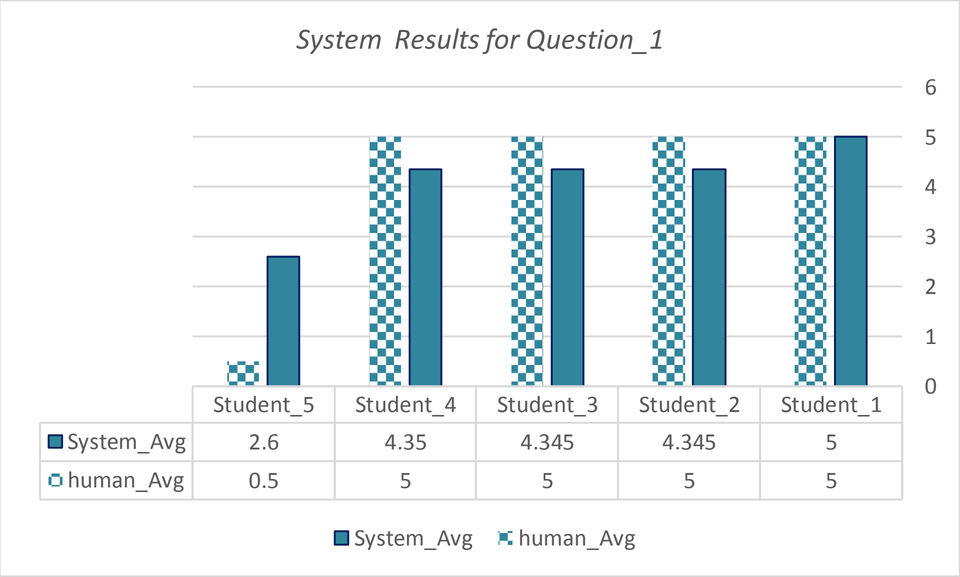
From table 5.9 we can see that the Pearson correlation which is a measure of the strength of between two variables varies from (0.048076084) to (0.94928735) which means that there is a positive relationship between human and system results and we can say it is a good indicator as long as it does not a negative number.

The value of Pearson correlation in question 2 which equals ((0.048076084)) indicates a weak positive relation since it is very close to zero, this is because both the reference answer and the students answer are relatively long text.

This objective has been achieved with a reasonable percentage as shown in Figure 5.1.

Figure 5.1 is a chart between the human score and the system score for 5 students' answers of question_1. From the chart we can notice that the difference between the human mark and the system mark is 0 for student_1 answer, this difference is acceptable in the case of student_2, student_3 and student_4, but it is unacceptable in for student_5.

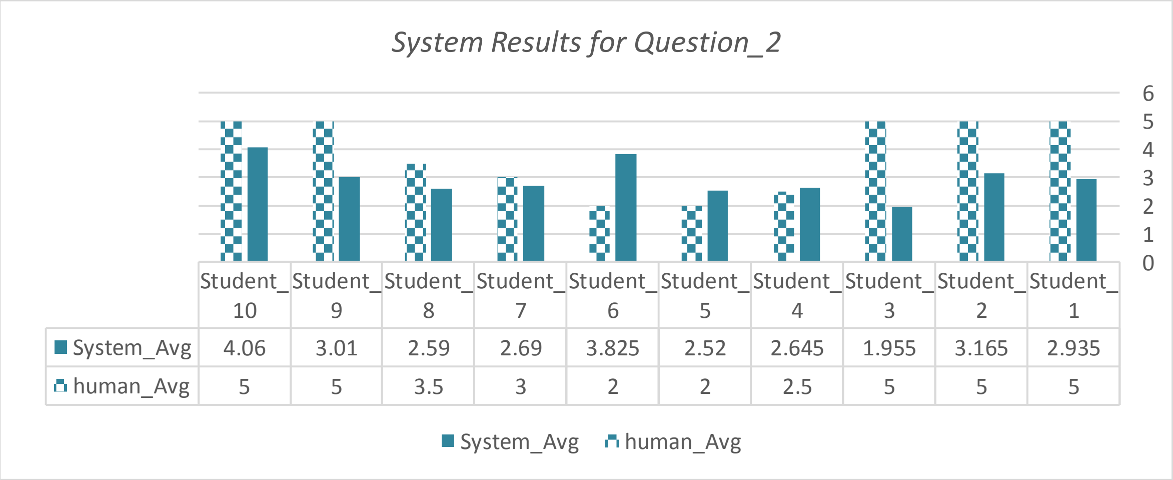
The case of students_5 is case of False Positive (FP), in which the system predict the answer to be true while it is actually false. such FP prediction cased a drop down of the system accuracy.



	Student_5	Student_4	Student_3	Student_2	Student_1
■ System_Avg	2.6	4.35	4.345	4.345	5
▨ human_Avg	0.5	5	5	5	5

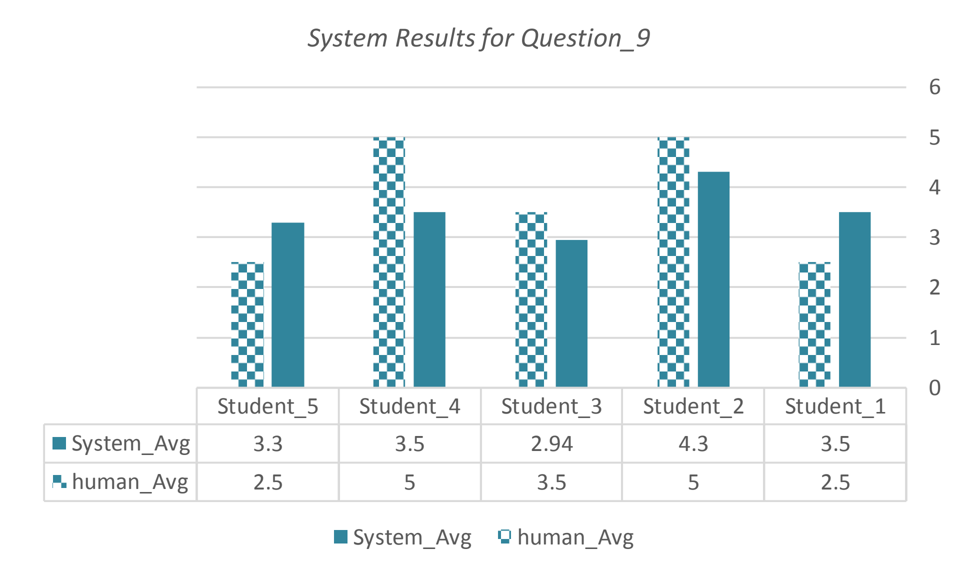
Figure 5.1: System Results for Question_1

Figure 5.2, Figure 5.3, and Figure 5.4 are chart samples for the system results of question_2, question_9 and question_10 respectively.



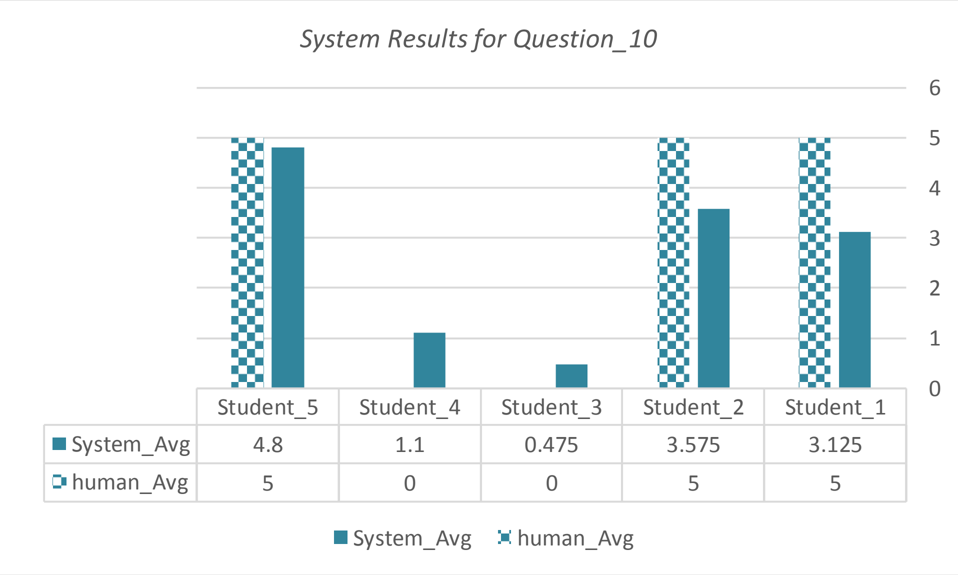
	Student_10	Student_9	Student_8	Student_7	Student_6	Student_5	Student_4	Student_3	Student_2	Student_1
■ System_Avg	4.06	3.01	2.59	2.69	3.825	2.52	2.645	1.955	3.165	2.935
▨ human_Avg	5	5	3.5	3	2	2	2.5	5	5	5

Figure 5.2. : System Results for Question_2



	Student_5	Student_4	Student_3	Student_2	Student_1
■ System_Avg	3.3	3.5	2.94	4.3	3.5
▨ human_Avg	2.5	5	3.5	5	2.5

Figure 5.3: System Results for Question_9



	Student_5	Student_4	Student_3	Student_2	Student_1
■ System_Avg	4.8	1.1	0.475	3.575	3.125
▨ human_Avg	5	0	0	5	5

Figure 5.4: System Results for Question_10

5.2. Evaluation

Before calculating the accuracy of the system, we would like to give a definition of the accuracy. Accuracy is a term that is used to describe the closeness of a measurement to the true value and it can be calculated from the following equation

$$\boxed{\text{accuracy} = \frac{\text{number of true positives} + \text{number of true negatives}}{\text{number of true positives} + \text{false positives} + \text{false negatives} + \text{true negatives}}}$$

Equation 5.1 : Accuracy Formula

Another representation for the above equation is

Accuracy = (TP + TN) / (TP + TN+ FP + FN)

Where

TP: Number of sentences predicted to be similar sentences that actually are similar.

TN: Number of sentences predicted to be dissimilar sentences that actually are dissimilar

FP: Number of sentences predicted to be similar that are actually dissimilar

FN: Number of sentences predicted to be dissimilar that are actually similar

In the context of our system we consider the TP is the case when the human judge the answer to be true and the system judge the answer so, TN is the case when the human judge the answer to be false and the system judge the answer so, FP is the case when the human judge the answer to be false and the system judge the answer to be true and finally, FN is the case when the human judge the answer to be true and the system judge the answer to be false.

Referring to table 5.3 which represents the average result of the system for question_1 we found the following:

No. TP results = 4, No. TN= 0, No. FP = 1, No. FN = 0 ⇒

$$\text{Accuracy} = \frac{4+0}{4+0+1+0} = \frac{4}{5} = 80\%$$

By the same way we calculate the accuracy for the questions from question_2 to question_10.

Table 5.7 summarizes the accuracy for the question that had been entered in the experiment.

Table 5.10: Accracy for Question_1 to Question_10

Question No	Accuracy
Question_1	$4/5 = 80\ \%$
Question_2	$8/10 = 80\ \%$
Question_3	$6/10 = 60\ \%$
Question_4	$8/10 = 80 = \%$
Question_5	$10/10 = 100\ \%$
Question_6	$4/5 = 80\ \%$
Question_7	$4/5 = 80\ \%$
Question_8	$4/5 = 80\ \%$
Question_9	$3/5 = 60\ \%$
Question_10	$5/5 = 100\ \%$

From the above table, the overall accuracy = 80%

5.3. Discussion

The results obtained shows a variation in the system accuracy, in some questions, the system gives a close result to that of human but in other it goes far from it. This can be due to the following reasons:

- The system uses WordNet ontology which is an upper domain ontology, that means that in some cases it may fail in providing equivalents for some domain specific words, for example the word " Phenomenon" has no equivalents in WordNet, so if the students writes another synonym word like "event" or "incedent" or "episode" instead of "Phenomenon", the system wouldn't be able to recognize that these word carry the same meaning as the word "equivalence". This will result in marking the answer with a score less than it deserves.
- Another factor that may had contributed to the difference noticed between the human score and the system score is that some personal name such as "Ali" has

synonyms and meaning in WordNet, so on the time we mean to use the name "as is", the system take in consideration its meaning, this cause a low score from the system than that of the human.

- The accuracy of used semantic Role labeling tool (SENNA) drop down from 88.5% to 75.49% whenever the length of the sentence becomes longer. And as some answers in the data set is longer than 30 word this may affects the system accuracy.
- In the stage of preprocessing we adopt word tokenization, but this is not the only type of tokenization, as there exists a tokenization at the sentence level, in which a string is splitted into a meaningful sentences. Because some answers were long, they need to be separated into sentences and deal with each sentence as a unit, but we didn't use sentence tokenization because it is still challenging and ambiguous process.

The accuracy of the system is 80% i.e. from 100 questions it score 80 question correctly and the rest 20 questions are scored in a false manner.

To be honest the accuracy of the system need to be improved and as we mention in the Introduction chapter, the field of automated scoring systems for short answer is challenging field and need more research.

At this stage we can use this scoring system to determine if the answer is write or wrong without giving a specific score to the student to avoid being unfair to the student.

Chapter 6.

Conclusion and Future Work

We have developed a system for Automatic ontology based scoring for short answers. It aim is to handle the problem of time and effort consuming manual assessement method for short answers.

Our model consists of two stages: preprocessing stage and computation stage, the preprocessing stage formalize the raw data into a form that can be used in the computation stage, on the other hand the computational stage performs the necessary operation on the preprocessed data to measure the percentage of their similarity.

This ontology-based system uses ontology for matching students' answers and reference answer to measure the strength of their relationship by matching not only words but their equivalents also.

Experiments were performed depending on a data set that contains a set of question along with the student answer for these questions and the scores given by two human specialists.

For evaluation purposes, the accuracy was measured. The Results shows that the accuracy of the system is 80% which is very good percentage but still need some improvement.

Using our system model we have partially overcome the problem of the traditional way of scoring short answers in the process of education. This means saving time and effort and returns good results but we have to be careful since the accuracy of the system is just 80% and it fail to score some answers as required.

The system can be improved in multiple directions:

- Using extra domain specific ontology besides WordNet to overcome its generality and be able to deal with specific word related to a Specific domain
- try semantic role labeling tools that can deal good with long sentences and hence gives better results than those of SENNA
- Try the system on Arabic language short answer questions.
- Improve a user friendly interface for the system.

References

[1] A. Budanitsky, "Semantic distance in WordNet: An experimental, application-oriented evaluation of five measures".

[2] B. Eng, "Summary of Semantic Roles and Grammatical Relatiions," 2007.

[3] B. Sami, "Automated Score Evaluation of Unstructured Text using Ontology," in *International Journal of Computer Applications*, 2012.

[4] C. Corley, "Measuring the Semantic Similarity of Texts," in *Empirical Modeling of Semantic Equivalence and Entailment*, 2005.

[5] C. Kulkarni, "Scaling Short-answer Grading by Combining Peer Assessment with Algorithmic Scoring," in *ACM*, 2014.

[6] C. Kulkarni" ‹Scaling Short-answer Grading by Combining Peer Assessment with Algorithmic Scoring " *ACM* .2014 ‹

[7] D. E. Powers, "Challenging the Validity of Automated Essay Scoring," 2001.

[8] D. Jayasri, "Semantic Similarity Measures on Different Ontologies: Survey and a Proposal of Cross Ontology based Similarity Measure," in *International Journal of Science and Research*, 2013.

[9] D. Metzler, "Similarity Measures for Short Segments of Text".

[10] D. whittington, "Approaches to the computerized Assessment of free text responces".

[11] E. T. Services, "Short answer marking engines," 2009.

[12] F. ˇSari´, "Systems for Measuring Semantic Text Similarity," in *Lexical and Computational Semantics*, 2012.

[13] K.Saruladha, "A Survey of Semantic Similarity Methods for Ontology based Information Retrieval," in *Second International Conference on Machine Learning and Computing*, 2010.

[14] M. Lintean, "Measuring Semantic Similarity in Short Texts through Greedy Pairing and Word Semantics," in *International Florida Artificial Intelligence Research Society Conference*, 2012.

[15] M. Mohler, "Text-to-text Semantic Similarity for Automatic Short Answer Grading".

[16] M. Rada, "Corpus-based and Knowledge-based Measures of Text Semantic Similarity," in *American Association for Artificial Intelli- gence*, 2006.

[17] M. Rafi, "An improved semantic similarity measure for document clustering based on topic maps".

[18] M. Sahami, "A Web-based Kernel Function for Measuring the Similarity of Short Text Snippets," 2006.

[19] M. Zhang, "Contrasting Automated and Human Scoring of Essays," 2013.

[20] P. Kumari, "Measuring Semantic Similarity between Words using Page-Count and Pattern Clustering Methods," in *International Journal of Innovative Technology and Exploring Engineering*, 2013.

[21] R. Siddiqi, "On the Automated Assessment of Short Free-Text Responses".

[22] S. Basu, "Powergrading: a Clustering Approach to Amplify Human Effort for Short Answer Grading," 2013.

[23] S. Dikli, "An Overview of Automated Scoring of Essays," in *The Journal of Technology, Learning, and Assessment*, 2006.

[24] S. DIKLI, "Automated Essay Scoring," in *Turkish Online Journal of Distance Education-*, 2006.

[25] S. Jordan, "Short-answer e-assessment questions," 2012.

[26] S. Saqer, "An Automated Grader for Short Answer Responses," 2010.

[27] S. Toranj, "Automated Versus Human Essay Scoring: A Comparative Study," in *nternational Conference on Language, Medias and Culture*, 2012.

[28] S. Valenti, "An Overview of Current Research on Automated Essay Grading," in *Journal of Information Technology Education*, 2003.

[29] T. Slimani, "A New Similarity Measure based on Edge Counting," in *World Academy of Science, Engineering and Technology*, 2008.

[30] V. S. kumaran, "Towards an automated system for short-answer assessment using ontology mapping," in *International Arab Journal of e-Technology*, 2015.

[31] W. H. Gomaa, "Arabic Short Answer Scoring with Effective Feedback for Students," in *nternational Journal of Computer Applications*, 2014.

[32] W.-t. Yih, "Improving Similarity Measures for Short Segments of Text," in *Association for the Advancement of Artificial Intelligence*, 2007.

[33] Y.-S. Lin, "A Similarity Measure for Text Classification and Clustering," in *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING*, 2013.

[34] "https://moodle.org," May 2014. [Online].

[35] "http://en.wikipedia.org/wiki/Sharable_Content_Object_Reference_Model," June 2014. [Online].

[36] "http://en.wikipedia.org/wiki/Grading," May 2014. [Online].

[37] "http://echo.edres.org:8080/betsy/three_prominent.htm," [Online].

[38] "http://www-ksl.stanford.edu/kst/what-is-an-ontology.html," [Online].

[39] "http://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html," [Online].

[40] "http://en.wikipedia.org/wiki/Semantic_role_labeling," [Online].

[41] "http://code.google.com/p/mate-tools," [Online].

[42] "http://ronan.collobert.com/senna/," [Online].

[43] "http://en.wikipedia.org/wiki/WordNe," [Online].

[44] http://www.codeproject.com/Articles/11835/WordNet-based-semantic-similarity-measurement," [Online].

[45] "https://en.wikipedia.org/wiki/Tokenization," [Online].

[46] "https://en.wikipedia.org/wiki/Text_segmentation#Word_segmentation," [Online].

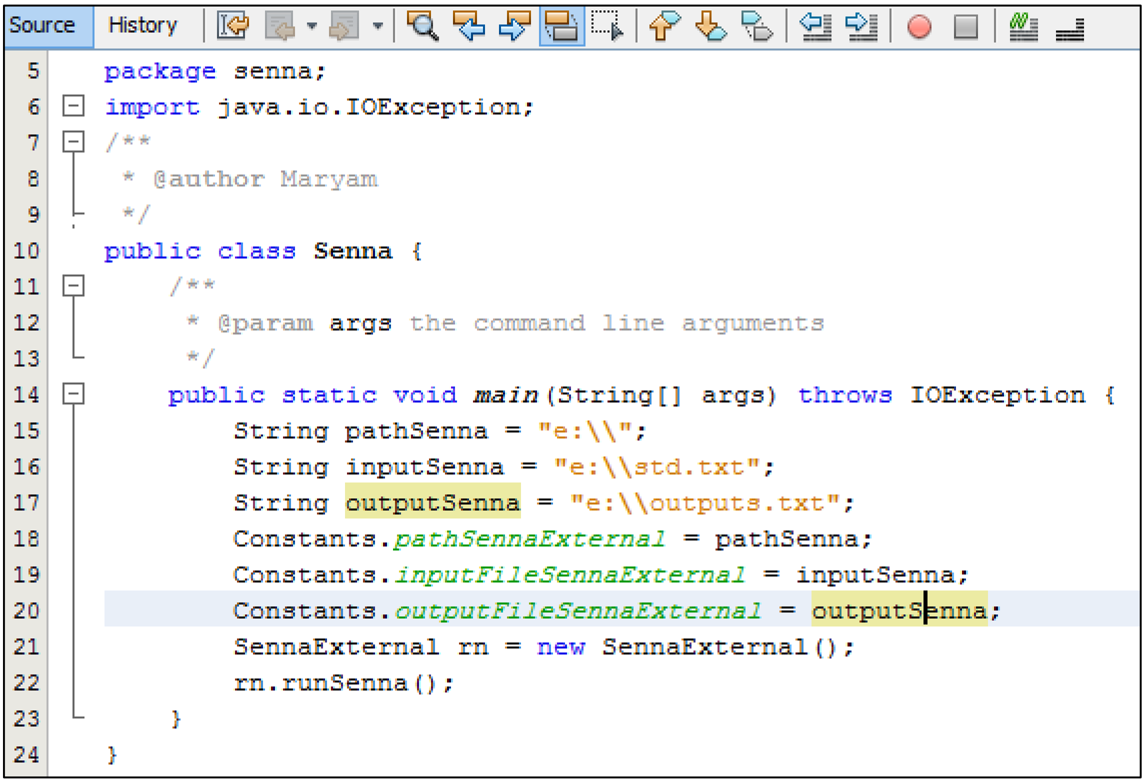
Appendix A: common word tags

- CC Coordinating conjunction
- CD Cardinal number
- DT Determiner
- EX Existential there
- FW Foreign word
- IN Preposition or subordinating conjunction
- JJ Adjective
- JJR Adjective, comparative
- JJS Adjective, superlative
- LS List item marker
- MD Modal
- NN Noun, singular or mass
- NNS Noun, plural
- NNP Proper noun, singular
- NNPS Proper noun, plural
- PDT Predeterminer
- POS Possessive ending
- PRP Personal pronoun
- PRP$ Possessive pronoun
- RB Adverb
- RBR Adverb, comparative
- RBS Adverb, superlative
- RP Particle
- SYM Symbol
- TO to
- UH Interjection
- VB Verb, base form
- VBD Verb, past tense
- VBG Verb, gerund or present participle
- VBN Verb, past participle

- VBP Verb, non3rd person singular present
- VBZ Verb, 3rd person singular present
- WDT Whdeterminer
- WP Whpronoun
- WP$ Possessive whpronoun
- WRB Whadverb

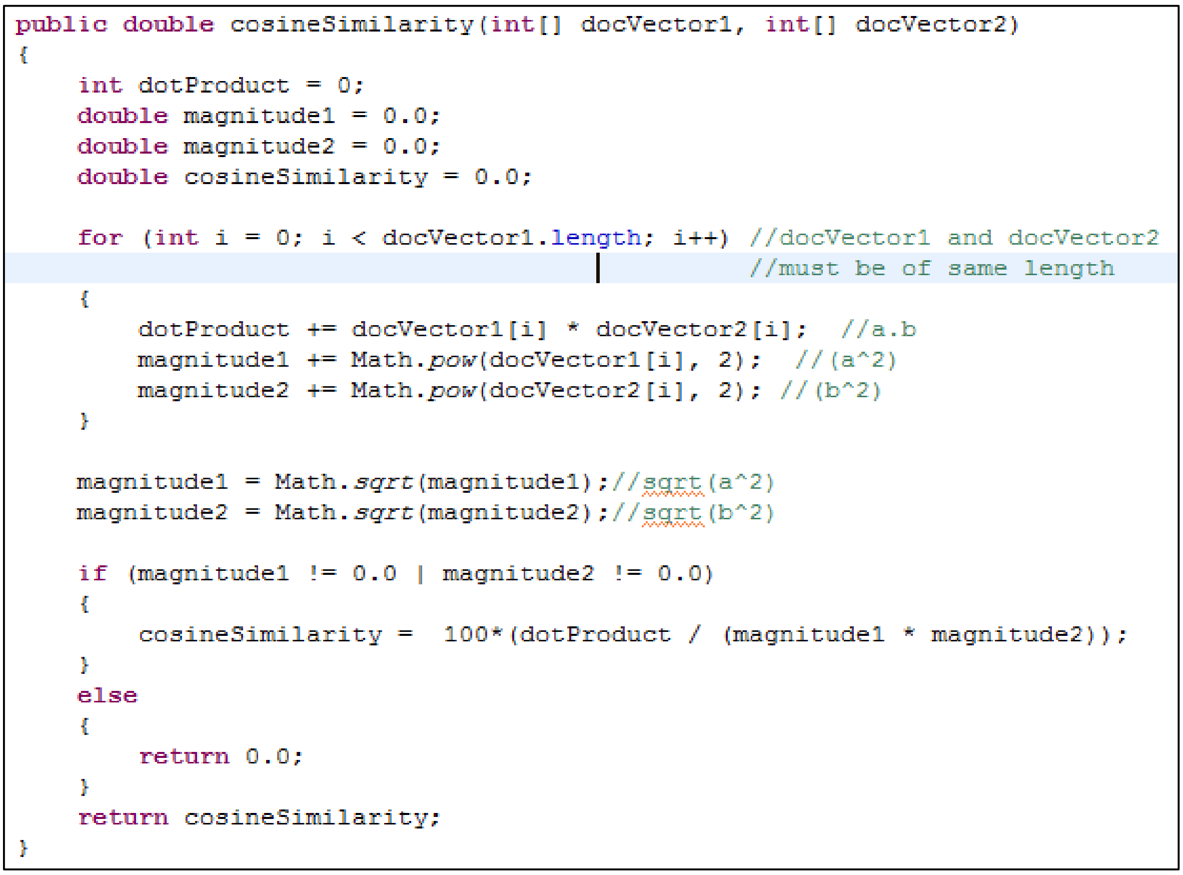
Appendix B

Senna.java



```java
package senna;
import java.io.IOException;
/**
 * @author Maryam
 */
public class Senna {
    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) throws IOException {
        String pathSenna = "e:\\";
        String inputSenna = "e:\\std.txt";
        String outputSenna = "e:\\outputs.txt";
        Constants.pathSennaExternal = pathSenna;
        Constants.inputFileSennaExternal = inputSenna;
        Constants.outputFileSennaExternal = outputSenna;
        SennaExternal rn = new SennaExternal();
        rn.runSenna();
    }
}
```

Cosine Similarity code



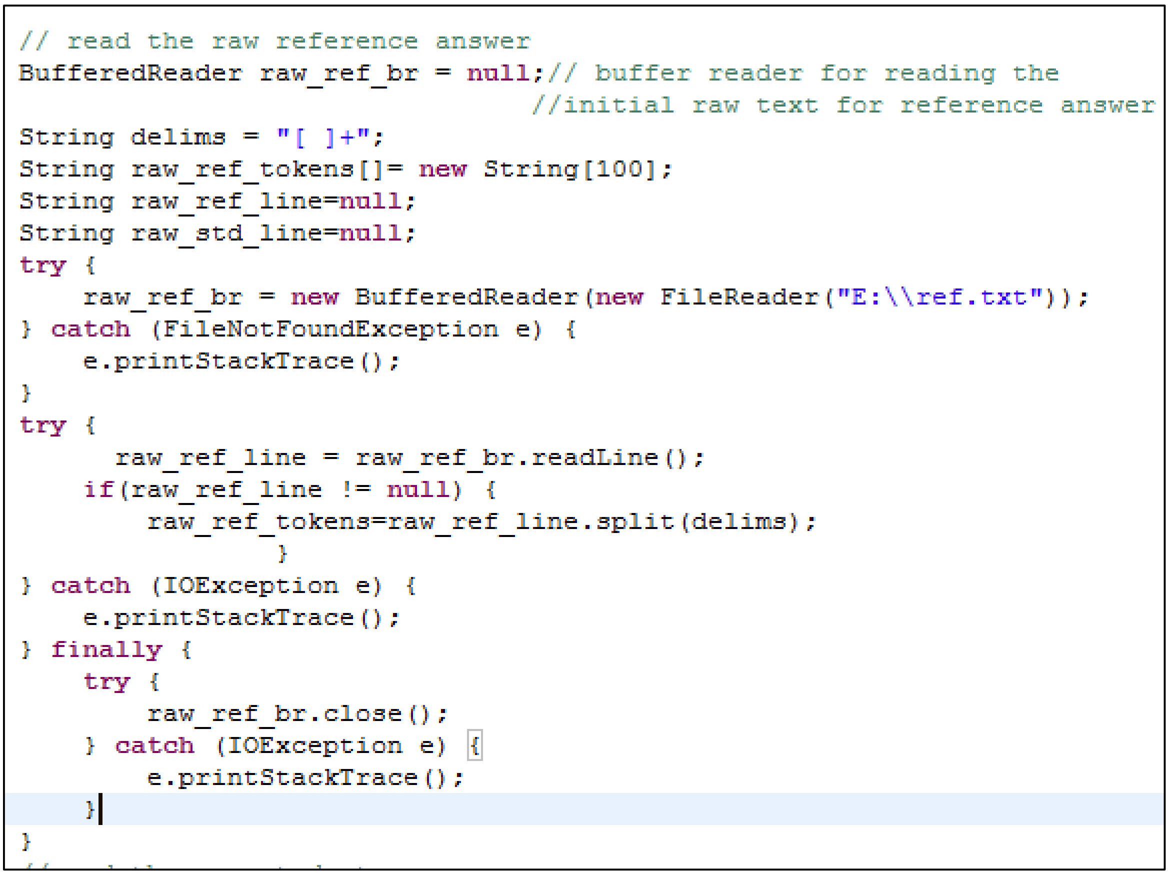
```
public double cosineSimilarity(int[] docVector1, int[] docVector2)
{
    int dotProduct = 0;
    double magnitude1 = 0.0;
    double magnitude2 = 0.0;
    double cosineSimilarity = 0.0;

    for (int i = 0; i < docVector1.length; i++) //docVector1 and docVector2
                                                //must be of same length
    {
        dotProduct += docVector1[i] * docVector2[i];  //a.b
        magnitude1 += Math.pow(docVector1[i], 2);  //(a^2)
        magnitude2 += Math.pow(docVector2[i], 2); //(b^2)
    }

    magnitude1 = Math.sqrt(magnitude1);//sqrt(a^2)
    magnitude2 = Math.sqrt(magnitude2);//sqrt(b^2)

    if (magnitude1 != 0.0 | magnitude2 != 0.0)
    {
        cosineSimilarity =  100*(dotProduct / (magnitude1 * magnitude2));
    }
    else
    {
        return 0.0;
    }
    return cosineSimilarity;
}
```

Reading reference answer texts code



```
// read the raw reference answer
BufferedReader raw_ref_br = null;// buffer reader for reading the
                                //initial raw text for reference answer
String delims = "[ ]+";
String raw_ref_tokens[]= new String[100];
String raw_ref_line=null;
String raw_std_line=null;
try {
    raw_ref_br = new BufferedReader(new FileReader("E:\\ref.txt"));
} catch (FileNotFoundException e) {
    e.printStackTrace();
}
try {
      raw_ref_line = raw_ref_br.readLine();
    if(raw_ref_line != null) {
        raw_ref_tokens=raw_ref_line.split(delims);
                }
} catch (IOException e) {
    e.printStackTrace();
} finally {
    try {
        raw_ref_br.close();
    } catch (IOException e) {
        e.printStackTrace();
    }
}
```

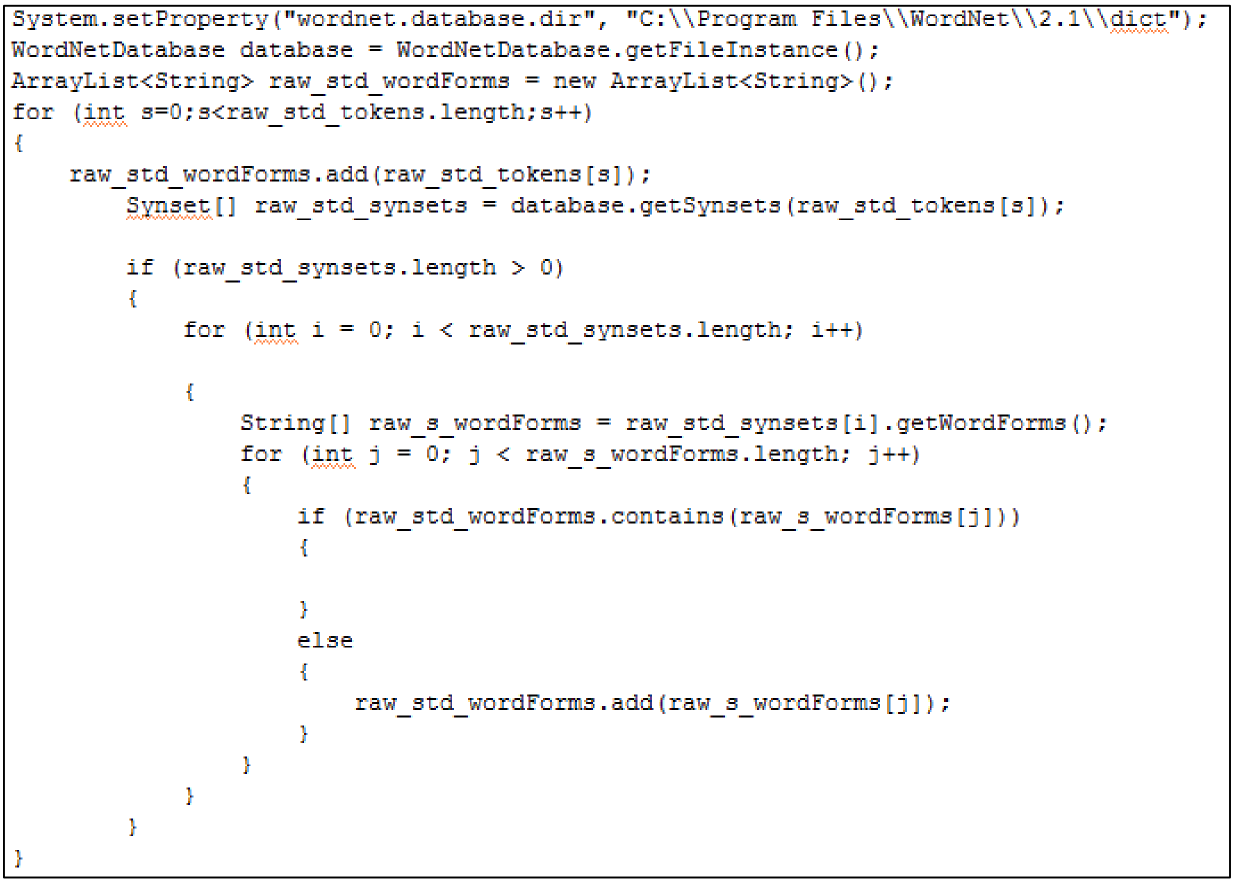
Reading reference equivalents from WordNet code

```
System.setProperty("wordnet.database.dir", "C:\\Program
Files\\WordNet\\2.1\\dict");
            WordNetDatabase database = WordNetDatabase.getFileInstance();
            // read the raw reference answer
            BufferedReader raw_ref_br = null;// buffer reader for reading
the initial raw text for reference answer
            String delims = "[ ]+";
            String raw_ref_tokens[]= new String[100];
            String raw_ref_line=null;
            String raw_std_line=null;

            try {
                  raw_ref_br = new BufferedReader(new
FileReader("E:\\ref.txt"));
            } catch (FileNotFoundException e) {
                  // TODO Auto-generated catch block
                  e.printStackTrace();
            }
            try {

                  raw_ref_line = raw_ref_br.readLine();
                if(raw_ref_line != null) {
                  raw_ref_tokens=raw_ref_line.split(delims);
                            }
            } catch (IOException e) {
                  // TODO Auto-generated catch block
                  e.printStackTrace();
            } finally {
                try {
                  raw_ref_br.close();
                  } catch (IOException e) {
                        // TODO Auto-generated catch block
                        e.printStackTrace();
                  }
            }
```

Reading sysnonyms from WordNet code



```
System.setProperty("wordnet.database.dir", "C:\\Program Files\\WordNet\\2.1\\dict");
WordNetDatabase database = WordNetDatabase.getFileInstance();
ArrayList<String> raw_std_wordForms = new ArrayList<String>();
for (int s=0;s<raw_std_tokens.length;s++)
{
    raw_std_wordForms.add(raw_std_tokens[s]);
        Synset[] raw_std_synsets = database.getSynsets(raw_std_tokens[s]);

        if (raw_std_synsets.length > 0)
        {
            for (int i = 0; i < raw_std_synsets.length; i++)

            {
                String[] raw_s_wordForms = raw_std_synsets[i].getWordForms();
                for (int j = 0; j < raw_s_wordForms.length; j++)
                {
                    if (raw_std_wordForms.contains(raw_s_wordForms[j]))
                    {

                    }
                    else
                    {
                        raw_std_wordForms.add(raw_s_wordForms[j]);
                    }
                }
            }
        }
}
```