العنوان:	Automatic Distractor Generation Using Multiple Similarity Measurements for Multiple Choice Questions
المؤلف الرئيسي:	Saeed, Wael Saeed
مؤلفين آخرين:	Liu, Wei(Super.)
التاريخ الميلادي:	2013
موقع:	سيدني
الصفحات:	1 - 36
رقم MD:	615570
نوع المحتوى:	رسائل جامعية
اللغة:	English
الدرجة العلمية:	رسالة ماجستير
الجامعة:	Western Australia University
الكلية:	School of Computer Science and Software Engineering
الدولة:	أستراليا
قواعد المعلومات:	Dissertations
مواضيع:	الجاسبات الإلكترونية ، هندسة البرمجيات، أسئلة الاختبارات
رابط:	https://search.mandumah.com/Record/615570

CHAPTER 1

Introduction

1.1 Overview

Automated systems for generating multiple choice questions (MCQ) is a very useful technique of consideration in the creation of assessment materials and tools, not only for e-learning systems in educational departments but also for lifelong learning. This is because they provide a self-assessment and analytical evaluation tools to the users, together with immediate feedback which gives the users the opportunity to find out whether the answers submitted are correct or not in order to measure their learning progresses.

The high demand that has been experienced in the implementation of self-assessment tools, has thus created the necessity for the development of automated multiple choice question generators. Educators propose to use it as a system to ease the generation of multiple choice questions, gauge the learning progress of students and provide a mean of reasonable assessment. An automated system for the generation of questions can greatly reduce time, cost and effort that would otherwise be required in manual question setting.

A multiple choice question consists of a sentence stem representing the question which the student is supposed to answer, together with three to four answers, only one of which is the correct one and the rest serving as distractors. Though reasonable evaluation results have been received from the multiple choice questions, their successful effectiveness is inherently connected to the quality, strength and effectiveness of the distracters that are associated with the questions. Effective or reasonable distractors in multiple choice questions have the capability of assessing if the students have a clear understanding of the concept, without unduly presenting too much pressure or being too easily detectable [12].

Papasalouros et.al (2008) [14] generated multiple choice questions automatically by utilising three entities; a knowledge base containing facts pertaining to a specific domain, semantic relationships connecting entities in the knowledge base, and a natural language generation component. Until recently, multiple-choice questions were churned out by applying term extraction, semantic distance calculation and sentence restructuring method on ontologies like WordNet.

According to Papasalouros et.al (2008) [14], generating questions using natural language generation methods on a knowledge base developed using Ontology Web Language (OWL) would be a more effective way to generate multiple choice questions. Utilising OWL can prove to be more efficient, as it is domain-independent and helps access multiple domain ontologies. As the entities are arranged based on class hierarchies, it becomes easier to generate distractors. Moreover, entities are categorised based on their properties, which helps to automatically determine relationships between entities as well as determine data type of an entity. Thus, the process of acquiring correct answers and phrasing distracting options is simplified. However, a problem arises when the same word can mean different things, under different contexts. Such words are often referred to as ambiguous words.

In accordance to Bollegala et.al (2007) [6], semantic similarity can be measured, by using Internet search engines to determine the most plausible and relevant context for a word, using “page count”as a statistical measure. This measure basically analyses and ranks the different contexts in which a word has been searched. To determine the semantic similarity between two words, the individual page count of the two words as well as the page counts of the two words combined can be used. Thus, the word relevant to the context can be determined and used while framing distractors in multiple-choice questions.

Another approach to formulating multiple-choice question is to gather sentences from questions mentioned in a specified set of learning materials, employing a statistical technique to generate a blank part for the question, and create distractors using grammatical and statistical patterns [1]. While this approach may be useful in generating multiple-choice questions for a specified text, it is not as flexible or powerful as the method recommended by Papasalouros et.al (2008) [14].

1.2 Project Objectives and Methodology

In this regards, the study seeks to carry out an analysis of the effectiveness of automated multiple choice question generation and the quality of the distractors that are associated with the questions. It is thus geared towards the generation of effective and high quality distractors through the use of multiple similarity measurements for automatic multiple choice question generation.

This research concentrates on the generation of effective distracters that allow for accurate progress assessment through the implementation of two main similarity measures which are Ontology Distance measures and Normalised Google Distance(NGD) measures. First we make use of automated text summarisation system that analyses the entire text corpus to generate a set of significant sentences and a set of significant keywords. According to the results of the summariser we will be able to generate closest distractors to the desired key by the combination measure of NGD and ontology distance. In other words, our system emploies approaches such as text summariser, NGD and ontology distance to find similar distinct word associated with a query keyword, with the hope to give a set of high quality and effective distractors as a result.

1.3 Dissertation Outline

The remainder of this research dissertation is structured in the following way: Chapter 2 reviews the literature concerning the usefulness of using MCQs and some fundamental background information. Chapter 3 seeks to address the design and methodology of the proposed system, and chapter 4 explains the results of distractors generation system evaluation. The last Chapter 5 concludes the paper with reviewing system's findings and indicating suggestions for future work.

العنوان:	Automatic Distractor Generation Using Multiple Similarity Measurements for Multiple Choice Questions
المؤلف الرئيسي:	Saeed, Wael Saeed
مؤلفين آخرين:	Liu, Wei(Super.)
التاريخ الميلادي:	2013
موقع:	سيدني
الصفحات:	1 - 36
رقم MD:	615570
نوع المحتوى:	رسائل جامعية
اللغة:	English
الدرجة العلمية:	رسالة ماجستير
الجامعة:	Western Australia University
الكلية:	School of Computer Science and Software Engineering
الدولة:	أستراليا
قواعد المعلومات:	Dissertations
مواضيع:	الجاسبات الإلكترونية ، هندسة البرمجيات، أسئلة الاختبارات
رابط:	https://search.mandumah.com/Record/615570

CHAPTER 2

Literature Review

2.1 Overview

Alsubait et al [4] categorised the learning assessment items into two formats:

- First subjective assessment such as short/long answer questions, essays and research paper, which require a lot input from the users and markers.

- The second format is an objective assessment using multiple-choice questions (MCQ) as well as questions that require selective answers.

They argued that objective tests are much harder to generate, often not well-structured, requires a considerable amount of time in their preparation and analysis of relevance as compared to subjective essays. However, they require minimal supervision from the examiners, are more reliable because of marks do not rely on examiners opinions and reflects obvious differentiations among examinees, and can be used in evaluating a wide range of knowledge from a vast amount of information resources. In MCQ assessment, the marking process is much easier than other types of learning assessments and very scalable to large classes. In addition, marks for the questions are easily calculated, with the objectivity of the activity being guaranteed by the fact that the feedback is expected by the users, which can easily be converted into statistical figures for further representation and analysis of the assessment results.

Recent developments in e-learning systems lead to offer abundance of electronic textual resources (e.g ebook, web documents, lecture notes, journals, e-class). While having these e-textual resources presents a myriad of advantages to learners and instructors , there is a need to use an electronic or automatic assessment as a key constituent of E-learning. In other words, the measurement towards students capabilities is significantly required by using e-assessment techniques. However, the reliability of e-assessment systems in such environment is critically essential.

They must be well constructed to identify whether or not examinees have achieved desired objectives successfully.

Multiple-choice questions thus become one of the most popular measures for assessing students in a given learning environment. However, creating multiple-choice question is a time consuming, difficult task and requires expertise. Given the current advances in the field of natural language processing and text mining, powered by the advance of electronic textual resources, it is quite feasible to create such a system capable of automatically generating MCQs, in a bid to reduce time and effort in manually creating such questions.

2.2 Automated Text Summarisation

Automated Text Summarisation is a new technique to automatically summarise original text by extracting the most significant sentences or keyphrases that convey the fundamental meaning of the input document. In comparison with manual summarisation methods, it reduces time and effort on making such document. According to Al-Hashemi [2] there are two main types of text summarisers namely extractive and abstractive summariser. The extractive one is commonly used rather than the abstractive method due to the lack of attempt in producing a summary as much similar as a “human”summary. Furthermore, this popular type of text summariser is mainly based on the concept of information extraction, which extracts the main terms or sentences in a desired text in order to formulate a compressed text [2].

Al-Hashemi [2] followed four phases in his proposed study to design the system that summarises a candidate text using a keywords extraction strategy. The author started with a text preprocessing phase that consists of subprocesses which are firstly restructuring an unstructured text file, stop words removal process, word tagging and stemming. The outputs of the first phase is a set of important keywords that is used in the second phase which is a significant keywords/phrases extraction process. There are several ranking strategies implemented by Al-Hashemi to measure the importance of each word, the frequency of a candidate word in the document, Inverse Document Frequency (IDF) and word location in the text. In his article, he stated that words that have high weight usually exist in the title/headings of the document, have a capital letter and a different font type. However, the third phase describes sentence selection strategy that ranks each sentence in accordance to a set of metrics such as sentence location in both paragraph and text, the length of the candidate sentence and presence of the significant extracted keywords/phrases in the sentence. In the final phrase, the implementation of TFIDF measure is involved to minimising or “filtering”the number of significant sentence and to give more quality to the resulted summary. In terms of system evaluation, the author concluded that the finding of performing Precision and Recall measurements on the system, illustrates 70% overall precision which means the system generated a high quality summary [2].

2.3 Automatic multiple-choice questions Generation

A multiple-choice question consists of items that present short sentences, stating particularly a single question or proposition, and a set of choices (e.g four short answers). A student needs to select the one(s) from these answers. In this research, we focus on MCQs that have a single correct answer. This is the simplest case for automaton the process of MCQ creation. Only one of the answers is the correct answer which is called the key. The rest of the answers are wrong and referred to as distractors. Alsubait et al [4] explained the main structure of multiple-choice questions in more detail. A multiple-choice question has two components: a short sentence (stem), demonstrating a certain problem or question that is extracted from a certain text. A number of options from which the user chooses the correct answer. This option set can be represented as $A = \{a_i \mid 2 \leq i \leq max\}$. This implies that we need at least two answers, one being the distractor. The upper bound is not specified, which measure that we can have as many distractors as the system sees suits. This can further be subdivided as a set of correct options and a set of incorrect answers. The set of correct options can be known as keys (K) depending on the type and number of solutions of the question being asked, which can be represented mathematically as, $K = \{K_m \mid 1 \leq m \leq i\}$. However, the set of incorrect answers that act as distractors (it will be explained in more details in next section) to the student, thus working towards the determination of their academic acuity and knowledge. This can be represented as $D = \{D_n \mid n\colon = i - m\}$ [4].

Furthermore, in their article [4], the explanation of two primary types of similarity are sufficiently represented, which are *semantic* and *relational* similarities. The relational similarity highlights similarities between concepts in terms of their relations. The main advantage of the relational similarity is that it generates multiple-choice questions that tests the higher level of “cognitive abilities”of students.The semantic similarity has the ability to produce a plausible distractor for a set of question, which is critically important for generation multiple-choice questions [4].

2.4 Characteristics of High Quality Distractors

As has been mentioned early in this paper, the multiple choice questions comprise statements of the question, otherwise referred to as the stem of the question, followed by a series of multiple-choice solutions to the question with only one of the answers being the correct solution to the question. The correct solution is referred to as the key of the series with the incorrect solutions being referred to as the distractors. Karamanis et al [11] defined an appropriate distractor as “a concept semantically close to the anchor [key] which, however, cannot serve as the right answer itself”.

An important consideration in the automatic generation of distractors for multiple choice questions is the number of distractors used. The more distractors used, the less the chance of getting the correct answer through guesses. The larger number of distractors, the greater possibility of ambiguity and implausibility of the solutions, presented to the students [13]. An effective distractor plays a major role in generating a high quality automatic multiple-choice questions,however it is also “a major challenge in preparing MCQs” [4]. A distractor must be closely similar to the true answer (the key), to highlight students who have not achieved the required level of experience, abilities and knowledge in the learning subject being examined. Goto et.al (2010) [1] stated that effective distractors should have similar part of speech to generate a reasonable question.These features are significant in the process of creating correct educational assessments. Mostow et al [13] introduced three classes of distractors which are: Ungrammatical, Nonsensical and Plausible distractors. The main effective type is the plausible distractor, because of the main features that it has such as the close similarity to the key in terms of part of speech (POS) and it forms a plausible sentence. This literature reports on number of recent studies exploring the generation of distractors for multiple choice questions, particularly those applied Normalised Google Distance (NGD) [7, 17] which is explained in section 4, and ontology concept [5, 3, 14, 10] that is presented in section 5.

2.5 Normalised Google Distance

A considerable amount of literature has been published on measuring semantic similarity between words and phrases based on web search engines such as Google, Bing and Yahoo. In this study, we will examine whether Google distance can be used to measure similarities between a potential distractor and the key for MCQs. Normalised Google Distance (NGD) is used Google search engine to calculate the similarities between objects based on their names [17]. In accordance to Wong et.al(2007) [17], NGD can be calculated as following:

$$NGD(x,y) = \frac{G(x,y) - min\{G(x), G(y)\}}{max\{G(x), G(y)\}} \tag{2.1}$$

where

$$G(x) = \log \frac{1}{g(x)} \qquad \text{and} \qquad G(x,y) = \log \frac{1}{g(x,y)} \tag{2.2}$$

The $g(x)$ and $g(x,y)$ are the probability of existence of terms x and y. They can be calculated as

$$g(x) = \frac{|x|}{N}, \qquad g(x,y) = \frac{|x \cap y|}{N}, \tag{2.3}$$

where y is the number of Google pages that contain term y and x is the number of Google pages with in term x. $|x \cap y|$ is the number of Google pages contains both terms x and y, also N is defined as $N = |x| + |x \cap y|$.

The authoring environment that is introduced by the system of Cilibrasi and Vitanyi [7] demonstrated the utilisation of Google search engine to facilitate its approach. Cilibrasi and Vitanyi (2007) [7] stated the construct of Google semantics of words as a number of web pages returned by the desired query. Additionally, in their research, they represented the theory behind (NGD) that it is a result of Google-based estimation of the Normalised Information Distance (NID). Where the utilisation of Google code-word length allowed them to estimate the NID with Normalised Compression Distance (NCD) and using Google distribution as a compressor for Google semantics [7].

Similarly, Wong et.al [17] applied Google distance as a gauge for similitude and term length, together with new Tree-Traversing Aunt (TTA) algorithm. In their paper they pointed that for term clustering TTA operates on a two-stage approach; first stage the nodes are divided into sub-nodes by TTAs, and then terms are repositioned in order to attaine ideal clusters using NGD. The findings of this approach are promising. Wong, et al. (2007) highlighted that NGD significantly relies on Google search engine to maintain the capability of measuring the similarity and distance between words at the level of compression. Wilson, et al. (2007) asserted that the innovative application of featureless similarity on the basis of NGD and Wikipedia , exhibited brilliant results, additionally they pointed a set of advantages of this approach such as the ability to detect outliers, generate consistent outputs and identify concealed structures of clusters [17]. On the basis of the studies above which applied NGD in their approaches, we can obtain an approach that uses NGD to measure similarity distance between the key and distractors.

2.6 Ontology Based Measures

In recent years, there has been an increasing amount of literature on the generation of distractors for automated multiple-choice questions. A number of studies have conducted a series of generation MCQs using ontology. According to Al-Yahya [3] ontologies are defined as “Knowledge representation structures which provide a conceptual model of the domain”. The system developed by Papasalouros et al. [14] aims at developing a novel approach for creating multiple optional questions automatically, based on explicit ontologies domain and some linguistic resources. They [14] propose this formulated approach to automate the entire process of assessment, to provide dependable evaluation process.

The tool developed receives an input ontology and produces as output multiple choice questionnaires. The study [14] outlines ways through which some domain ontologies can be used as inputs for questionnaire creation in the education setting. This included domain manual summarisation by both ontology engineers and pedagogic experts, domain manual summarisation in concept map order, and ontology generation automation. Moreover, for education purposes the paper recommends the reuse of the domain ontologies created by a field expert.

Furthermore, the article [14] indicates definite ontology related routine that developing such a system needs to follow. In addition, They use alphabets as the routines for instance; A, B, C, D are used as concepts names, R, S are used as roles names and a, b, c are used as individuals names. Consequently, by use of routines, some strategies were formulated to choose the right answers, and select the distractors. In this regard, the proposed strategies are only concentrated on the semantic aspects in the generation of ontology-based questions process. These strategies include:

- Class-based strategies, which the creation of distractors is dependant on their individuals and classes.
- Property-based strategies; which generate a set of distractors according to their roles.
- Terminology-based strategies contain strategies based on relationships without directly involving individual ontology [14].

To evaluate this approach, Papasalouros et al. [14], utilised five different domain ontologies such as Eupalineio tunnel ontology to examine the system. Afterwards, the set of resulted questions was evaluated by a three-dimension perspective, in terms of pedagogical quantity, syntactical correctness, and number of questions generated. Domain experts found all questions generated from this approach passable for assessment. However, the proposed approach is good at defining of questions semantics, but it offers little in creating syntactically correct questions thus leaving room for future work. Moreover, they suggested that the utilisation of online search engine such as Google, will assist in overcoming the weakness of domain ontologies in future.

Similarly, the work described by Al-Yahya [3] applies ontologies to generate a set of learning assessments. They have explored a system that generate MCQs using an OntoQue engine, from a domain ontology. The OntoQue generates assessment items by iterating through entities in the ontology and implementing the Jena API. The researcher classified the strategy of the study into three categories; class-membership strategy, individual strategy and property strategy. In fact, the author has used the same strategies that Papasalouros et al. [14] have applied. However, she implemented individual based strategy instead of terminology-based strategies, with aim to create Fill-In (FI) items. During the strategy of class-membership, the process of generating distractors applied a random model from classes. The researcher pointed that the approach showed a perfect performance in generating such an assessment, however, it needs improvements in terms of wording using WorldNet and the analysis process should rely on real use cases [3].

Another study of ontology based MCQ is carried out by Bin et al (2009) [5], that conducted ontology-based measure of semantic similarity between concepts. From what has been published in the paper, it is obvious that the implementation of semantic similarity is an essential consideration in knowledge sharing, web mining and MCQs generating.

The problem being solved in the research, however, is centred on the fact that, most studies tend to focus on the measurements of semantic similarity between words rather than concepts, whereas, semantic distance between concepts is the fundamental in this matter. Therefore, Bin et al (2009) [5] discussed two traditional measures of semantics in their work. These are graph based measure and information content based measure. Graph based measure investigates mainly the length and depth between concepts. On the other hand, information content based measure is based on the perception that the illustration of semantic distance between concepts, must be delivered accurately by information content.

The study carried out by Bin et al. (2009) [5] tries to understand the semantic similarities that exist between traditional measures.To overcome problems associated with both measures, they have developed an idea that combines graph based measure and information content based measure as a new measure that known as Ontology Hierarchy Information and Information Content semantic similarity measure (OHIIC) [5].

With the combination of the two measures, values produced between their concepts indicates the relationship that exists between them. Although the new idea is only based on theoretical analysis, it helped to achieve results that both measures could not achieve when used separately. However, to achieve their final results, Bin et al (2009) were required to construct a Concept Tree (CT) from WordNet. The approach was examined against 28 word pairs. To evaluate the success of the new measure, the results produced using OHIIC were compared to those of other measures. The results obtained in the study shows that, in semantic similarity measures, Ontology Hierarchy Information is a significant consideration [5].

Based on ontology distance measure, Jing et.al (2006) [10] explore a new clustering technique that enhances the performance of text clustering. Indeed, this approach can be implemented in distractors generation for MCQs.Therefore, the main reason for producing the study is to develop a new clustering scheme based on the measure of ontology distance. However, as the authors assert it was essential to calculate the term mutual information matrix. This was done with the aid of some methods and technique such as WordNet and other ontology methods.

They have tried to resolve most difficult problems caused by text clustering in text documents. As far as they are concerned, text clustering is a challenging problem when it comes to critical information volumes, complex semantics and high dimensionality. To resolve the problems associated with complex semantics, they have proposed the utilisation of the existing learning ontology techniques, with aid of WordNet in order to calculate the term mutual information (TMI) [10].

Furthermore, they designed a new data model that combines mutual information matrix (MIM) and traditional vector space model (VSM), to evaluate their system. The new model designed considers relationship that exists between terms. With new ontology-based distance measure, the research [10] employed two k-means type clustering algorithms, the standard k-means and the FW-KMeans. The main reason why the authors had to employ these algorithms is due to the fact that k-means algorithms are “efficient” as well as “scalable”. The obtained findings from the study demonstrated that, the two clustering algorithms have performed better progress when using ontology distance [10].

2.7 Conclusion

This literature paper presents a number of related research to automatic distractor generation using multiple similarity measurement for MCQs. We have reviewed studies that employed automated text summarisation, Normalised Google Distance and ontology based measure in their approaches. In regard to text summariser, we have covered some aspects in generating such system like the preprocessing task, keywords extraction, metrics for sentence selection and ranking words/sentences strategies. On the basis of NGD research, we will investigate an approach that uses NGD to measure similarity between distractors and the key for MCQs. Moreover, the studies of ontology-based measure support us with useful explanation of various types of ontology measures such as graph-based measure, information content-based measure and Ontology Hierarchy Information and Information Content semantic similarity measure. In addition, some ontology strategies such as class-based strategy and property based strategy, were presented to explain the process of generating distractors and keys in MCQs. We will apply both NGD and Ontology as multiple similarity measurements to produce distractors automatically for multiple choice questions.

CHAPTER 3

Methodology

3.1 Overview: The proposed System Architecture

In this chapter, the methodology we used to generate MCQs distractors, will be described in detail. The following sections introduce the implementation of the automated distractors generation system, which combines three different approaches (subsystems) as it can be seen from Figure 3.1. Firstly, the automated text summariser generates a set of significant sentences (stems) from the input corpus, and extracts domain keywords (keys) from these sentences, which allows the desired system to create two possible types of questions Fill-In (FI) and MCQs. Secondly, Ontology-based measuring approach provides the study with a list of similar words to the key, and Normalised Google Distance (NGD) presents another list of closest words to the key as well. We selected two nearest ontology terms and one NDG distractor from the finding lists, in order to compose a combined set of distractors.

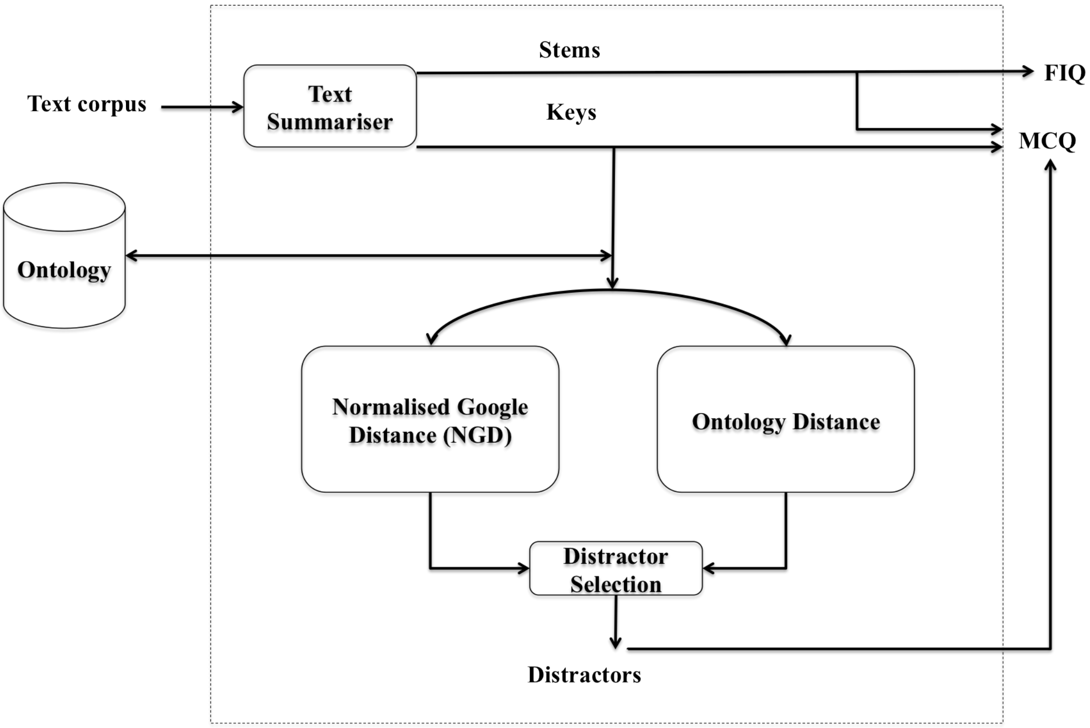


Figure 3.1: The proposed System Architecture

3.1.1 Automated Text Summarisation

Basically, the process of this technique involves three phases as Hovy [9] stated which are topic identification, interpretation and summary generation. During the first phase a type of metrics for scoring sentences must be specified. Additionally, Hovy [9] introduced six types of criteria that can be used in text summarisation: sentence position in the text, Cue phase indicator, Word frequency-based measure, Query and title overlap metric, Cohesive (lexical) connectedness metric and Discourse structure metric. There is no obvious best metric for scoring sentences, however some metrics perform better when it is used in a particular genre. For instance, it is recommended to implement positional criteria in newspaper articles. This is due to the structure of the newspaper which requires to locate the important information in certain parts such as titles or first paragraphs [9].

The approach that we implemented was obtained from the University of Waterloo in Canada which adopts an extractive summarisation method. It was assigned to student as a group project assignment in Computational Linguistics course (CS784) in Spring, 2013 [8]. We employed the starter code that was provided in a Zip file on the course's website. With further developments, implemented another word frequency method from Learner's approach[1] rather than the provided code. The model consists of five stages as shown in Figure 3.2, which are text preprocessing stage, generating terms' values stage that is based on word-frequency metric, calculation of all sentences' scores, significant sentences generation stage and lastly summary generation stage. The resulted significant sentences are used later on to form question sets (stems) from which the domain keywords are also extracted.

[1] Learner, Word Frequency Counter, available from: http://javabycode.blogspot.com.au/2010/12/word-frequency-counter.html

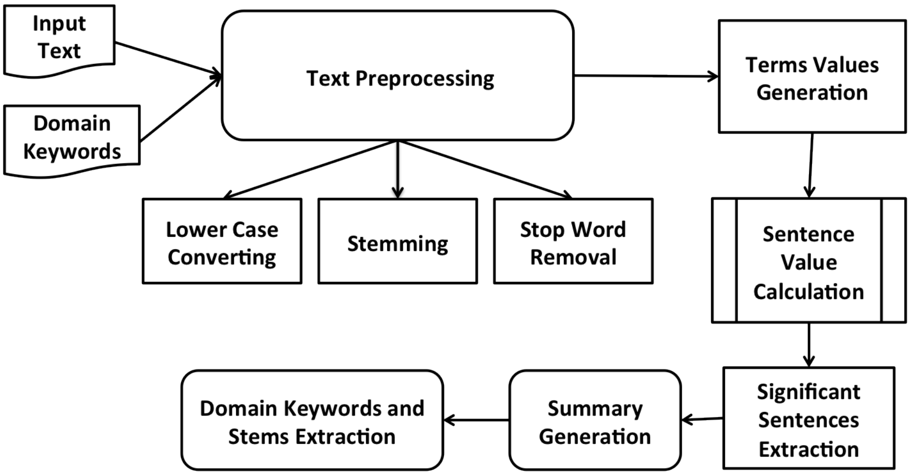


Figure 3.2: The Structure of A Text Summariser

Text Summariser Architecture:

An automated text summariser constructs of the following six stages as mentioned earlier:

1. Text Preprocessing Stage: This stage is essential to load the document into the system initially, then making some critical processes on that text such as converting all words inside the file to lower case which enables the system to merge terms only variation by case. In addition, removing punctuations and stop words also are included in this stage. Stop words are function words that don't have any meaning to the summary like: english articles ("a","an","the"), prepositions, conjunctions, adverbs...etc. Word stemming also takes a place at this stage, which removes suffixes of each word and returns word's stem. A stemming algorithm that is employed here is Porter stemmer [8]. Involving a stemmer in the model is not only useful for keywords extraction process, but also it minimises the size of entire data which enhances the efficiency and performance of the system [2]. The result of this stage is a set of variable keywords in the form of sentences, which will be used in next stages.

2. Generating Terms' Values stage: We used the extracted keywords from pervious stage in order to rank them according to word frequency-based metric. This measure assigns a number of occurrence for each keywords in the text as a value in order to calculate the importance for each sentences (stems) and extract significant nouns (keys) in next stages. Therefore, if a sentence in the text contains words with high rank of frequency (high weight) , then this sentence is probably significant. The output of this stage is that each keyword in the text is obtained a value which represents its frequency (weight) in the document.

3. Calculation of All Sentences' Scores Stage: This is the most challenging stage of automated text summariser. During this calculation process we made use of the outputs from the terms' values generation stage. The calculation method scores each candidate sentence according to the value of each word that we have specified early. Then this score is multiplied with the compression ratio for getting a compressed value. In other words, in this stage each term in a sentence will be checked, then the sentence is scored according to the weight of its words.

4. Generation of Significant Sentences Stage: This stage only selects the highest scored sentences from the calculation stage. The outputs of this operation is a set of the most significant sentences in the entire document. The selection

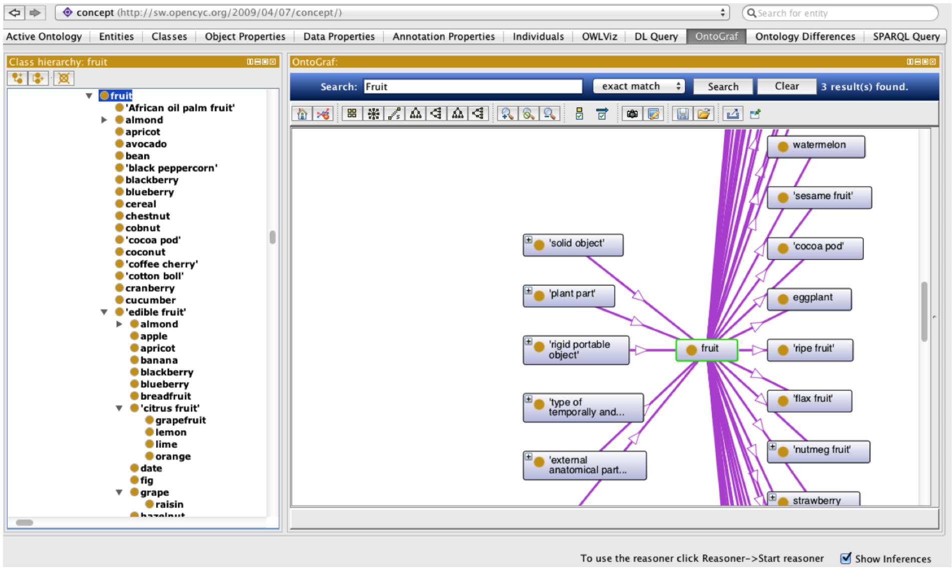
strategy among scored sentences is that we only save the sentence which has threshold value (must be set) above the set value. The consequences of this stage is sent to the following process.

5. Summary Generation Stage: The final outcome of this subsystem is a list of significant sentences that express the core meaning of the text. Therefore, this stage receives filtered sentences from past process, and then print them as a list to form a summary. Indeed, the list can be utilised to create two types of questions, first Fill-In question and second Multiple Choice Questions (MCQs). However,the research only concentrated on one type of questions that is MCQs. Furthermore, we used the list of sentences in our system as stems which illustrate question sets as well as we extracted their significant nouns as keys. The next part describes the strategy of significant nouns extraction.

6. Domain Keywords Extraction Process: Domain keywords must be specified to the system initially, these keywords actually are related to the selected domain ontology. We extracted them from the compressed text to form a set of keys, and then we will investigate their similar words using ontology similarity measure and NGD measure to produce distractors. In the case of one keyword is located in more than one sentence, we compared the significancy of these sentences using the same strategy of generating significant sentences, which elects only the sentence that has the highest value.

3.2 Ontology-based measuring approach

This section explains a set of techniques that are employed to model the ontology-based measuring approach.The ontology approach initially requires to choose an exiting ontology model from any knowledge-base engines such as OpenCyc or from exiting Web Ontology Language (OWL) files that are available on Protégé's website, therefore we select fruits topic as our entire domain. Then, we implemented the selected ontology using Java tree, after visualising the architecture of the model by Protégé. Later on, we are going to measure the distance between two certain nodes (terms) of the tree using recursive path finding algorithm, which leads us to discover the nearest terms to the key. The next parts present each phase in more details.



Search: Fruit

3 result(s) found.

To use the reasoner click Reasoner->Start reasoner

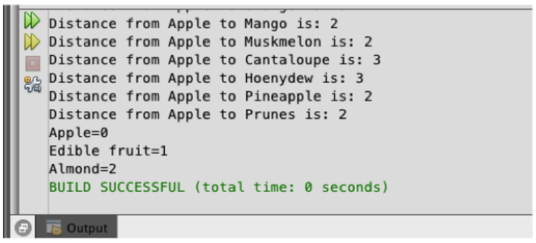
☑ Show Inferences

Figure 3.3: Protégé visualisation of the Fruit Ontology

3.2.1 OpenCyc and Protégé

OpenCyc is a version of Cyc ontology that is a large and comprehensive knowledge-base, and commonsense reasoning engine [16]. The implementation of this ontological library, provides our study with enormous amount of constant and accurate conceptual knowledge. Additionally, knowledge base allows our approach to measure ontology distance among different kinds of domains. Briefly, we selected a relevant ontology on fruits topic from OpenCyc 2009 version library after visualising the library via Protégé and then we built a tree that presents the desired fruits ontology.

Protégé is a visualisation software for Web Ontology Language (OWL) and Resource Description Framework (RDF). As it can be seen from Figure 3.3, we imported OpenCyc library file into Protégé and then we visualised the fruits ontology that to be simulated by Java tree in the following process. Additionally, the utilisation of Protégé allows us to discover a wide range of ontologies from OpenCyc library that can be used for future work.



```
Distance from Apple to Mango is: 2
Distance from Apple to Muskmelon is: 2
Distance from Apple to Cantaloupe is: 3
Distance from Apple to Hoenydew is: 3
Distance from Apple to Pineapple is: 2
Distance from Apple to Prunes is: 2
Apple=0
Edible fruit=1
Almond=2
BUILD SUCCESSFUL (total time: 0 seconds)
```

Figure 3.4: Ontology-based measure example

3.2.2 Java Tree

Java tree is used here to implement the selected OWL file, measuring distance between two nodes (terms) and discover distractors for MCQs. In accordance to the presentation of Protégé, we built a general Java tree that is similar to the architecture of fruits ontology domain. Then we measured the distance between a certain node that illustrates the key, and every single node (term) in the tree. In fact, we made use of Sirker's approach[2] that measures distance between two nodes in a binary tree, but we developed the approach to fit our ontology structure in several aspects. The structure of the tree is changed from a binary tree to a general tree, traversing from bottom to top of the tree is possible, the nodes are named rather than numbered and the distance between a specified node that presents the keyword and each node in the tree, is measured.

In Sirker's approach the recursive path finding algorithm is utilised which counts the path between desired nodes from the root of the tree. Then the size of the path between the candidate node is discovered by traversing through the path till "a mismatch"is caught. The calculation formula for this approach is:

```
First path length + Second path length - 2*common part length.
```

In the proposed system, the distance measurement process between the extracted key and each node (term) in the tree, is executed, which allows us to make a ranked list of closest terms to the key as a result. The ranking strategy here is relied on the distance for each term from the key in a descending order. Indeed, there are only two closest terms are chosen to be distractors of the question.

[2] Partha Sirker, Find distance between two nodes in a binary tree, http://www.dsalgo.com/2013/02/find-distance-between-two-nodes-in.html

3.3 Normalised Google Distance (NGD) Approach

NGD is another similarity distance measure that calculates the similarity of two words using the Google search engine. This research proposed to take advantage of the NGD algorithm to detect potential distractors that closely similar to the key. The calculation formula of NGD is defined by Cilibrasi and Vitanyi [7] and Wong et.al [17] as below:

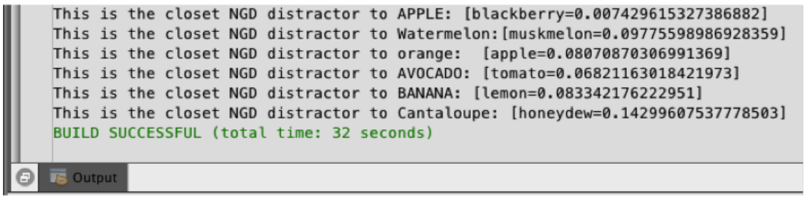
$$NGD(x,y) = \frac{G(x,y) - min\{G(x), G(y)\}}{max\{G(x), G(y)\}} = \frac{max\{\log f(x), \log f(y)\} - \log f(x,y)}{\log N - min\{\log f(x), \log f(y)\}}, \tag{3.1}$$

We obtained NGD implementation from Cassell's[3], which employed different types of similarity distance measures. Extracting NGD calculation method was a challenging task, due to the complexity of the project. Eventually, we have to understand their study and have to perform major modifications to suit our system.

There are three main functions that in the NGD calculation. The first function is the calculation of Google distance that gives the NGD distance between two words. For calculating the Normalised Google Distance(NGD), it calls the second function which is Number of results from web. It initialises the connection and finds $f(x)$,$f(y)$ and $f(x,y)$,by performing the following tasks:

1. First it makes the URL which calls the third function that is making a query URL. This function produces Google query and passes result to number of results from web function.
2. Then it calls the other function Getting count from query to find numeric value of $f(x)$,$f(y)$ and $f(x,y)$.
3. Finally the calculation of Google distance received the result from the Number of results from web to calculate NGD.

[3] Keith Cassell, An Eclipse plug-in for helping to perform the Extract Class refactoring, https://code.google.com/p/ext-c/

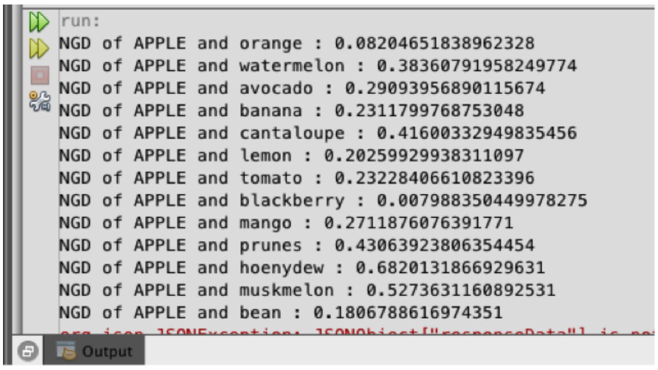


```
This is the closet NGD distractor to APPLE: [blackberry=0.007429615327386882]
This is the closet NGD distractor to Watermelon:[muskmelon=0.09775598986928359]
This is the closet NGD distractor to orange:  [apple=0.08070870306991369]
This is the closet NGD distractor to AVOCADO: [tomato=0.06821163018421973]
This is the closet NGD distractor to BANANA: [lemon=0.083342176222951]
This is the closet NGD distractor to Cantaloupe: [honeydew=0.14299607537778503]
BUILD SUCCESSFUL (total time: 32 seconds)
```

Figure 3.5: NGD measure example

The objective of utilising NGD in the study is to measure the distance between the key and each terms in the ontology. According to the result of NGD measuring process, we had another ranked list of distractors that can be compared with ontology distance's list. The ranked list elaborates normed semantic distance for each term in the range between 0 and 1, where 0 is identical and 1 is unrelated [7]. Lastly, we selected only one NGD distractor from the list, which has the lowest NGD distance to be used with the two ontology distractors.

```
This is NGD pseudo code:
1. Initialise string 1 and string 2
2. Return number of web pages that contain string 1
3. Return number of web pages that contain string 2
4. Return number of web pages that contain both string 1 and 2.
5. Select the max of log of Step 2 and log of Step 3.
6. Select the min of log of Step 2 and log of Step.
7. Use Log N as log(1.0e12).
8. Compute the Result Using NGD formula.
```



```
run:
NGD of APPLE and orange : 0.08204651838962328
NGD of APPLE and watermelon : 0.38360791958249774
NGD of APPLE and avocado : 0.29093956890115674
NGD of APPLE and banana : 0.2311799768753048
NGD of APPLE and cantaloupe : 0.41600332949835456
NGD of APPLE and lemon : 0.20259929938311097
NGD of APPLE and tomato : 0.23228406610823396
NGD of APPLE and blackberry : 0.007988350449978275
NGD of APPLE and mango : 0.2711876076391771
NGD of APPLE and prunes : 0.43063923806354454
NGD of APPLE and hoenydew : 0.6820131866929631
NGD of APPLE and muskmelon : 0.5273631160892531
NGD of APPLE and bean : 0.1806788616974351
org.json.JSONException: JSONObject["responseData"] is no
```

Figure 3.6: NGD distance for apple

CHAPTER 4

Experiment and Evaluation

4.1 Experiment Overview

The purpose of this chapter is to highlight the findings from the study of Automatic Distractor Generation in detail and in a sufficient manner, which also explain the systematic application of the methodology of the study. As described early, our system obtained sentences and keys from the text summariser. these sentences formed a set of questions, where the keys represent the correct answers. For each key, distractive plausible answers are produced from multiple similarity measurements that applies Ontology-based measure and NGD.

In previous chapter, we specified fruits as a topic and ontology domain for this research, so initially we loaded a text about fruits and health to the system in order to run the system and experiment. During this preliminary experiment, we measured cosine similarity between each key and its distractors, to investigate whether the approach generated effective distractors or not, and which similarity measure produced the highest similarity distractor that known as reasonable answer. The following sections describe the systematic experiments and results, followed by experiment analysis and evaluation.

4.2 Experiment method and results

The applied experiment method in this study, is mainly based on a quantitive test. Firstly, we conducted six sample of questions and their keys from text summariser's outputs. The proposed system produced three distractors for each candidate key, with total of 18 distractors for the selected questions, where ontology distance measure produced two distractors and NGD produced one distractor. We implemented a cosine similarity measure to examine whether the system provides high quality distractors or not. Furthermore, cosine similarity identifies which similarity measure generated the largest percentage of reasonable answers.

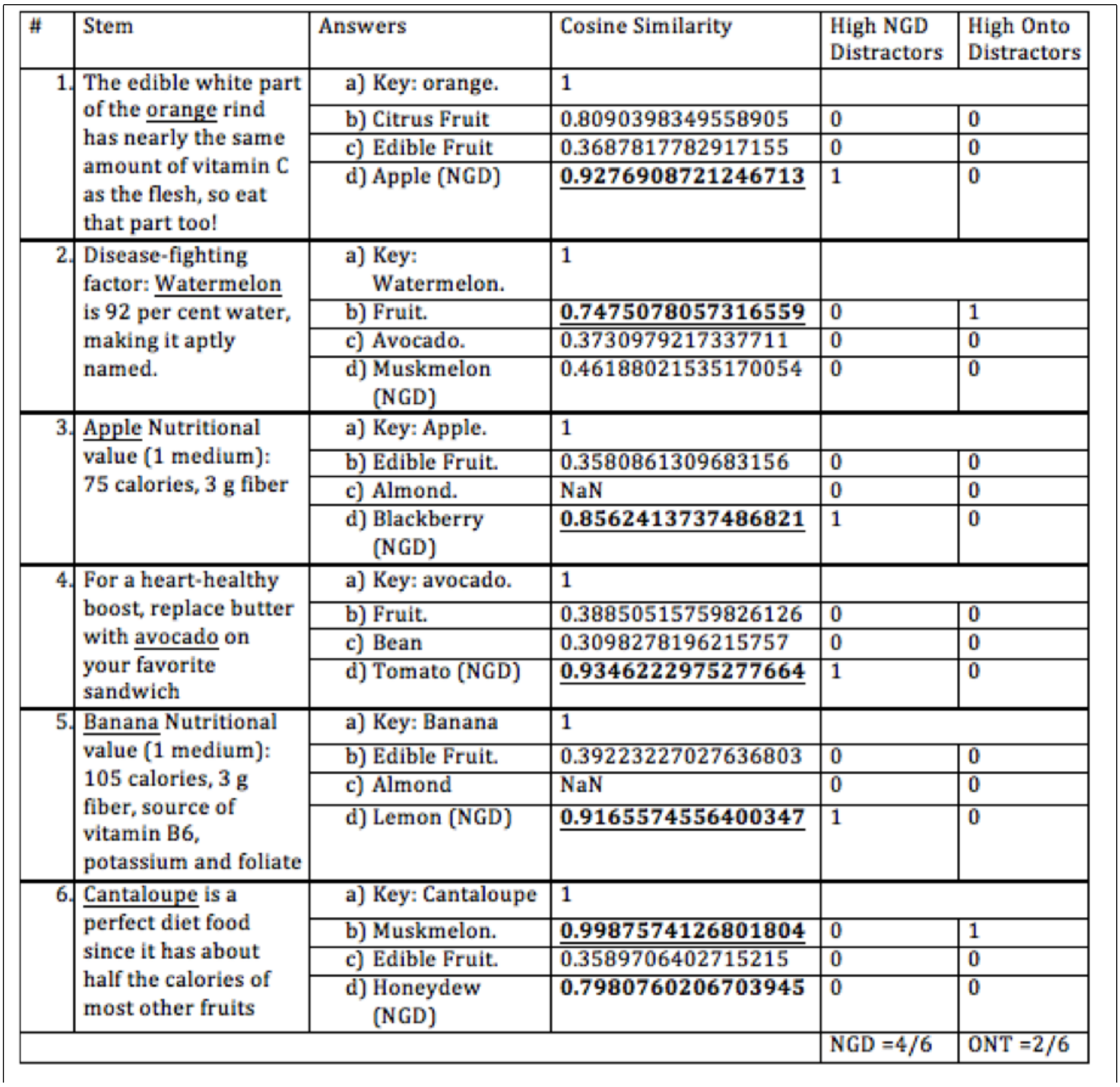
Briefly, cosine similarity measures the similarity between two terms using dot product $(a \cdot b)$. It is based on the concept of words frequency metric, where each vector represents frequency of a certain word in a set of documents. The findings of cosine similarity are ranged from 0 to 1, where 0 is dissimilar and 1 is exactly identical, also NaN result means one of the entire words does not exist in the candidate text files. The formula of cosine similarity is defined by Shirude and Kolhe [15]as:

$$Cosine\,Similarity\,(A \cdot B) = cos(\theta) = \frac{A \cdot B}{\|A\| \cdot \|B\|} \tag{4.1}$$

We applied Kurdagi's[1] cosine similarity implementation in Java with some developments. At this stage, initially we selected 8 input text files about fruits to be used in cosine similarity, and then we run the experiment of measuring cosine similarity between each candidate key and its distractors. The results of this experiment are shown in Table 4.1.

In addition to the experimental evidence that is represented in Figure 4.1, we can see that 66.66% of the reasonable answers were generated by NGD measure, where Ontology similarity measure produced 33.33% of the distractors. This means NGD is higher than ontology distance by around 33.33%, even though we only placed one NGD distractor.

[1] Sandeep Kurdagi, Cosine similarity implementation in Java, available from http://bytes4u.blogspot.com.au/2013/03/cosine-similarity-implementation-in-java.html



#	Stem	Answers	Cosine Similarity	High NGD Distractors	High Onto Distractors
1.	The edible white part of the orange rind has nearly the same amount of vitamin C as the flesh, so eat that part too!	a) Key: orange.	1		
		b) Citrus Fruit	0.8090398349558905	0	0
		c) Edible Fruit	0.3687817782917155	0	0
		d) Apple (NGD)	**0.9276908721246713**	1	0
2.	Disease-fighting factor: Watermelon is 92 per cent water, making it aptly named.	a) Key: Watermelon.	1		
		b) Fruit.	**0.7475078057316559**	0	1
		c) Avocado.	0.3730979217337711	0	0
		d) Muskmelon (NGD)	0.46188021535170054	0	0
3.	Apple Nutritional value (1 medium): 75 calories, 3 g fiber	a) Key: Apple.	1		
		b) Edible Fruit.	0.3580861309683156	0	0
		c) Almond.	NaN	0	0
		d) Blackberry (NGD)	**0.8562413737486821**	1	0
4.	For a heart-healthy boost, replace butter with avocado on your favorite sandwich	a) Key: avocado.	1		
		b) Fruit.	0.38850515759826126	0	0
		c) Bean	0.3098278196215757	0	0
		d) Tomato (NGD)	**0.9346222975277664**	1	0
5.	Banana Nutritional value (1 medium): 105 calories, 3 g fiber, source of vitamin B6, potassium and foliate	a) Key: Banana	1		
		b) Edible Fruit.	0.39223227027636803	0	0
		c) Almond	NaN	0	0
		d) Lemon (NGD)	**0.9165574556400347**	1	0
6.	Cantaloupe is a perfect diet food since it has about half the calories of most other fruits	a) Key: Cantaloupe	1		
		b) Muskmelon.	**0.9987574126801804**	0	1
		c) Edible Fruit.	0.3589706402715215	0	0
		d) Honeydew (NGD)	**0.7980760206703945**	0	0
				NGD =4/6	ONT =2/6

Table 4.1: Cosine Similarity experiment

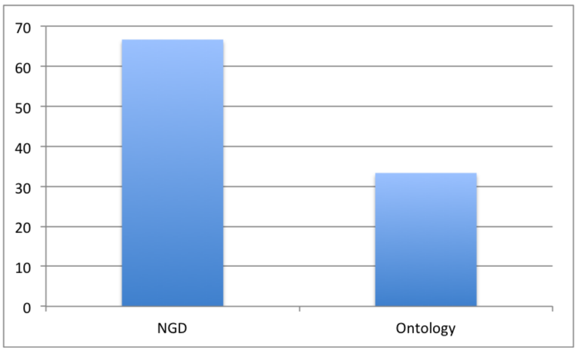


Figure 4.1: Difference between NGD and Ontology distractors in terms of generating reasonable answer

4.3 Experiment Analysis and Evaluation

The involvement of cosine similarity measure clearly led to the fact that the proposed approach produced very effective and high quality distractors. For example, from Table 4.1, we can see that for each investigated question there is at least one or two distractors are very close to 1, which means they are very close to be identical to their keys. In other words, there is a good percentage of reasonable answers generated by the desired study, around 44% of all answers. In the area of MCQs, reasonable answer means an answer that can assess examinees' understanding of the desired information without unduly being too easily detectable [12].

Another important consideration on this study is that, in accordance to the finding results, we realised that the number of NGD distractors should be increased once. Therefore, it possible that the more NGD distractors in the question, the more effective or reasonable distractors likely are. However, connecting to Google search engine has some restrictions, causing some difficulties to obtain results. Google allows very limited number of queries for every IP address per day, and around six queries per 15 minutes. We resolved this problem by making our own NGD database on the system after collecting required data from the Google search engine. Moreover, the results from NGD is changeable from time to time, due to the reliance on Google web page counts. There is an alternative solution that uses Yahoo search engine, but it is costly and not free any more.

In regard to the small percentage of ontology distractors, we considered this lack may due to the accuracy of fruits ontology. In fact, there is a need to enhance the number of ontology library to provide very strong knowledge base. Additionally, we noticed that the implementation of Java tree does not give efficient results for all cases, thus we suggest implementing Java graph for further development on this research. Although cosine similarity reflected reasonable evaluation results, the involvement of expert opinions should be considered. Conducting a survey among language experts and scientists, provides more accurate analysis and evaluation results.

CHAPTER 5

Conclusion

5.1 Overview

This chapter reviews briefly prime aspects of the research of automated distractors generation for MCQs using multiple similarity measurements. The main proposed objective of the study is to produce meaningful and effective distractors, which have the ability to assess learners knowledge for a certain concept efficiently and clearly.

In order to reach this aim, we decided to select two effective similarity measures that are ontology based measures and Normalised Google Distance (NGD) measures. Each one has a strong capability to measure the similarity between two candidate words in different ways. For instance, NGD measures the similarity distance by utilising web search engine, whereas the implemented ontology in this study used general tree in Java. The next sections gives a short review of main points of our novel approach and some suggestions for future work.

5.2 Summary and contribution

The proposed project consists of three combined approaches to generate distractors automatically. The first approach is the automated text summariser, which produced two documents, a summary of the entire text file and a set of significant sentences from the summary after specifying topic's keywords. These outputs can be used to form two types of question that are multiple choice questions and fill-in-the-blank questions. This study used these outputs to generate the first type of question, and then attempted to make distractors for domain keywords using ontology and NGD measures. Basically, text summariser has six steps to produce a compressed text. The text preprocessing step takes place initially, performing three process that are converting words to lower case state, word stemming that removes words' suffixes, and stop words removal. After this, the generation of terms value start assigning values for each term, based on word frequency metric. Then sentence value calculation computes the value for each sentence, followed by the stage of extracting the significant sentences that have the highest values to be listed on summary generation stage. Finally, our system extracted the sentence that contains the desired domain keywords from the summarised text to constitute the question sets.

In the ontology-based measure, we built a Java tree that represents the fruits ontology. We made use of OpenCyc library and Protégé in order to display the desired ontology. Each node of the tree indicates one name of fruits, thus we can measure distance between nodes (fruits' terms) on the tree. The project employed Siker's approach which implements the concept of lowest common ancestor. The approach finds distance between the key that the summariser specified early, and every node. We ranked the resulted terms in descending order, so the system select two nodes that have the closest distance. These two terms formed two distractors to the key.

NGD is another similarity measure implemented by our approach. At this stage, Google search engine was chosen to be used for searching about two candidate words on the web. Furthermore, NGD returns results of similarity between these word on the range between 0 (identical) and 1 (dissimilar). Every single term on the domain ontology is measured with the key, then we ranked them to select only one word that has the closet distance to be the third distractor of the question.

In terms of evaluating the study, the cosine similarity was executed to examine the effectiveness of the generated distractors and determine the best performance of the two similarity measures. During experiment phase, six questions were taken to run cosine similarity assessment. Each resulted distractor was remeasured, with aim of discovering the reasonable answer.

The findings from the experiment results revealed two essential points: firstly, our approach generated at least one reasonable answer, which means the produced distractors are plausible answers. Secondly, NGD had better performance than ontology-based measure, due to the lack of designing fruits ontology.

5.3 Potential Future Work

Due to the study combined three different systems, it has a lot of rooms for future work. In text summariser there is a need to enhance its results by combining another sentence scoring metric to word frequency metric such as sentence location metric. Additionally, Part of Speech (POS) tagging may be used to identify significant nouns of the summary rather than extracting domain keywords.

Another possible future development is to employ Java graph rather than tree, which provides more flexibility and accuracy to the study. Also the structure of fruit ontology does not seem to be comprehensive enough, thus further enhancement on Opencyc library should be preformed.

The reliance on Google search engine for NGD measure, unexpectedly affects the efficiency of the project. Therefore, another search engine API can be used such as Yahoo or Bing. The results of NGD measure are unreliable as a result of using web page counts that are indeed changeable.

العنوان:	Automatic Distractor Generation Using Multiple Similarity Measurements for Multiple Choice Questions
المؤلف الرئيسي:	Saeed, Wael Saeed
مؤلفين آخرين:	Liu, Wei(Super.)
التاريخ الميلادي:	2013
موقع:	سيدني
الصفحات:	1 - 36
رقم MD:	615570
نوع المحتوى:	رسائل جامعية
اللغة:	English
الدرجة العلمية:	رسالة ماجستير
الجامعة:	Western Australia University
الكلية:	School of Computer Science and Software Engineering
الدولة:	أستراليا
قواعد المعلومات:	Dissertations
مواضيع:	الجاسبات الإلكترونية ، هندسة البرمجيات، أسئلة الاختبارات
رابط:	https://search.mandumah.com/Record/615570

Abstract

Self-learning becomes a vey essential type of learning in recent years, requiring further developments on current electronic learning technologies. In fact,there is a need to enhance present self-assessment tools as a key constituent of self-learning. Multiple choice question is one of the most common and popular assessment in the discipline of learning, however creating such an assessment manually is costly and time consuming. The developments in the field of texting mining and natural language process, increase the possibility of generating multiple choice questions (MCQs) automatically. A number of studies have conducted on the systematic generation of MCQs on based on different criteria. Similarity measures is the most recommended one, in particular ontology-based measures and Normalised Google Distance (NGD) measures. In spite of each measure has its own potential strengths and weaknesses, the combination of the two is a possible way to achieve more effective and efficient MCQs generation system. The creation of meaningful distractors for MCQs is a measure to evaluate the strength of the system. In this paper, we designed an approach that generates distractors through the integration of two similarity measures, NGD measures and Ontology-based measures. They generated different set of distractors after obtaining questions from an automated text summariser. We employed the Google search engine to generate NGD distractors, whereas Java tree produced ontology distractors. In regard to text summariser, it was built based on words frequency metric, which generated a list of significant sentences that can be used for MCQs or fill-in-the blank questions. Our preliminary evaluation applied cosine similarity measure to investigate the effectiveness of our approach. The findings of the evaluation demonstrates that the proposed system generates effective and reasonable distractors.

Keywords: Text Mining, Automated Multiple Choice Questions, Automated Text Summariser, Ontology-Based Measures, Normalised Google Distance, Cosine Similarity Measure
CR Categories: A.2, I.7.2

Contents

List of Tables

List of Figures

العنوان:	Automatic Distractor Generation Using Multiple Similarity Measurements for Multiple Choice Questions
المؤلف الرئيسي:	Saeed, Wael Saeed
مؤلفين آخرين:	Liu, Wei(Super.)
التاريخ الميلادي:	2013
موقع:	سيدني
الصفحات:	1 - 36
رقم MD:	615570
نوع المحتوى:	رسائل جامعية
اللغة:	English
الدرجة العلمية:	رسالة ماجستير
الجامعة:	Western Australia University
الكلية:	School of Computer Science and Software Engineering
الدولة:	أستراليا
قواعد المعلومات:	Dissertations
مواضيع:	الجاسبات الإلكترونية ، هندسة البرمجيات، أسئلة الاختبارات
رابط:	https://search.mandumah.com/Record/615570

Automatic Distractor Generation Using Multiple Similarity Measurements for Multiple Choice Questions

The University of Western Australia
School of Computer Science and Software Engineering

Supervisor: Asst/Prof. Wei Liu
Author: Wael Saeed Saeed
STD No. 20471262

This report is submitted as partial fulfilment
of the requirements for the Master of Computer Science of the
School of Computer Science and Software Engineering,
The University of Western Australia,
2013

Abstract

Self-learning becomes a vey essential type of learning in recent years, requiring further developments on current electronic learning technologies. In fact,there is a need to enhance present self-assessment tools as a key constituent of self-learning. Multiple choice question is one of the most common and popular assessment in the discipline of learning, however creating such an assessment manually is costly and time consuming. The developments in the field of texting mining and natural language process, increase the possibility of generating multiple choice questions (MCQs) automatically. A number of studies have conducted on the systematic generation of MCQs on based on different criteria. Similarity measures is the most recommended one, in particular ontology-based measures and Normalised Google Distance (NGD) measures. In spite of each measure has its own potential strengths and weaknesses, the combination of the two is a possible way to achieve more effective and efficient MCQs generation system. The creation of meaningful distractors for MCQs is a measure to evaluate the strength of the system. In this paper, we designed an approach that generates distractors through the integration of two similarity measures, NGD measures and Ontology-based measures. They generated different set of distractors after obtaining questions from an automated text summariser. We employed the Google search engine to generate NGD distractors, whereas Java tree produced ontology distractors. In regard to text summariser, it was built based on words frequency metric, which generated a list of significant sentences that can be used for MCQs or fill-in-the blank questions. Our preliminary evaluation applied cosine similarity measure to investigate the effectiveness of our approach. The findings of the evaluation demonstrates that the proposed system generates effective and reasonable distractors.

Keywords: Text Mining, Automated Multiple Choice Questions, Automated Text Summariser, Ontology-Based Measures, Normalised Google Distance, Cosine Similarity Measure
CR Categories: A.2, I.7.2

Acknowledgements

This dissertation would never be achieved successfully without the encouragement and support from many awesome people. First, and foremost, I would like to acknowledge with much appreciation and deep thank to Professor Wei Lui, for her guidance, unstoppable enthusiasm, patient and kindness throughout the project.

I would like to express my special thanks to my wife Manal Abdullah, my brother Waleed Saeed and my family in Saudi Arabia, who provided me with great love, caring and inspiration during difficult times in my life.

A special gratitude to all of my friends, school lecturers and professors for their contributions of enhancing my academic career.

Contents

List of Tables

List of Figures

CHAPTER 1

Introduction

1.1 Overview

Automated systems for generating multiple choice questions (MCQ) is a very useful technique of consideration in the creation of assessment materials and tools, not only for e-learning systems in educational departments but also for lifelong learning. This is because they provide a self-assessment and analytical evaluation tools to the users, together with immediate feedback which gives the users the opportunity to find out whether the answers submitted are correct or not in order to measure their learning progresses.

The high demand that has been experienced in the implementation of self-assessment tools, has thus created the necessity for the development of automated multiple choice question generators. Educators propose to use it as a system to ease the generation of multiple choice questions, gauge the learning progress of students and provide a mean of reasonable assessment. An automated system for the generation of questions can greatly reduce time, cost and effort that would otherwise be required in manual question setting.

A multiple choice question consists of a sentence stem representing the question which the student is supposed to answer, together with three to four answers, only one of which is the correct one and the rest serving as distractors. Though reasonable evaluation results have been received from the multiple choice questions, their successful effectiveness is inherently connected to the quality, strength and effectiveness of the distracters that are associated with the questions. Effective or reasonable distractors in multiple choice questions have the capability of assessing if the students have a clear understanding of the concept, without unduly presenting too much pressure or being too easily detectable [12].

Papasalouros et.al (2008) [14] generated multiple choice questions automatically by utilising three entities; a knowledge base containing facts pertaining to a specific domain, semantic relationships connecting entities in the knowledge base, and a natural language generation component. Until recently, multiple-choice questions were churned out by applying term extraction, semantic distance calculation and sentence restructuring method on ontologies like WordNet.

According to Papasalouros et.al (2008) [14], generating questions using natural language generation methods on a knowledge base developed using Ontology Web Language (OWL) would be a more effective way to generate multiple choice questions. Utilising OWL can prove to be more efficient, as it is domain-independent and helps access multiple domain ontologies. As the entities are arranged based on class hierarchies, it becomes easier to generate distractors. Moreover, entities are categorised based on their properties, which helps to automatically determine relationships between entities as well as determine data type of an entity. Thus, the process of acquiring correct answers and phrasing distracting options is simplified. However, a problem arises when the same word can mean different things, under different contexts. Such words are often referred to as ambiguous words.

In accordance to Bollegala et.al (2007) [6], semantic similarity can be measured, by using Internet search engines to determine the most plausible and relevant context for a word, using "page count"as a statistical measure. This measure basically analyses and ranks the different contexts in which a word has been searched. To determine the semantic similarity between two words, the individual page count of the two words as well as the page counts of the two words combined can be used. Thus, the word relevant to the context can be determined and used while framing distractors in multiple-choice questions.

Another approach to formulating multiple-choice question is to gather sentences from questions mentioned in a specified set of learning materials, employing a statistical technique to generate a blank part for the question, and create distractors using grammatical and statistical patterns [1]. While this approach may be useful in generating multiple-choice questions for a specified text, it is not as flexible or powerful as the method recommended by Papasalouros et.al (2008) [14].

1.2 Project Objectives and Methodology

In this regards, the study seeks to carry out an analysis of the effectiveness of automated multiple choice question generation and the quality of the distractors that are associated with the questions. It is thus geared towards the generation of effective and high quality distractors through the use of multiple similarity measurements for automatic multiple choice question generation.

This research concentrates on the generation of effective distracters that allow for accurate progress assessment through the implementation of two main similarity measures which are Ontology Distance measures and Normalised Google Distance(NGD) measures. First we make use of automated text summarisation system that analyses the entire text corpus to generate a set of significant sentences and a set of significant keywords. According to the results of the summariser we will be able to generate closest distractors to the desired key by the combination measure of NGD and ontology distance. In other words, our system emploies approaches such as text summariser, NGD and ontology distance to find similar distinct word associated with a query keyword, with the hope to give a set of high quality and effective distractors as a result.

1.3 Dissertation Outline

The remainder of this research dissertation is structured in the following way: Chapter 2 reviews the literature concerning the usefulness of using MCQs and some fundamental background information. Chapter 3 seeks to address the design and methodology of the proposed system, and chapter 4 explains the results of distractors generation system evaluation. The last Chapter 5 concludes the paper with reviewing system's findings and indicating suggestions for future work.

CHAPTER 2

Literature Review

2.1 Overview

Alsubait et al [4] categorised the learning assessment items into two formats:

- First subjective assessment such as short/long answer questions, essays and research paper, which require a lot input from the users and markers.
- The second format is an objective assessment using multiple-choice questions (MCQ) as well as questions that require selective answers.

They argued that objective tests are much harder to generate, often not well-structured, requires a considerable amount of time in their preparation and analysis of relevance as compared to subjective essays. However, they require minimal supervision from the examiners, are more reliable because of marks do not rely on examiners opinions and reflects obvious differentiations among examinees, and can be used in evaluating a wide range of knowledge from a vast amount of information resources. In MCQ assessment, the marking process is much easier than other types of learning assessments and very scalable to large classes. In addition, marks for the questions are easily calculated, with the objectivity of the activity being guaranteed by the fact that the feedback is expected by the users, which can easily be converted into statistical figures for further representation and analysis of the assessment results.

Recent developments in e-learning systems lead to offer abundance of electronic textual resources (e.g ebook, web documents, lecture notes, journals, e-class). While having these e-textual resources presents a myriad of advantages to learners and instructors , there is a need to use an electronic or automatic assessment as a key constituent of E-learning. In other words, the measurement towards students capabilities is significantly required by using e-assessment techniques. However, the reliability of e-assessment systems in such environment is critically essential.

They must be well constructed to identify whether or not examinees have achieved desired objectives successfully.

Multiple-choice questions thus become one of the most popular measures for assessing students in a given learning environment. However, creating multiple-choice question is a time consuming, difficult task and requires expertise. Given the current advances in the field of natural language processing and text mining, powered by the advance of electronic textual resources, it is quite feasible to create such a system capable of automatically generating MCQs, in a bid to reduce time and effort in manually creating such questions.

2.2 Automated Text Summarisation

Automated Text Summarisation is a new technique to automatically summarise original text by extracting the most significant sentences or keyphrases that convey the fundamental meaning of the input document. In comparison with manual summarisation methods, it reduces time and effort on making such document. According to Al-Hashemi [2] there are two main types of text summarisers namely extractive and abstractive summariser. The extractive one is commonly used rather than the abstractive method due to the lack of attempt in producing a summary as much similar as a “human”summary. Furthermore, this popular type of text summariser is mainly based on the concept of information extraction, which extracts the main terms or sentences in a desired text in order to formulate a compressed text [2].

Al-Hashemi [2] followed four phases in his proposed study to design the system that summarises a candidate text using a keywords extraction strategy. The author started with a text preprocessing phase that consists of subprocesses which are firstly restructuring an unstructured text file, stop words removal process, word tagging and stemming. The outputs of the first phase is a set of important keywords that is used in the second phase which is a significant keywords/phrases extraction process. There are several ranking strategies implemented by Al-Hashemi to measure the importance of each word, the frequency of a candidate word in the document, Inverse Document Frequency (IDF) and word location in the text. In his article, he stated that words that have high weight usually exist in the title/headings of the document, have a capital letter and a different font type. However, the third phase describes sentence selection strategy that ranks each sentence in accordance to a set of metrics such as sentence location in both paragraph and text, the length of the candidate sentence and presence of the significant extracted keywords/phrases in the sentence. In the final phrase, the implementation of TFIDF measure is involved to minimising or “filtering”the number of significant sentence and to give more quality to the resulted summary. In terms of system evaluation, the author concluded that the finding of performing Precision and Recall measurements on the system, illustrates 70% overall precision which means the system generated a high quality summary [2].

2.3 Automatic multiple-choice questions Generation

A multiple-choice question consists of items that present short sentences, stating particularly a single question or proposition, and a set of choices (e.g four short answers). A student needs to select the one(s) from these answers. In this research, we focus on MCQs that have a single correct answer. This is the simplest case for automaton the process of MCQ creation. Only one of the answers is the correct answer which is called the key. The rest of the answers are wrong and referred to as distractors. Alsubait et al [4] explained the main structure of multiple-choice questions in more detail. A multiple-choice question has two components: a short sentence (stem), demonstrating a certain problem or question that is extracted from a certain text. A number of options from which the user chooses the correct answer. This option set can be represented as $A = \{a_i \mid 2 \leq i \leq max\}$. This implies that we need at least two answers, one being the distractor. The upper bound is not specified, which measure that we can have as many distractors as the system sees suits. This can further be subdivided as a set of correct options and a set of incorrect answers. The set of correct options can be known as keys (K) depending on the type and number of solutions of the question being asked, which can be represented mathematically as, $K = \{K_m \mid 1 \leq m \leq i\}$. However, the set of incorrect answers that act as distractors (it will be explained in more details in next section) to the student, thus working towards the determination of their academic acuity and knowledge. This can be represented as $D = \{D_n \mid n\colon = i - m\}$ [4].

Furthermore, in their article [4], the explanation of two primary types of similarity are sufficiently represented, which are *semantic* and *relational* similarities. The relational similarity highlights similarities between concepts in terms of their relations. The main advantage of the relational similarity is that it generates multiple-choice questions that tests the higher level of “cognitive abilities”of students.The semantic similarity has the ability to produce a plausible distractor for a set of question, which is critically important for generation multiple-choice questions [4].

2.4 Characteristics of High Quality Distractors

As has been mentioned early in this paper, the multiple choice questions comprise statements of the question, otherwise referred to as the stem of the question, followed by a series of multiple-choice solutions to the question with only one of the answers being the correct solution to the question. The correct solution is referred to as the key of the series with the incorrect solutions being referred to as the distractors. Karamanis et al [11] defined an appropriate distractor as "a concept semantically close to the anchor [key] which, however, cannot serve as the right answer itself".

An important consideration in the automatic generation of distractors for multiple choice questions is the number of distractors used. The more distractors used, the less the chance of getting the correct answer through guesses. The larger number of distractors, the greater possibility of ambiguity and implausibility of the solutions, presented to the students [13]. An effective distractor plays a major role in generating a high quality automatic multiple-choice questions,however it is also "a major challenge in preparing MCQs" [4]. A distractor must be closely similar to the true answer (the key), to highlight students who have not achieved the required level of experience, abilities and knowledge in the learning subject being examined. Goto et.al (2010) [1] stated that effective distractors should have similar part of speech to generate a reasonable question.These features are significant in the process of creating correct educational assessments. Mostow et al [13] introduced three classes of distractors which are: Ungrammatical, Nonsensical and Plausible distractors. The main effective type is the plausible distractor, because of the main features that it has such as the close similarity to the key in terms of part of speech (POS) and it forms a plausible sentence. This literature reports on number of recent studies exploring the generation of distractors for multiple choice questions, particularly those applied Normalised Google Distance (NGD) [7, 17] which is explained in section 4, and ontology concept [5, 3, 14, 10] that is presented in section 5.

2.5 Normalised Google Distance

A considerable amount of literature has been published on measuring semantic similarity between words and phrases based on web search engines such as Google, Bing and Yahoo. In this study, we will examine whether Google distance can be used to measure similarities between a potential distractor and the key for MCQs. Normalised Google Distance (NGD) is used Google search engine to calculate the similarities between objects based on their names [17]. In accordance to Wong et.al(2007) [17], NGD can be calculated as following:

$$NGD(x,y) = \frac{G(x,y) - min\{G(x),G(y)\}}{max\{G(x),G(y)\}} \tag{2.1}$$

where

$$G(x) = \log\frac{1}{g(x)} \qquad \text{and} \qquad G(x,y) = \log\frac{1}{g(x,y)} \tag{2.2}$$

The $g(x)$ and $g(x,y)$ are the probability of existence of terms x and y. They can be calculated as

$$g(x) = \frac{|x|}{N}, \qquad g(x,y) = \frac{|x \cap y|}{N}, \tag{2.3}$$

where y is the number of Google pages that contain term y and x is the number of Google pages with in term x. $|x \cap y|$ is the number of Google pages contains both terms x and y, also N is defined as $N = |x| + |x \cap y|$.

The authoring environment that is introduced by the system of Cilibrasi and Vitanyi [7] demonstrated the utilisation of Google search engine to facilitate its approach. Cilibrasi and Vitanyi (2007) [7] stated the construct of Google semantics of words as a number of web pages returned by the desired query. Additionally, in their research, they represented the theory behind (NGD) that it is a result of Google-based estimation of the Normalised Information Distance (NID). Where the utilisation of Google code-word length allowed them to estimate the NID with Normalised Compression Distance (NCD) and using Google distribution as a compressor for Google semantics [7].

Similarly, Wong et.al [17] applied Google distance as a gauge for similitude and term length, together with new Tree-Traversing Aunt (TTA) algorithm. In their paper they pointed that for term clustering TTA operates on a two-stage approach; first stage the nodes are divided into sub-nodes by TTAs, and then terms are repositioned in order to attaine ideal clusters using NGD. The findings of this approach are promising. Wong, et al. (2007) highlighted that NGD significantly relies on Google search engine to maintain the capability of measuring the similarity and distance between words at the level of compression. Wilson, et al. (2007) asserted that the innovative application of featureless similarity on the basis of NGD and Wikipedia , exhibited brilliant results, additionally they pointed a set of advantages of this approach such as the ability to detect outliers, generate consistent outputs and identify concealed structures of clusters [17]. On the basis of the studies above which applied NGD in their approaches, we can obtain an approach that uses NGD to measure similarity distance between the key and distractors.

2.6 Ontology Based Measures

In recent years, there has been an increasing amount of literature on the generation of distractors for automated multiple-choice questions. A number of studies have conducted a series of generation MCQs using ontology. According to Al-Yahya [3] ontologies are defined as “Knowledge representation structures which provide a conceptual model of the domain”. The system developed by Papasalouros et al. [14] aims at developing a novel approach for creating multiple optional questions automatically, based on explicit ontologies domain and some linguistic resources. They [14] propose this formulated approach to automate the entire process of assessment, to provide dependable evaluation process.

The tool developed receives an input ontology and produces as output multiple choice questionnaires. The study [14] outlines ways through which some domain ontologies can be used as inputs for questionnaire creation in the education setting. This included domain manual summarisation by both ontology engineers and pedagogic experts, domain manual summarisation in concept map order, and ontology generation automation. Moreover, for education purposes the paper recommends the reuse of the domain ontologies created by a field expert.

Furthermore, the article [14] indicates definite ontology related routine that developing such a system needs to follow. In addition, They use alphabets as the routines for instance; A, B, C, D are used as concepts names, R, S are used as roles names and a, b, c are used as individuals names. Consequently, by use of routines, some strategies were formulated to choose the right answers, and select the distractors. In this regard, the proposed strategies are only concentrated on the semantic aspects in the generation of ontology-based questions process. These strategies include:

- Class-based strategies, which the creation of distractors is dependant on their individuals and classes.
- Property-based strategies; which generate a set of distractors according to their roles.
- Terminology-based strategies contain strategies based on relationships without directly involving individual ontology [14].

To evaluate this approach, Papasalouros et al. [14], utilised five different domain ontologies such as Eupalineio tunnel ontology to examine the system. Afterwards, the set of resulted questions was evaluated by a three-dimension perspective, in terms of pedagogical quantity, syntactical correctness, and number of questions generated. Domain experts found all questions generated from this approach passable for assessment. However, the proposed approach is good at defining of questions semantics, but it offers little in creating syntactically correct questions thus leaving room for future work. Moreover, they suggested that the utilisation of online search engine such as Google, will assist in overcoming the weakness of domain ontologies in future.

Similarly, the work described by Al-Yahya [3] applies ontologies to generate a set of learning assessments. They have explored a system that generate MCQs using an OntoQue engine, from a domain ontology. The OntoQue generates assessment items by iterating through entities in the ontology and implementing the Jena API. The researcher classified the strategy of the study into three categories; class-membership strategy, individual strategy and property strategy. In fact, the author has used the same strategies that Papasalouros et al. [14] have applied. However, she implemented individual based strategy instead of terminology-based strategies, with aim to create Fill-In (FI) items. During the strategy of class-membership, the process of generating distractors applied a random model from classes. The researcher pointed that the approach showed a perfect performance in generating such an assessment, however, it needs improvements in terms of wording using WorldNet and the analysis process should rely on real use cases [3].

Another study of ontology based MCQ is carried out by Bin et al (2009) [5], that conducted ontology-based measure of semantic similarity between concepts. From what has been published in the paper, it is obvious that the implementation of semantic similarity is an essential consideration in knowledge sharing, web mining and MCQs generating.

The problem being solved in the research, however, is centred on the fact that, most studies tend to focus on the measurements of semantic similarity between words rather than concepts, whereas, semantic distance between concepts is the fundamental in this matter. Therefore, Bin et al (2009) [5] discussed two traditional measures of semantics in their work. These are graph based measure and information content based measure. Graph based measure investigates mainly the length and depth between concepts. On the other hand, information content based measure is based on the perception that the illustration of semantic distance between concepts, must be delivered accurately by information content.

The study carried out by Bin et al. (2009) [5] tries to understand the semantic similarities that exist between traditional measures.To overcome problems associated with both measures, they have developed an idea that combines graph based measure and information content based measure as a new measure that known as Ontology Hierarchy Information and Information Content semantic similarity measure (OHIIC) [5].

With the combination of the two measures, values produced between their concepts indicates the relationship that exists between them. Although the new idea is only based on theoretical analysis, it helped to achieve results that both measures could not achieve when used separately. However, to achieve their final results, Bin et al (2009) were required to construct a Concept Tree (CT) from WordNet. The approach was examined against 28 word pairs. To evaluate the success of the new measure, the results produced using OHIIC were compared to those of other measures. The results obtained in the study shows that, in semantic similarity measures, Ontology Hierarchy Information is a significant consideration [5].

Based on ontology distance measure, Jing et.al (2006) [10] explore a new clustering technique that enhances the performance of text clustering. Indeed, this approach can be implemented in distractors generation for MCQs.Therefore, the main reason for producing the study is to develop a new clustering scheme based on the measure of ontology distance. However, as the authors assert it was essential to calculate the term mutual information matrix. This was done with the aid of some methods and technique such as WordNet and other ontology methods.

They have tried to resolve most difficult problems caused by text clustering in text documents. As far as they are concerned, text clustering is a challenging problem when it comes to critical information volumes, complex semantics and high dimensionality. To resolve the problems associated with complex semantics, they have proposed the utilisation of the existing learning ontology techniques, with aid of WordNet in order to calculate the term mutual information (TMI) [10].

Furthermore, they designed a new data model that combines mutual information matrix (MIM) and traditional vector space model (VSM), to evaluate their system. The new model designed considers relationship that exists between terms. With new ontology-based distance measure, the research [10] employed two k-means type clustering algorithms, the standard k-means and the FW-KMeans. The main reason why the authors had to employ these algorithms is due to the fact that k-means algorithms are “efficient” as well as “scalable”. The obtained findings from the study demonstrated that, the two clustering algorithms have performed better progress when using ontology distance [10].

2.7 Conclusion

This literature paper presents a number of related research to automatic distractor generation using multiple similarity measurement for MCQs. We have reviewed studies that employed automated text summarisation, Normalised Google Distance and ontology based measure in their approaches. In regard to text summariser, we have covered some aspects in generating such system like the preprocessing task, keywords extraction, metrics for sentence selection and ranking words/sentences strategies. On the basis of NGD research, we will investigate an approach that uses NGD to measure similarity between distractors and the key for MCQs. Moreover, the studies of ontology-based measure support us with useful explanation of various types of ontology measures such as graph-based measure, information content-based measure and Ontology Hierarchy Information and Information Content semantic similarity measure. In addition, some ontology strategies such as class-based strategy and property based strategy, were presented to explain the process of generating distractors and keys in MCQs. We will apply both NGD and Ontology as multiple similarity measurements to produce distractors automatically for multiple choice questions.

CHAPTER 3

Methodology

3.1 Overview: The proposed System Architecture

In this chapter, the methodology we used to generate MCQs distractors, will be described in detail. The following sections introduce the implementation of the automated distractors generation system, which combines three different approaches (subsystems) as it can be seen from Figure 3.1. Firstly, the automated text summariser generates a set of significant sentences (stems) from the input corpus, and extracts domain keywords (keys) from these sentences, which allows the desired system to create two possible types of questions Fill-In (FI) and MCQs. Secondly, Ontology-based measuring approach provides the study with a list of similar words to the key, and Normalised Google Distance (NGD) presents another list of closest words to the key as well. We selected two nearest ontology terms and one NDG distractor from the finding lists, in order to compose a combined set of distractors.

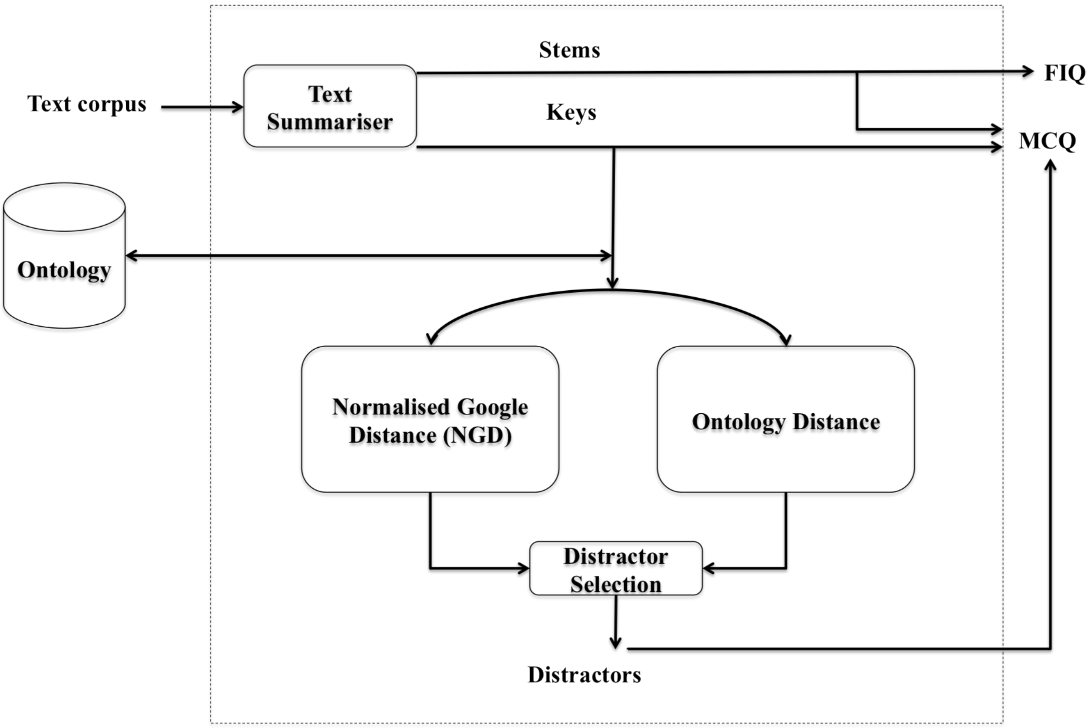


Figure 3.1: The proposed System Architecture

3.1.1 Automated Text Summarisation

Basically, the process of this technique involves three phases as Hovy [9] stated which are topic identification, interpretation and summary generation. During the first phase a type of metrics for scoring sentences must be specified. Additionally, Hovy [9] introduced six types of criteria that can be used in text summarisation: sentence position in the text, Cue phase indicator, Word frequency-based measure, Query and title overlap metric, Cohesive (lexical) connectedness metric and Discourse structure metric. There is no obvious best metric for scoring sentences, however some metrics perform better when it is used in a particular genre. For instance, it is recommended to implement positional criteria in newspaper articles. This is due to the structure of the newspaper which requires to locate the important information in certain parts such as titles or first paragraphs [9].

The approach that we implemented was obtained from the University of Waterloo in Canada which adopts an extractive summarisation method. It was assigned to student as a group project assignment in Computational Linguistics course (CS784) in Spring, 2013 [8]. We employed the starter code that was provided in a Zip file on the course's website. With further developments, implemented another word frequency method from Learner's approach[1] rather than the provided code. The model consists of five stages as shown in Figure 3.2, which are text preprocessing stage, generating terms' values stage that is based on word-frequency metric, calculation of all sentences' scores, significant sentences generation stage and lastly summary generation stage. The resulted significant sentences are used later on to form question sets (stems) from which the domain keywords are also extracted.

[1] Learner, Word Frequency Counter, available from: http://javabycode.blogspot.com.au/2010/12/word-frequency-counter.html

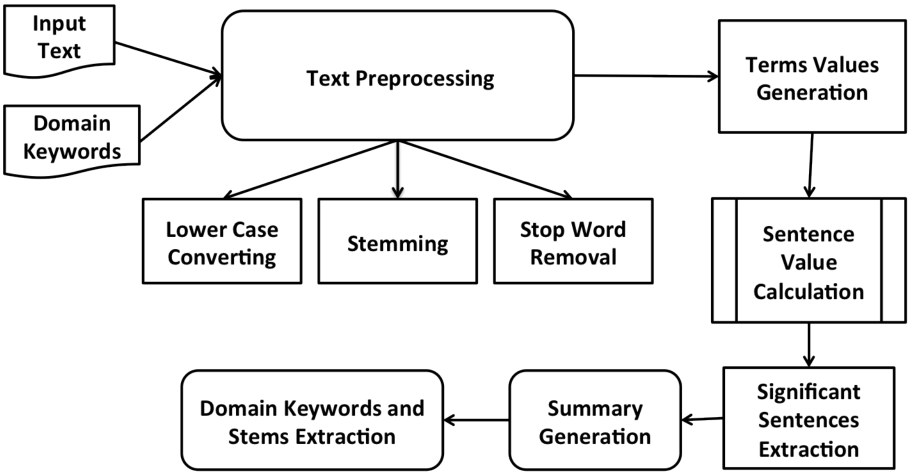


Figure 3.2: The Structure of A Text Summariser

Text Summariser Architecture:

An automated text summariser constructs of the following six stages as mentioned earlier:

1. Text Preprocessing Stage: This stage is essential to load the document into the system initially, then making some critical processes on that text such as converting all words inside the file to lower case which enables the system to merge terms only variation by case. In addition, removing punctuations and stop words also are included in this stage. Stop words are function words that don't have any meaning to the summary like: english articles ("a","an","the"), prepositions, conjunctions, adverbs...etc. Word stemming also takes a place at this stage, which removes suffixes of each word and returns word's stem. A stemming algorithm that is employed here is Porter stemmer [8]. Involving a stemmer in the model is not only useful for keywords extraction process, but also it minimises the size of entire data which enhances the efficiency and performance of the system [2]. The result of this stage is a set of variable keywords in the form of sentences, which will be used in next stages.

2. Generating Terms' Values stage: We used the extracted keywords from pervious stage in order to rank them according to word frequency-based metric. This measure assigns a number of occurrence for each keywords in the text as a value in order to calculate the importance for each sentences (stems) and extract significant nouns (keys) in next stages. Therefore, if a sentence in the text contains words with high rank of frequency (high weight) , then this sentence is probably significant. The output of this stage is that each keyword in the text is obtained a value which represents its frequency (weight) in the document.

3. Calculation of All Sentences' Scores Stage: This is the most challenging stage of automated text summariser. During this calculation process we made use of the outputs from the terms' values generation stage. The calculation method scores each candidate sentence according to the value of each word that we have specified early. Then this score is multiplied with the compression ratio for getting a compressed value. In other words, in this stage each term in a sentence will be checked, then the sentence is scored according to the weight of its words.

4. Generation of Significant Sentences Stage: This stage only selects the highest scored sentences from the calculation stage. The outputs of this operation is a set of the most significant sentences in the entire document. The selection

العنوان:	Automatic Distractor Generation Using Multiple Similarity Measurements for Multiple Choice Questions
المؤلف الرئيسي:	Saeed, Wael Saeed
مؤلفين آخرين:	Liu, Wei(Super.)
التاريخ الميلادي:	2013
موقع:	سيدني
الصفحات:	1 - 36
رقم MD:	615570
نوع المحتوى:	رسائل جامعية
اللغة:	English
الدرجة العلمية:	رسالة ماجستير
الجامعة:	Western Australia University
الكلية:	School of Computer Science and Software Engineering
الدولة:	أستراليا
قواعد المعلومات:	Dissertations
مواضيع:	الجاسبات الإلكترونية ، هندسة البرمجيات، أسئلة الاختبارات
رابط:	https://search.mandumah.com/Record/615570

Bibliography

[1] akuya Goto ; Tomoko Kojiri ; Toyohide Watanabe ; Tomoharu Iwata ; Takeshi Yamada. Automatic generation system of multiple-choice cloze questions and its evaluation. *Knowledge Management and E-Learning : an International Journal 2*, 3 (2010), 210.

[2] Al-hashemi, R. Text summarisation extraction system (TSES) using extracted keywords. In *International Arab Journal of e-Technology* (June 2010), vol. 1, p. 164.

[3] Al-Yahya, M. Ontoque: A question generation engine for educational assessment based on domain ontologies. In *Advanced Learning Technologies (ICALT), 2011 11th IEEE International Conference on* (july 2011), pp. 393 –395.

[4] Alsubait, T., Parsia, B., and Sattler, U. Mining ontologies for analogy questions: A similarity-based approach. In *OWLED* (2012), P. Klinov and M. Horridge, Eds., vol. 849 of *CEUR Workshop Proceedings*, CEUR-WS.org.

[5] Bin, S., Liying, F., Jianzhuo, Y., Pu, W., and Zhongcheng, Z. Ontology-based measure of semantic similarity between concepts. In *Software Engineering, 2009. WCSE '09. WRI World Congress on* (may 2009), vol. 2, pp. 109 –112.

[6] Bollegala, D., Matsuo, Y., and Ishizuka, M. Measuring semantic similarity between words using web search engines. In *Proceedings of the 16th international conference on World Wide Web* (New York, NY, USA, 2007), WWW '07, ACM, pp. 757–766.

[7] Cilibrasi, R. L., and Vitnyi, P. M. B. P.m.b.: The google similarity distance. *IEEE Transactions on Knowledge and Data Engineering* (2007).

[8] DiMarco, C. Computational linguistics course, the university of waterloo, this is available from https://cs.uwaterloo.ca/ cdimarco/cs784.

[9] Hovy, E. *The Oxford Handbook of Computational Linguistics*. Oxford University Press Inc, 2003, ch. 32.

[10] Jing, L., Zhou, L., Ng, M., and Huang, J. Ontology-based distance measure for text clustering. In *Proc. of the 4th Workshop on Text MiningIn: Proc. of the 4th Workshop on Text Mining, the 6th SIAM International Conference on Data Mining* (2006).

[11] Karamanis, N., Ha, L. A., and Mitkov, R. Generating multiple-choice test items from medical text: A pilot study. In *In Proceedings of INLG 2006* (2006), pp. 104–107.

[12] Moser, J., Gtl, C., and Liu, W. Refined distractor generation with lsa and stylometry for automated multiple choice question generation. In *AI 2012: Advances in Artificial Intelligence*, M. Thielscher and D. Zhang, Eds., vol. 7691 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2012, pp. 95–106.

[13] Mostow, J., and Jang, H. Generating diagnostic multiple choice comprehension cloze questions. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP* (Montréal, Canada, June 2012), Association for Computational Linguistics, pp. 136–146.

[14] Papasalouros, A., Kanaris, K., and Kotis, K. Automatic generation of multiple choice questions from domain ontologies. In *e-Learning'08* (2008), pp. 427–434.

[15] SHIRUDE, S. B., and KOLHE, S. R. A library recommender system using cosine similarity measure and ontology based measure. *Advances in Computational Research 4*, 1 (2012), 91 – 94.

[16] Sicilia, M., Garcia, E., Sanchez, S., and Rodriguez, E. On integrating learning object metadata inside the opencyc knowledge base. In *Advanced Learning Technologies, 2004. Proceedings. IEEE International Conference on* (2004), pp. 900–901.

[17] Wong, W., Liu, W., and Bennamoun, M. Tree-traversing ant algorithm for term clustering based on featureless similarities. *Data Min. Knowl. Discov. 15*, 3 (Dec. 2007), 349–381.

APPENDIX A

Original Master Dissertation Proposal

A.1 Motivation:

Automated multiple choice questions generation becomes a very useful technique to create electronic educational assessments in E-learning systems. It provides a unique self-assessment methodology and immediate individual feedback to the examinees. This sort of self-assessment tool is in high demand by end-users, as well as educators who would use it as a teaching tool for feedback and tests in terms of providing evaluation results.

A multiple-choice question (MCQ) consists of a sentence (stem), which illustrates a question and a set of choices or answers. There is only one of the choices is the correct answer and the rest are wrong ones (distractors). However, the quality of the MCQ depends on the quality of distractors, thus in this research we will concentrate on generating distractors by using multiple similarity measurements for automatic multiple choice question.

A.2 Background and Related Research:

Papasalouros et.al (2008) [14] generated multiple choice questions automatically by utilising three entities; a knowledge base containing facts pertaining to a specific domain, semantic relationships connecting entities in the knowledge base, and natural language generation techniques. Until recently, multiple-choice questions were churned out by applying term extraction, semantic distance calculation and sentence restructuring method on ontologies like WordNet.

According to Papasalouros et.al (2008) [14], generating questions using natural language generation methods on a knowledge base developed using ontology web language (OWL) would be a more effective way to generate multiple choice questions.Utilising OWL can prove to be more efficient, as it is domain-independent and help access multiple domain ontologies. As the entities are arranged based on class, it becomes easier to generate distractors. Moreover, entities are categorised based on their properties, which helps to automatically determine relationships between entities as well as determine data type of an entity. Thus, the process of acquiring correct answers and phrasing distracting options is simplified using this approach. However, a problem arises when the same word can mean different things, under different contexts.

In accordance to Bollegala et.al (2007) [6], semantic similarity can be measured, by using Internet search engines to determine the most plausible and relevant context for a word, using a statistical measure known as page count. This measure basically analyses and ranks the different contexts in which a word has been searched. To determine the semantic similarity between two words, the individual page ranks of the two words as well as the page rank of the two words combined can be used. Thus, the word relevant to the context can be determined and used while framing distractors in multiple-choice questions.

Another approach to formulating multiple-choice question is to gather sentences from questions mentioned in a specified set of learning materials, employ a statistical technique to generate a blank part for the question, and create distractors using grammatical and statistical patterns [1]. While this approach may be useful in generating multiple-choice questions for a specified amount of text, it is not as flexible or powerful as the method recommended by Papasalouros et.al [14].

A.3 Project Aim (Objectives):

The project aims to generate distractors by different method that applies multiple similarity measurements for producing multiple-choice questions from electronic English text.

We will involve Normalised Google Distance (NGD) to generate distractors using Google search engine as well as ontology distance, which is another method to generate distractors. In other words, the project targets to find similar distinct word associated with a query word using multiple similarity measurement for MCQ.

A.4 Methodology:

In order to gain better knowledge and information about MCQ generator, we will research into recent approaches of MCQ generation such as the ones use Natural Language Generation (NLG) and domain ontologies. We will then attempt to integrate them together to form a model that generates distractors using multiple similarity measurements.

In addition, searching about how to find similarities or distances between two words (strings) as well as how to measure similarities between concept will assist us to achieve further understanding of many aspects of the project. The GATE -Natural language processing framework- will be used to process the input text corpus, and we will employ NGD, ontology distance and WordNet to generate distractors.

A.5 Timeline:

Stage 1	Background and further reading.
	Identify project's tasks and tools.
	Write a project proposal.
	Prepare and collect information for literature review.
	Learn Java and WordNet.
	Submit final draft of the literature review and proposal.
Stage 2	Produce codes and tests.
	Write a dissertation.
	Prepare poster and presentation.
	Submit first draft dissertation.
	Submit a poster.
	Seminar presentation.
	Submit final dissertation.

Automatic Distractor Generation Using Multiple Similarity Measurements for Multiple Choice Questions

The University of Western Australia
School of Computer Science and Software Engineering

Supervisor: Asst/Prof. Wei Liu
Author: Wael Saeed Saeed
STD No. 20471262

*This report is submitted as partial fulfilment
of the requirements for the Master of Computer Science of the
School of Computer Science and Software Engineering,
The University of Western Australia,
2013*