GWPS: Geometry Word Problem Solver

A Thesis

Presented to the Faculty of the

College of Computer and Information Sciences

Polytechnic University of the Philippines

In Partial Fulfilment

of the Requirements for the Degree

Bachelor of Science in Computer Science

Apostol, John David O.

Orsolino, Albert Y.

March 2016

AUTHORIZATION

We hereby declare that we are the sole authors of the thesis.

We authorize the Polytechnic University of the Philippines and the College of Computer and Information Sciences to lend this thesis to other institutions or individuals for the purpose of scholarly research.

We further authorize the Polytechnic University of the Philippines and the College of Computer and Information Sciences to reproduce the thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

____

Apostol, John David O.

____

Orsolino Albert Y.

March 2016

The thesis **"GWPS: Geometry Word Problem System"** submitted and presented by [Group members] in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science has been

Recommended for Approval and Acceptance: Date:

______________________________ ________________

Ranil M. Montaril
Faculty-In-charge

Approved by the Thesis Defense Panel:

______________________________ ________________

Michael B. DelaFuente

______________________________ ________________

Ria A.Sagum

Accepted for the Department of Computer Sciences

______________________________ ________________

Michael B. DelaFuente
Department Chairperson

Accepted for the College of Computer and Information Sciences

______________________________ ________________

Gisela May A Albano
Dean of the College

ABSTRACT

This research presented by the researchers entitled, “GWPS: A Geometry Word Problem Solver”, is a system that can solve automatically a given word problem, show a diagram and its solution. The system focuses on Basic Geometry Curriculum. The general problem of the study is recognizing the entities and used it to solve the word problem. To solve this general problem, the researchers used Name-Entity Recognition and POS tagging an algorithm for labelling sequence of words in a text which are the geometric figures, keywords and numbers written in numeric or words.

The research aims to find an answer to the following questions: (1) What is the performance of the developed system in recognizing entities based on the following criteria:(a) Precision (b) Recall (c) Error Rate (2) What is the performance of the developed system in solving the answer based on accuracy? After the gathering of necessary data, the study shows that (1) The overall performance of the system in recognizing entities gives a satisfactory evaluation of the system (2) The over-all performance of the system in solving the answer gives a satisfactory evaluation of the system (3) and although the study shows that the software is effective for the respondents, improvement can still be made in the system. This includes (1) adding more rules in recognizing entities to solve the given word problem. (2) use more keywords, parameters and geometric figures. (3) improve the retrieval of parameters.

ACKNOWLEDGEMENT

Through the help and support from everyone including: parents, professors, classmates, friends and family, this research project made possible. So please allow us to dedicate this acknowledgement of gratitude toward the following advisors and contributors.

First of all, we would like to thank God, for the wisdom, strength and perseverance that He has bestowed upon us during this research project.

To our Thesis Writing Professor, Professor Ranil M. Montaril, for his guidance that helped us throughout the research and writing of this thesis.

To our friends, for their endless friendship and encouragement. Thank you for listening, offering us advice, being there when we needed a friend. Thank you for supporting us during this research project.

Last but not the least, we would like to thank our dear parents for their unconditional support, both financially and emotionally throughout our college life in pursuing our respective degrees.

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

CHAPTER 1

THE PROBLEM AND ITS BACKGROUND

This chapter discusses the introduction, background of the study, theoretical framework, conceptual framework, statement of the problem, scope and limitation, significance of the study, and definition of terms. Goals and objectives of the study are stated in this chapter.

1.1 INTRODUCTION

There are many students who report to mathematics classes each day reading below grade level. Many of them seem to enjoy mathematics because they struggle reading and perceive mathematics to be a subject with little reading and writing. Word problems are an obstacle to those students and they feel frustrated in their attempts to solve them. It is difficult to assess whether or not the students are unsuccessful due to the lack of comprehension while reading the problem, or to the mathematical skills involved in finding the solution.

Computation and comprehension of word problems are both critical in the success of solving word problems. From Fordham University, (Robinowitz and Woolley, 1995) did a study to try to determine if computation and comprehension were tied together or if only one or the other played a role in student success. This study found that there is no interaction between computation and calculation when solving word problems. The researchers found that with an increase in the size of the problem, only the difficulty level of the problem goes up, but the understanding of the numbers involved and the way to manipulate them does not. (Tara Scwanebeck, July 2008)

Thus, the proponents decided to make a system that helps students to understand and comprehend a given word problem by making a system that formulates equation, solutions and solves answer for the given problem automatically.

1.1.1 BACKGROUND OF THE STUDY

There are relatively few studies on the difficulties learners encounter while solving geometry word problems. Due to the fact that geometry word problems have many characteristics of arithmetic word problems, learners are expected to have similar difficulties when solving both types of problems.

Arithmetic word problems play an important role in math education and they are taught in elementary schools at several grades. A lot of researches suggest that many students have trouble in solving arithmetic word problems (e.g., Cummins, Kintsch, Reusser, & Weimer, 1988; Cummins, 1991; Davis-Dorsey, Ross, & Morrison, 1991). Some students perform poorly even in solving very simple word problems. Some researchers found that word problems are difficult for learners because so many skills are involved, such as reading comprehension, equation writing 7 and arithmetic calculation (Cummins, Kintsch, Reusser, & Weimer, 1988; Mayer, 1992; Stern, 1993). Moreover, some studies indicate that students encounter difficulties at the stage of problem comprehension (Mayer, 1987; Wu, 1990). Indeed, many factors accounting for the difficulties of word problem comprehension have been found through empirical experiments. Some studies show that students fail to solve word problems because they could only recognize the problems' surface information rather than fully comprehend the implications of the problems (Cummins, Kintsch, et al, 1988; Cummins, 1991; Davis-Dorsey, Ross, & Morrison, 1991; Verschaffel, De Corte, &Pauwels, 1992). While dealing with a problem of addition and subtraction, students may misunderstand the problem by interpreting some keywords out of context. For example, a student might take the word more in a problem as a signal of addition and the word less as a subtraction signal. Consider the following problem: “There is a rectangle whose base is 10cm longer than its height. The base is 20cm. Please find the area of this triangle.” Some students would see the word longer as an addition signal and jump to the incorrect conclusion that the height of the rectangle is 20cm plus 10cm. Some literatures (Tan, 1998; Tai, 2001) indicate that when students solve geometry word problems, they always directly use a formula without comprehending what answer the problem asks for. Consider the problem: “There is a round skating rink with diameter 30m. Please find the area of this rink.” When solving this problem, most students will directly apply the formula of circumference and ignore that the problem is to find the area rather than the circumference. In addition, research results indicate that many students fail to comprehend the linguistic descriptions of some key concepts (Cummins, Kintsch, Reusser, & Weimer, 1988; Cummins, 1991). Some researchers suggest that other design features of word problems can contribute to comprehension difficulties as well (Dark &Benbow, 1990).

For instance, difficult vocabularies or too much irrelevant information in the problems can overwhelm and confuse students. In addition, when students lack the domain knowledge assumed by the word problems, they are also more likely to have difficulties with problem comprehension.

1.2 CONCEPTUAL FRAMEWORK

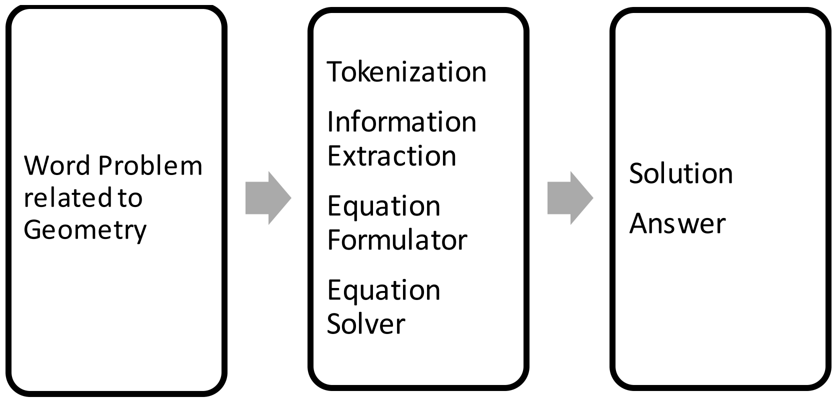


Figure 1 Conceptual Framework of the System

Figure 1 shows Conceptual Framework of the System where the input is the Word Problem related to Geometry. The processes are Tokenization, Information Extraction Equation Formulation and Equation Solver. Lastly the outputs are the solution of the word problem and its answer.

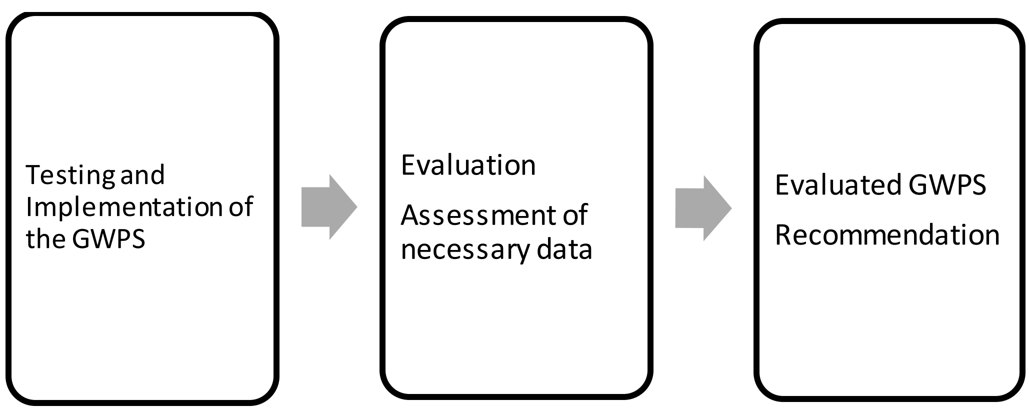


Figure 2 Conceptual Framework of the Study

Figure 2 shows Conceptual Framework of the System where Testing and Implementation of the proposed system is the input. The processes are Evaluation and Assessment of necessary data. The outputs are the Evaluated GWPS and other recommendation.

1.3 STATEMENT OF THE PROBLEM

This study aims to develop a system that automatically solves a Geometry Word Problem using Name Entity Recognition. Specifically, this study sought to answer the following questions:

1. What is the performance of the developed system in recognizing entities based on the following criteria:
   a. Precision
   b. Recall
   c. Error Rate
2. What is the performance of the developed system in solving the answer based on accuracy?

1.4 SCOPE AND LIMITATION

The study mainly focused on the development of Geometry Word Problem Solver System. It aims to make a computer system that automatically solves a Geometry Word Problem. The proposed system also shows diagram or illustration on how the given problem was solved. Basic Geometry Curriculum will be the range of the system.

The sentence or word problem must have a geometric figure or shape not more than one but can be repeated in the sentence or word problem.

The parameters (Example: length, width, side, area, perimeter etc.) must be followed by keywords “is”, “of”, “is equal to”, etc. and followed by numeric value to acquire the required grammar rule.

The spelling of the parameters and keywords must be correct.

1.5 SIGNIFICANCE OF THE STUDY

The result of this study benefits the following groups of people:

1. Students

The study is beneficial to students who have reading comprehension and computational problem. It helps them to solve word problem easily and learn in the given solution.

2. Teachers

The study can be a good basis of teachers in checking the solution of their students in a given word problem. It helps them identify error in their students' reasoning.

3. Future Researchers

The study benefits the future researchers if their research is associated to the topic of Automatic Word Problem Solver. They can continue, enhance, and modify the scope of the study.

1.6 DEFINITION OF TERMS

Geometry - the branch of mathematics that deals with the deduction of the properties, measurement, and relationships of points, lines, angles, and figures in space from their defining conditions by means of certain assumed properties of space.

Word Problem - any mathematics exercise expressed as a hypothetical situation explained in words

Natural Language Processing - is a field of computer science, artificial intelligence, and linguistics concerned with the interactions between computers and human (natural) languages.

Information Extraction-is the task of automatically extracting structured information from unstructured and/or semi-structured machine-readable documents.

Tokenizer – it breaks up text into individual objects. These objects may be Strings, Words, or other Objects

Equation Formulator – formulates an equation from the input geometry word problem for the computation of the answer.

CHAPTER 2

REVIEW OF RELATED LITERATURE/STUDIES

This chapter gives importance to related literatures and studies that deals with the present study. Related literatures and studies in this chapter were carefully read and studied by the proponents for it is ought to provide them ideas and knowledge in doing the study.

2.1 REVIEW OF RELATED LITERATURE/STUDIES

2.1.1 Math word problem solving

Most previous work on automatic word problem solving is symbolic. STUDENT (Bobrow, 1964a, b) handles algebraic problems by first transforming NL sentences into kernel sentences using a small set of transformation patterns. The kernel sentences are then transformed to math expressions by recursive use of pattern matching. CARPS (Charniak, 1968, 1969) uses a similar approach to solve English rate problems. The major difference is the introduction of a tree structure as the internal representation of the information gathered for one object. Liguda& Pfeiffer (2012) propose modeling math word problems with augmented semantic networks. Addition/subtraction problems are studied most in early research (Briars & Larkin, 1984; Fletcher, 1985; Dellarosa, 1986; Bakman, 2007; Ma et al., 2010). Please refer to Mukherjee &Garain (2008) for a review of symbolic approaches before 2008. 1 No empirical evaluation results are reported in most of the above work. Almost all of these approaches parse NL text by simply applying pattern matching rules in an ad-hoc manner. For example, as mentioned in Bobrow (1964b), due to the pattern "($, AND $)", the system would incorrectly divide "Tom has 2 apples, 3 bananas, and 4 pears." into two “sentences”: “Tom has 2 apples, 3 bananas.” and “4 pears.” WolframAlpha1 shows some examples2 of automatically solving elementary math word problems, with technique details unknown to the general public. Other examples on the web site demonstrate a large coverage of short phrase queries on math and other domains. By randomly selecting problems from our dataset and manually testing on their web site, we find that it fails to handle most problems in our problem collection. Statistical learning methods have been proposed recently in two papers: Hosseini et al. (2014) solve single step or multi-step homogenous addition and subtraction problems by learning verb categories from the training data. Kushman et al. (2014) can solve a wide range of word problems, given that the equation systems and solutions are attached to problems in the training set. The method of the latter paper (referred to as KAZB henceforth) is used as one of our baselines.

2.1.2 Tokenization

Tokenization is a process of identification of token/topics within input documents and it helps to reduced search with significant extent. The secondary advantage of tokenization in effective use of storage space, as it reduces the storage spaces required to store tokens identified from input documents. In modern age of data/information, when data/information is expanding manifold on every day from its origin, in form of documents, web pages etc, so importance of effective and efficient tokenization algorithm become critical for an IR system. There are various traditional techniques for tokenization is designed, Porter’s algorithm is one of the most prominent tokenization among all such techniques, but this algorithm suffers from accuracies during the identification and efficiency. The enhanced algorithm is also designed to overcome the inaccuracy in token identification, but problem still persists. In this paper, an approach is proposed for of tokenization, in which is token identification is completely based on the documents vectors.

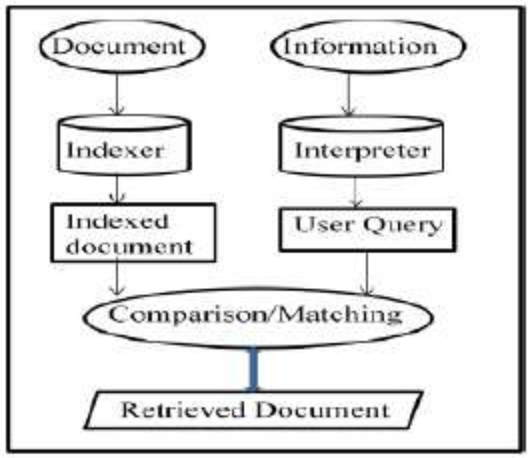


Figure 1Formal IR Model System

Tokenization process is an integral part of IR systems, involves pre-processing of given documents and generates respective tokens. In some, tokenization techniques count of token were used to establish a value “Word Count or Token Count” which is used in indexing/ranking process of document. A typical structure of tokenization process is explained in figure 2. Information retrieval models historically many years back to the beginning of written language as information retrieval is related to knowledge stored in textual form. Ranking algorithm/Indexing algorithm uses the input from tokenization, which is either word count or token count? The affectivity of indexing algorithm is heavily depends upon the quality of token generated by tokenization process. Proposed tokenization model for tokenization is shown in figure 2, primary objective of the tokenization is to identify the words/token/concept and their frequency within input document.

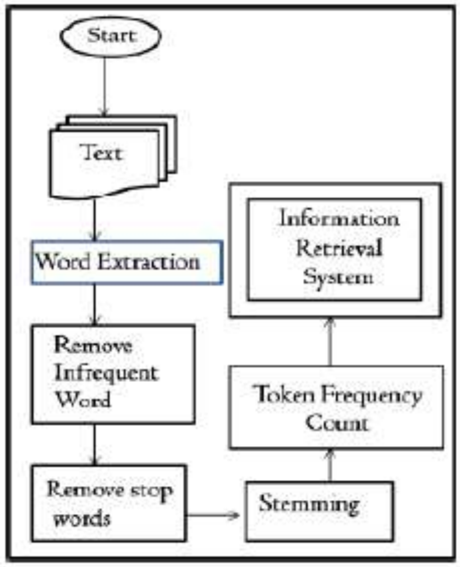


Figure 2Tokenization Process

Clearly, the crucial focus of an IR model is to find the relevant document to issue of finding the relevant document to user's query. Such a decision is usually dependent on an indexing/ranking algorithm which attempts to establish a simple ordering of the documents retrieved. Documents appearing at the top of this ordering are considered to be more likely to be relevant and useful for future patterns. Thus, ranking algorithms are at the core of information retrieval systems. A ranking algorithm operates according to basic premises (regarding document relevance) yield distinct information retrieval models. The IR model adopted determines the predictions of what is relevant and what is not (i.e. the notion of relevance implemented by the system).

Related Work- Traditional document tokenization techniques are being used in various unsupervised learning approaches for solving problems. Traditional approaches often fail to obtain good tokenization solution when users want to group documents according to their need. An approach to make an effective pre-Processing steps to save both space and time requirements by using improved Stemming Algorithm. Stemming algorithms are used to International Journal of Database Management Systems (IJDMS) Vol.6, No.6, December 2014 16 transform the words in texts into their grammatical root form . Several algorithms exist with different techniques. This is most widely used porter's stemming algorithm. The other enhanced working model is also proposed, in which inaccuracies encountered during the stemming process has been removed by proposing a solutions. The tokenization involves multiple activities to be performed during the life cycle. There are still a lot of scope of improvement on the accuracy of token identification capability of algorithm & efficiency of approach.

2.1.3 Information Extraction

Information extraction (IE) is the task of automatically extracting structured information from unstructured documents, mainly natural language texts. Due to the ambiguity of the term"structured information", information extraction covers a broad range of research, from simple data extraction from Web pages using patterns and regular grammars, to the semantic analysis of language for extracting meaning, such as the research areas of word sense disambiguation or sentiment analysis. The basic idea behind information extraction (the concentration of important information from a document into a structured format, mainly in the form of a table) is fairly old, with early approaches appearing in the 1950s, where the applicability of information extraction was proposed by the Zellig Harris for sub-languages, with the first practical systems appearing at the end of the 1970s, such as Roger Schank's systems, which exported "scripts" from newspaper articles. The ease of evaluation of information extraction systems in comparison to other natural language processing technologies such as machine translation or summarisation, where evaluation is still an open research issue, made IE systems quite popular and led to the Message Understanding Conferences (MUC) that redefined this research field. Information extraction can be decomposed into several sub-tasks:

- Linguistic pre-processing, responsible for tasks such as token/sentence identification, part-of-speech tagging, morphological analysis, etc.
- Named-entity recognition, where domain specific entities, such as names of persons, organisations, and locations, monetary/time expressions, etc., are identified.
- Co-reference resolution, where named entity names or other mentions (such as pronouns) that refer to the same entity are grouped/related.
- Template element filling, a task responsible for grouping all properties of a real object into a single template element that represents the real task or event.
- Template relation, a task responsible for identifying relations among template elements.
- Scenario template, a task where related template elements are combined into a template that represents an event.

This thesis has investigated the use of machine learning in three key sub-tasks of information extraction: part of speech tagging, named entity recognition, and relation extraction. Part of speech tagging is an important sub-task of linguistic pre-processing, named entity recognition is an essential subtask of information extraction, and relation extraction is the main activity behind template element filling, template relation and scenario template.

2.1.4 Name-Entity Recognition

The Yarowsky algorithm (Yarowsky, 1995), originally proposed for word sense disambiguation, makes the assumption that it is very unlikely for two occurrences of a word in the same discourse to have different senses. This assumption is exploited by selecting words classified with high confidence according to sense and adding other contexts of the same words in the same discourse to the training data, even if they have low confidence. This allows the algorithm to learn new contexts for the senses leading to higher accuracy. Our algorithm also uses multi-mention features. However, the application of the Yarowsky algorithm to NER involves several domain-specific choices as will become evident below.

Wong and Ng (Wong and Ng, 2007) use the same idea of multiple mentions of a token sequence being to the same named entity for feature engineering. They use a named entity recognition model based on the maximum entropy framework to tag a large unlabelled corpus. Then the majority tags of the named entities are collected in lists. The model is then retrained by using these lists as extra features. This method requires a sufficient amount of manually tagged data initially to work. Their paper shows that, if the initial model has a low F-score, the model with the new features leads to low F-score too. Our method works with a small amount of gold data because, instead of constructing new features, we use independent evidence to enrich the training data with high-accuracy and non-redundant data.

The co-training algorithm proposed by Blum and Mitchell (Blum and Mitchell, 1998) assumes that the features can be split into two class-conditionally independent sets or “views” and that each view is sufficient for accurate classification. The classifier built on one of the views is used to classify a large unlabelled corpus and the data classified with high- 59 confidence are added to the training set on which the classifier on the other view is trained. This process is iterated by interchanging the views. The main reason that co-training works is that, because of the class-conditional independence assumptions, the high-confidence data from one view, in addition to being highly precise, is unbiased when added to the training set for the other view. We could not apply co-training for semi-supervised named entity recognition because of the difficulty of finding informative yet class-conditionally independent feature sets.

Collins et al.(Collins and Singer, 1999) proposed two algorithms for NER by modifying Yarowsky’s method (Yarowsky, 1995) and the framework suggested by (Blum and Mitchell, 1998). However, all their features are at the word sequence level, instead of at the token level. At the token level, the seed rules they proposed do not necessarily work. In addition, parsing sentences into word sequences is not a trivial task, and also not necessary for NER, in our opinion.

Jiao et al. propose semi-supervised conditional random fields (Jiao et al., 2006) that try to maximize the conditional log-likelihood on the training data and simultaneously minimize the conditional entropy of the class labels on the unlabelled data. This approach is reminiscent of the semi-supervised learning algorithms that try to discourage the boundary from being in regions with high density of unlabelled data. The resulting objective function is no longer convex and may result in local optima. Our approach in contrast avoids changing the CRF training procedure, which guarantees global maximum.

2.1.5 Automatic Word Problem Solver

Researchers in MIT's Computer Science and Artificial Intelligence Laboratory, working with colleagues at the University of Washington, have developed a new computer system that can automatically solve the type of word problems common in introductory algebra classes.

In the near term, the work could lead to educational tools that identify errors in students' reasoning or evaluate the difficulty of word problems. But it may also point toward systems that can solve more complicated problems in geometry, physics, and finance — problems whose solutions don't appear in the back of the teacher's edition of a textbook.

An MIT graduate student in electrical engineering and computer science and lead author on the new paper, the new work is in the field of "semantic parsing," or translating natural language into a formal language such as arithmetic or formal logic (Nate Kushman, 2014). Most previous work on semantic parsing — including his own — has focused on individual sentences, "In these algebra problems, you have to build these things up from many different sentences,"."The fact that you're looking across multiple sentences to generate this semantic representation is really something new."(Nate Kushman, 2014).

Kushman is joined on the paper by (Regina Barzilay), a professor of computer science and engineering and one of his two thesis advisors, and by the University of Washington's (YoavArtzi) and (Luke Zettlemoyer). The researchers will present their work at the annual meeting of the Association for Computational Linguistics in June.

The researchers' system exploits two existing computational tools. One is the computer algebra system Macsyma, whose initial development at MIT in the 1960s was a milestone in artificial-intelligence research. For Kushman and his colleagues' purposes, Macsyma provided a way to distill algebraic equations with the same general structure into a common template.

The other tool is the type of sentence parser used in most natural-language-processing research. A parser represents the parts of speech in a given sentence and their syntactic relationships as a tree — a type of graph that, like a family-tree diagram, fans out at successive layers of depth.

For the researchers' system, understanding a word problem is a matter of correctly mapping elements in the parsing diagram of its constituent sentences onto one of Macsyma's equation templates. To teach the system how to perform that mapping, and to produce the equation templates, the researchers used machine learning.

Kushman found a website on which algebra students posted word problems they were having difficulty with, and where their peers could then offer solutions. From an initial group of roughly 2,000 problems, he culled 500 that represented the full range of problem types found in the larger set.

In a series of experiments, the researchers would randomly select 400 of the 500 problems, use those to train their system, and then test it on the remaining 100.

For the training, however, they used two different approaches — or, in the parlance of machine learning, two different types of supervision. In the first approach, they fed the system both word problems and their translations into algebraic equations — 400 examples of each. But in the second, they fed the system only a few examples of the five most common types of word problems and their algebraic translations. The rest of the examples included only the word problems and their numerical solutions.

In the first case, the system, after training, was able to solve roughly 70 percent of its test problems; in the second, that figure dropped to 46 percent. But according to Kushman, that's still good enough to offer hope that the researchers' approach could generalize to more complex problems.

In determining how to map natural language onto equation templates, the system examined hundreds of thousands of "features" of the training examples. Some of those features related specific words to problem types: For instance, the appearance of the phrase "react with" was a good indication that the problem dealt with chemistry. Other features looked at the location of specific words in parsing diagrams: The appearance of the word "costs" as the main verb indicated a great deal about which sentence elements should be slotted into which equation templates.

Other features simply analyzed the syntactical relationships between words, regardless of the words themselves, while still others examined correlations between words' locations in different sentences. Finally, Kushman says, he included a few "sanity check" features, such as whether or not the solution yielded by a particular equation template was a positive integer, as is almost always the case with algebraic word problems.

“The idea of this kind of supervision they have will be useful for lots of things,” says Kevin Knight, a professor of computer science of the University of Southern California. “The approach of building a generative story of how people get from text to answers is a great idea.”

The system’s ability to perform fairly well even when trained chiefly on raw numerical answers is “super-encouraging,” Knight adds. “It needs a little help, but it can benefit from a bunch of extra data that you haven’t labeled in detail.”

Most previous work on automatic word problem solving is symbolic. STUDENT (Bobrow, 1964a, b) handles algebraic problems by first transforming NL sentences into kernel sentences using a small set of transformation patterns. The kernel sentences are then transformed to math expressions by recursive use of pattern matching. CARPS (Charniak, 1968, 1969) uses a similar approach to solve English rate problems. The major difference is the introduction of a tree structure as the internal representation of the information gathered for one object. Liguda& Pfeiffer (2012) propose modeling math word problems with augmented semantic networks. Addition/subtraction problems are studied most in early research (Briars & Larkin, 1984; Fletcher, 1985; Dellarosa, 1986; Bakman, 2007; Ma et al., 2010). Please refer to Mukherjee &Garain (2008) for a review of symbolic approaches before 2008. 1 http://www.wolframalpha.com No empirical evaluation results are reported in most of the above work. Almost all of these approaches parse NL text by simply applying pattern matching rules in an ad-hoc manner. For example, as mentioned in Bobrow (1964b), due to the pattern “($, AND $)”, the system would incorrectly divide “Tom has 2 apples, 3 bananas, and 4 pears.” into two “sentences”: “Tom has 2 apples, 3 bananas.” and “4 pears.” WolframAlpha1 shows some examples2 of automatically solving elementary math word problems, with technique details unknown to the general public. Other examples on the web site demonstrate a large coverage of short phrase queries on math and other domains. By randomly selecting problems from our dataset and manually testing on their web site, we find that it fails to handle most problems in our problem collection. Statistical learning methods have been proposed recently in two papers: Hosseini et al. (2014) solve single step or multi-step homogenous addition and subtraction problems by learning verb categories from the training data. Kushman et al. (2014) can solve a wide range of word problems, given that the equation systems and solutions are attached to problems in the training set. The method of the latter paper (referred to as KAZB henceforth) is used as one of our baselines.

CHAPTER 3

RESEARCH METHODOLOGY

This chapter of the study discusses the research methods used, system architecture, research paradigm, population frame and description of respondents, instrumentation, sampling technique, data gathering procedure, and statistical treatment. Collection of necessary and essential data will be conferred in this chapter.

3.1 Research Method

The proponents used experimental method of research design. Experimental research design's purpose is to provide answers to research question and to control variance.

The proponents used the experimental research method to measure the accuracy of the developed system in solving a word problem. With this, constant changes were done to achieve the accuracy and consistency. For every alteration, the effects were observed and scrutinized if the said changes benefited or detriment the system. Experimental method enabled the proponents to attain the desired outcome of the system. With the help of brainstorming within the group and the analysis of the system's current performance, the system was subject to continuous alterations until the system was successfully and desirably working.

3.2 System Architecture

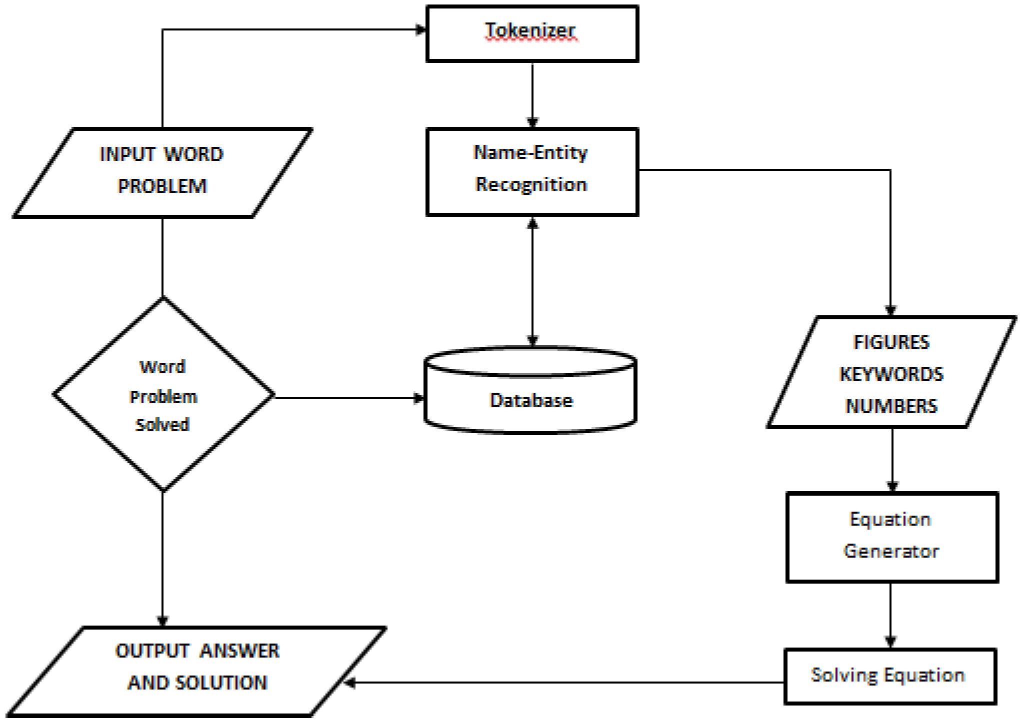


Figure 3System Flow of the System

User input text (Geometry Word Problem) in textual format. The system assumed that the input was grammatically, syntactically and semantically correct. Then system updated the database. During the pre-processing, system first tokenized the word problem where the Word Problem where subdivided into words then it performs Name-Entity for identifying and retrieval of all the Geometric Figures, numbers and keywords.

The system will first check if the word problem has already been solved. If the word problem has already been solved the system automatically display the solution and the answer. And if not, the system used the retrieved nouns and check if it was related to parameter, areas or degree. The retrieved and checked noun will be the basis on what formula should be used. After that, the retrieved numbers will be substitute in their corresponding places in the equation by using the keywords. Then the system computes for the unknown and display the solution and its answer.

3.4 Description of the Respondents

The respondents for this research were the proponents since the proponents are knowledgeable in geometric word problem. The proponents gathered word problem related to Geometry from the internet and used it to train the system. The researcher then evaluated the accuracy of the proposed system.

3.5 Sampling Technique

The proponents used purposive sampling technique since this type of sampling is a non-probability sampling in which decisions concerning the individuals to be included in the sample are taken by the researcher, based upon a variety of criteria which may include specialist knowledge of the research issue, or capacity and willingness to participate in the research. (Paul Oliver, 2013)

3.6 Instrumentation

An experiment paper was used as data-gathering instrument since experimental method was used in the study. The collected geometric word problem samples were tested individually. After that, all required data were tallied to evaluate the developed system in recognizing entities and in generating correct answer.

The proponents evaluated their system in recognizing entities by the total number of entities needed to be recognized (E), total number of correct recognized entities (CE), total number of incorrect recognized entities (IE) and total number of unrecognized entities (UE). From these the proponents computed the error rate, precision and recall of the developed system in recognizing entities. Just like in evaluation of the system accuracy in generating correct answer. The proponents evaluated their system by the total number of word problem (WP) and the total number of the correct answer (CA). From these the proponents computed for the accuracy of the system in generating correct answer.

The proponents also used the internet as a source of related facts and important information that can help developing the proposed system. To create the proposed system the proponents programming language used was Visual Basic.

3.7 Data Gathering

The proponents gathered important data from the present and previous literature and studies related to the proponent's current study. . These literatures and studies came from journals of natural language processing conferences and math word problem studies, and also from portable document files existing in the internet. The proponents formulated ideas on how to create the system with the help of the said information.

The proponents perform some experiment to test the accuracy of the system in solving the answer and in recognizing the entities by inputting 35 sample of word problem related in Geometry. One way to assess the accuracy of a system is to compare its results to the known actual value. The proponents compare the computed answer to the accepted answer then use the formula discuss in Statistical Treatment to compute for the accuracy.

Data gathering was done as long as the system required training to finally execute accurately.

1.7 3.8 Statistical Treatment

Statistics is the science of learning from data, and of measuring, controlling, and communicating uncertainty. Through statistics, the researchers can determine the performance level of the system based from the results of the test conducted. Below are the common evaluation measures that the proponents will use in measuring the precision recall and accuracy.

Formula for getting the accuracy, precision, recall and error rate of GWPS in recognizing entities

$$\text{Accuracy} = \frac{Precision+Recall}{3}$$

Where:

$$\text{Precision} = \frac{CE}{CE+IE} * 100$$

Where:

CE = total no. of correct recognized entities

IE = total no. of incorrect recognized entities

100 = the 100% ratio

$$\text{Recall} = \frac{CE}{CE+UE} *100$$

Where:

CE = total no. of correct recognized entities

UE = total no. of unrecognized entities

100 = the 100% ratio

$$\text{Error Rate} = \frac{E-IE}{E} * 100$$

Where:

E = total no. of entities to be recognized

IE = total no. of incorrect recognized entities

100 = the 100 % ratio

$$\text{Overall Error Rate} = \frac{E1+E2+\cdots E3}{3}$$

Where:

E1 = Error Rate of Word Problem 1

E2 = Error Rate of Word Problem 2

…

E3 = Error Rate of Word Problem 7

$$\text{Overall Precision} = \frac{P1+P2+\cdots P3}{3}$$

Where:

P1 = Precision of Word Problem 1

P2 = Precision of Word Problem 2

…

P3 = Precision of Word Problem 7

$$\text{Overall Recall} = \frac{R1+R2+\cdots R3}{3}$$

Where:

R1 = Recall of Word Problem 1

R2 = Recall of Word Problem 7

…

R3 = Recall of Word Problem7

$$\text{Overall Accuracy} = \frac{A1+A2+A3}{3}$$

Where:

A1 = Accuracy of Word Problem 1

A2 = Accuracy of Word Problem 2

…

A3 = Accuracy of Word Problem 7

Formula for getting the accuracy of GWPS in solving word problem

$$\text{Accuracy} = \frac{CA}{WP} * 100$$

Where:

CA = Total no. of correct answer

WP = Total no. of word problem inputted

CHAPTER 4

ANALYSIS, PRESENTATION AND INTERPRETATION OF THE DATA

This chapter presents the data collected in this study by combining the tabular and textual forms, which are carefully analyzed and interpreted.

For the researchers to answer the problem statement, the researchers collect 5 samples of word problem in each category and tally the required data. Researchers use the formula shown in Statistical Treatment to compute for the error rate, precision, recall and accuracy of the system in recognizing entities and also to compute for the accuracy of the system in solving the given word problem. Tables show below the tallied data collected in the implementation and their interpretation and results.

Table 1 Gathered Data for Evaluation of GWPS in Recognizing Entities from the Word Problem Related to Circle

Word Problem	Entities	Recognized Entities	Unrecognized Entities	Corrected Recognized Entities	Incorrect Recognized Entities
1	4	3	1	3	0
2	7	6	1	6	0
3	4	4	0	4	0
4	4	3	1	4	0
5	7	6	1	6	0
Total	26	22	4	23	0

Table 1 shows that the total recognized entities in word problem related to circle is **26**, recognized entities is **22**, Unrecognized Entities is **4**, Correct Recognized Entities is **23** and Incorrect Recognized Entities is **0**.

Table 2 Evaluation Results of GWPS in Recognizing Entities from the Word Problem Related to Circle.

Error Rate	Precision	Recall	Accuracy
7.14%	100%	92.86%	96.43%

Table 2 shows that the overall error rate in recognizing entities based on the inputted word problem related to circle is **7.14%,** precise is **100%,** recall is **92.86%,** and accuracy **96.43%.**

Table 3 Gathered Data for Evaluation of GWPS in Recognizing Entities from the Word Problem Related to Rectangle

Word Problem	Entities	Recognized Entities	Unrecognized Entities	Corrected Recognized Entities	Incorrect Recognized Entities
1	7	7	0	7	0
2	9	9	0	9	0
3	6	5	1	5	0
4	8	8	0	8	0
5	6	6	0	6	0
Total	28	26	2	26	0

Table 3 shows that the total recognized entities in word problem related to rectangle is **28**, recognized entities is **26**, Unrecognized Entities is **2**, Correct Recognized Entities is **26** and Incorrect Recognized Entities is **0**.

Table 4 Evaluation Results of GWPS in Recognizing Entities from the Word Problem Related to Rectangle.

Error Rate	Precision	Recall	Accuracy
2.78%	100%	97.22%	98.61%

Table 4 shows that the overall error rate in recognizing entities based on the inputted word problem related to rectangle is **2.78%,** precise is **100%,** recall is **97.22%,** and accuracy **98.61%.**

Table 5 Gathered Data for Evaluation of GWPS in Recognizing Entities from the Word Problem Related to Square.

Word Problem	Entities	Recognized Entities	Unrecognized Entities	Corrected Recognized Entities	Incorrect Recognized Entities
1	5	5	0	5	0
2	8	8	0	8	0
3	4	4	0	4	0
4	5	4	1	4	0
5	5	4	1	4	0
Total	27	25	2	25	0

Table 5 shows that the total recognized entities in word problem related to square is **27**, recognized entities is **25**, Unrecognized Entities is **2**, Correct Recognized Entities is **25** and Incorrect Recognized Entities is **0**.

Table 6 Evaluation Results of GWPS in Recognizing Entities from the Word Problem Related to Square.

Error Rate	Precision	Recall	Accuracy
7.41%	100%	92.59%	96.3%

Table 6 shows that the overall error rate in recognizing entities based on the inputted word problem related to square is **7.41%,** precise is **100%,** recall is **92.59%,** and accuracy **96.3%.**

Table 7 Gathered Data for Evaluation of GWPS in Recognizing Entities from the Word Problem Related to Cube.

Word Problem	Entities	Recognized Entities	Unrecognized Entities	Corrected Recognized Entities	Incorrect Recognized Entities
1	4	3	1	3	0
2	7	6	1	6	0
3	4	4	0	4	0
4	4	3	1	4	0
5	7	6	1	6	0
Total	26	22	4	23	0

Table 4.7 shows that the total recognized entities in word problem related to cube is **26**, recognized entities is **22**, Unrecognized Entities is **4**, Correct Recognized Entities is **23** and Incorrect Recognized Entities is **0**.

Table 8 Evaluation Results of GWPS in Recognizing Entities from the Word Problem Related to Cube.

Error Rate	Precision	Recall	Accuracy
15.38%	100%	85.19%	92.59%

Table 8 shows that the overall error rate in recognizing entities based on the inputted word problem related to cube is **15.38%** precise is **100%,** recall is **85.19%,** and accuracy **92.59%.**

Table 9 Gathered Data for Evaluation of GWPS in Recognizing Entities from the Word Problem Related to Pyramid.

Word Problem	Entities	Recognized Entities	Unrecognized Entities	Corrected Recognized Entities	Incorrect Recognized Entities
1	8	8	0	8	0
2	16	16	0	16	0
3	8	7	1	7	0
4	8	8	0	8	0
5	8	8	0	8	0
Total	48	47	1	47	0

Table 9 shows that the total recognized entities in word problem related to pyramid is **48**, recognized entities is **47**, Unrecognized Entities is **1**, Correct Recognized Entities is **47** and Incorrect Recognized Entities is **0**.

Table 10 Evaluation Results of GWPS in Recognizing Entities from the Word Problem Related to Pyramid.

Error Rate	Precision	Recall	Accuracy
2.08%	100%	97.92%	98.96%

Table 10 shows that the overall error rate in recognizing entities based on the inputted word problem related to pyramid is **2.08%,** precise is **100%,** recall is **97.92%,** and accuracy **98.96%.**

Table 11 Gathered Data for Evaluation of GWPS in Recognizing Entities from the Word Problem Related to Trapezoid.

Word Problem	Entities	Recognized Entities	Unrecognized Entities	Corrected Recognized Entities	Incorrect Recognized Entities
1	8	6	2	6	0
2	12	10	2	10	0
3	8	8	0	8	0
4	8	8	0	8	0
5	8	8	0	8	0
Total	44	40	4	40	0

Table 11 shows that the total recognized entities in word problem related to trapezoid is **44**, recognized entities is **40**, Unrecognized Entities is **4**, Correct Recognized Entities is **40** and Incorrect Recognized Entities is **0**.

Table 12 Evaluation Results of GWPS in Recognizing Entities from the Word Problem Related to Trapezoid.

Error Rate	Precision	Recall	Accuracy
9.09%	100%	90.9%	95.45%

Table 12 shows that the overall error rate in recognizing entities based on the inputted word problem related to trapezoid is **9.09%,** precise is **100%,** recall is **90.9%,** and accuracy **95.45%.**

Table 13 Gathered Data for Evaluation of GWPS in Recognizing Entities from the Word Problem Related to Parallelogram.

Word Problem	Entities	Recognized Entities	Unrecognized Entities	Corrected Recognized Entities	Incorrect Recognized Entities
1	6	6	0	6	0
2	6	6	0	6	0
3	6	6	0	6	0
4	6	6	0	6	0
5	6	5	1	5	0
Total	30	29	1	29	0

Table 13 shows that the total recognized entities in word problem related to parallelogram is **30**, recognized entities is **29**, Unrecognized Entities is **1**, Correct Recognized Entities is **29** and Incorrect Recognized Entities is **0**.

Table 14 Evaluation Results of GWPS in Recognizing Entities from the Word Problem Related to Parallelogram.

Error Rate	Precision	Recall	Accuracy
3.33%	100%	96.67%	98.33

Table 14 shows that the overall error rate in recognizing entities based on the inputted word problem related to parallelogram is **3.33%,** precise is **100%,** recall is **96.67%,** and accuracy **98.33%.**

Table 15 Overall Evaluation Results of GWPS in Recognizing Entities.

Experiment	**Error Rate (%)**	**Precise (%)**	**Recall (%)**	**Accuracy (%)**
1. Circle	7.14%	100%	92.86%	96.43%
2. Rectangle	2.78%	100%	97.22%	98.61%
3. Square	7.41%	100%	92.59%	96.3%
4. Cube	15.38%	100%	85.19%	92.59%
5. Pyramid	2.08%	100%	97.92%	98.96%
6. Trapezoid	9.09%	100%	90.90%	95.45%
7. Parallelogram	3.33%	100%	96.67%	98.33
Overall	**6.75%**	**100%**	**93.34%**	**96.67%**

Table 15 shows that the overall error rate in recognizing entities based on the inputted word problems is **6.75%**, precise is **100%** recall is **93.34%** and accuracy **96.67%.**

From the gathered data, the word problem related to cube has the least accuracy which is 92.59%and highest error rate which is 15.38% among the 7 categories ofthe word problems. The researchers concluded that this is because of the unrecognized word “Surface area”, since this word is always included in the inputted word problem related to cube. The data trained by the researcher contains no spaces and is stored and written as “Surfacearea”. The possible solution for this problem is to add “Surface area” in stored data.

Since GWPS did not recognize entities which is not needed to be recognized the precision of the proposed system reach 100% in every category.

Table 16 Tallied Data for Evaluation of GWPS in Generating Correct Answer.

Circle	Correct	Rectangle	Correct	Square	Correct
1	✓	1	✓	1	✓
2	✓	2	✓	2	✓
3		3		3	✓
4	✓	4	✓	4	✓
5	✓	5	✓	5	
Cube	**Correct**	**Pyramid**	**Correct**	**Trapezoid**	**Correct**
1		1	✓	1	✓
2		2	✓	2	✓
3	✓	3		3	✓
4		4	✓	4	✓
5		5	✓	5	✓
Parallelogram	**Correct**				
1	✓				
2	✓				
3	✓				
4	✓				
5	✓				

From the gathered data shows in Table 16 the researchers come up with a result that the overall accuracy of GWPS in solving the answer of the word problem is 77.14% based on the thirty-five (35) inputted word problems.

From the gathered data, the experiment in Cube has generated only one correct answer. The researcher concluded that this is because of the unrecognized parameter “Surface area” as said in the evaluation in recognizing entities by which it is the parameter that doesn’t have a value and needed to be solved. Because of that, the system doesn’t know what to do and can’t solve and show the answer for the word problem. Thus, the researchers proposed to the future researchers to add more parameters, keywords and Geometric figures for the system to minimize errors and can solve more word problem.

Regarding the generated solution of GWPS, the researchers consulted a mathematician teacher to evaluate the given solution. According to her, the display of the step by step solution is understandable and correct. Thus, the researcher concluded that the generated solution of GWPS is effective to help students understand how the word problem is being solved.

CHAPTER 5

SUMMARY, CONCLUSION AND RECOMMENDATION

This chapter contains the summary, conclusion and recommendation of the study. The summary will give the researchers a chance to backtrack what was stated in the five previous chapters. The conclusion gives the list of ideas that the proponents learned during the implementation of the proposed software. And lastly, the recommendation gives some suggestions for future researchers who decide to examine the study.

5.1 Summary

The research entitled GPWS: Geometry Word Problem Solver is a system that can solve automatically a given word problem, shows a step by step solution and a visual diagram of the problem. The system focuses on word problem that is related in geometry.

This study acquired the response of the respondents based on the error rate, precision, recall and accuracy of the system in recognizing entities and the degree of correctness in terms of the Accuracy of the generated geometry word problems.

This study was made to develop software that can solve and give solutions and answers focusing in the field of Geometry. The researcher used different methods to support their documentation. The methods used were as follows: Information Extraction: Name Entity Recognition; Purposive Sampling used for selecting the respondents; and Non-Probability Sampling for determining the population for gathering data. After collecting information and gathering data the researchers came up with the conclusion and recommendation based on what was derived from the respondents.

Based from the gathered data, the overall accuracy of the system in solving the word problem is 86.65%. And for recognizing entities, the overall accuracy is 97.02%.

5.1 Conclusion

The system Word Problem Solver using Name Entity Recognition was evaluated by researchers. Based on the gathered data, the system can solve a geometry word problem with a high percentage of accuracy with solutions and answer. Using Name-Entity Recognition, the system can determine a geometric figure, keywords and parameters accurately to solve a word problem.

The researchers concluded that it is a helpful tool for the individuals who are taking-up geometry classes and for the individuals who wants to learn how to solve a geometric figure. Based on the gathered data the system has a less error rate. Researcher also concluded that the main factor that affects the error rate is the lack of parameters trained.

The researchers concluded that the system need some improvements like better Illustration or diagrams for each geometric figures to be able to determine it precisely. Regarding User-Friendliness, the researchers concluded that the objects inside the GUI were improperly organized, thus making them less effective.

5.3 Recommendation

The researchers would like to recommend the following for further improvements on the system and for further research. For the future study of other researchers, the researchers would like to enhance the system to capture the children's attention through the following recommendations:

1.Though the developed system gave a very satisfactory evaluation in solving geometry word problem future researchers and developers who will conduct similar or relate study, are suggested to enhance ―GWPS: Geometry Word

Problem Solver System by adding more rules in recognizing entities to solve the given word problem.

2. Future researchers and developers are suggested to use more keywords, parameters and geometric figures.

3.Future researchers and developers are also suggested to improve the retrieval of parameters by using more accurate POS tagger and Name-Entity Recognition and add more rules so that that parameter that has double or more words spaces can recognized as entities.

Reference

Wing-Kwong Wong, Sheng-Cheng Hsu, Shih-Hung Wu, Cheng-Wei Lee, Wen-LianHsuc, LIM-G: Learner-initiating Instruction Model based on Cognitive Knowledge for Geometry Word Problem Comprehension

Larry Hardesty, (2014, May 2). Computer system automatically solves word problems Retrieved from http://news.mit.edu/2014/computer-system-automatically-solves-word-problems-0502.

ShumingShi ,Yuehui Wang , Chin-Yew Lin, Xiaojiang Liu and Yong Rui,. Automatically Solving Number Word Problems by Semantic Parsing and Reasoning 1 Microsoft Research retrieved from http://www.emnlp2015.org/proceedings/EMNLP/pdf/EMNLP135.pdf

ChadliAbdelhafid, Bendella Fatima, BelmabroukKarima. COMPUTER-ASSISTED PROBLEM-BASED LEARNING FOR WORD-BASED MATHEMATICAL PROBLEM SOLVING retrieved from http://ceur-ws.org/Vol-991/paper4.pdf

Duco A. Schreuder (2014). Vision and Visual Perception retrieved from https://books.google.com.ph/

Stephen K., (1998). Word Problems: Research and Curriculum Reform By. Reed retrieved from https://books.google.com.ph

Tara Schwanebeck, (2008). A Study of Summarization of Word Problems retrieved from http://scimath.unl.edu/MIM/files/research/SchwanebeckT.pdf

Georgios P. Petasis. Machine Learning in Natural Language Processing retrieved form http://cgi.di.uoa.gr/~phdsbook/files/Petasis.pdf

Wenhui Liao and SriharshaVeeramachaneni.A Simple Semi-supervised Algorithm ForNamed Entity Recognition retrieved from http://www.aclweb.org/anthology/W09-2208

APPENDIX A

Implementation Report

Introduction

The research entitled GPWS: Geometry Word Problem Solver is a system that can solve automatically a given word problem, shows step by step solution and a visual diagram of the problem. The system focuses on word problem especially in Geometry.

Problem Statement

This study aims to develop a system that automatically solves a Geometry Word Problem using Name Entity Recognition. Specifically, it wants to answer the following questions:

1 What is the performance of the developed system in recognizing entities based on the following criteria:
   a. Precision
   b. Recall
   c. Error Rate

2 What is the performance of the developed system in solving the answer based on accuracy?

Respondents/Subjects

The respondents for this research were the proponents since the proponents are knowledgeable in geometric word problem. The proponents inputted and gathered also geometry word problem from the internet and used it to train the system. The researcher then evaluated the accuracy of the proposed system.

Time Frame

The implementation was done in 1 day. The researchers perform experiment around 20-30 minutes.

Implementation Procedures

The proponents gathered all necessary data by following the steps indicated below:

a. The proponents gathered samples of geometry word problem from internet.

b. The gathered samples then used in testing the system.

c. After every word problem solved by the system, proponents record the generated answer for evaluation of the GWPS in solving word problems and tallied all data that is needed in evaluating the created system for the evaluation of the GWPS in recognizing entities.

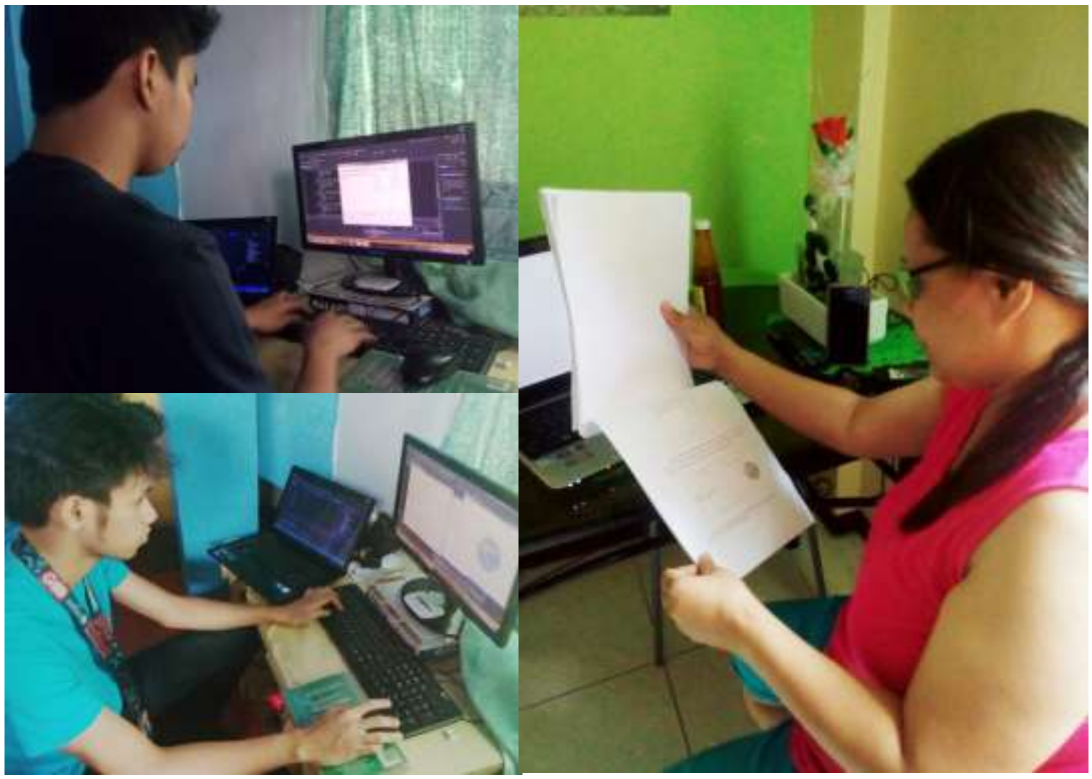
d. The collected data was computed and evaluated to interpret the result and sum up the findings of the study.

e. The researcher also consults a mathematician teacher to evaluate the generated solution of GWPS.

Issues and Concerns

- The researchers having hard time to gather sample of geometric word problem.
- The researchers having hard time to find Mathematics professors for to evaluate solution generated by the system.

Pictures of Implementation



Sample gathered geometry word problem implemented in the created system. From the POS tagger and NER, the proponents collected the necessary data for the evaluation of the created system in recognizing entities

Sample Output of the GWPS. After the compute button is click, the system shows solution, visual illustration of the problem and the answer.

APPENDIX B

Experiment Paper for evaluation of the error rate, precision and recall in recognizing entities of the problem.

	Total No. of Entities	Total No. of Recognized Entities	Total no. of Unrecognized Entities	Total no. of Correct Recognized Entities	Total no. of Incorrect Recognized Entities
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
.					
.					
.					
.					

Experiment Paper for the evaluation of the accuracy in solving the answer of the problem.

	Correct Answer	Generated Answer
1		
2		
3		
4		
5		
6		
7		
8		
9		
10		
.		
.		
.		
.		

APOSTOL, JOHN DAVID O.
#5293 Phase 1 Block 10 Ilugin, Pinagbuhatan, Pasig City
Contact no. 09062771429
Email add. debid07@yahoo.com.ph

Educational Background

College: Polytechnic University of the Philippines

(BS in Computer Science)

Anonas, Santa Mesa, Manila, 1016 Metro Manila, Philippines

S.Y 2012-Present

Secondary: PINAGBUHATAN HIGH SCHOOL

Pinagbuhatan Pasig City

S.Y 2008-2012

Elementary: ILUGIN ELEMENTARY SCHOOL

Ilugin, Pinagbuhatan Pasig City

S.Y 2002-2008

Skills

- Basic HTML Web Design, Java Programming
- Basic C ,Visual Basic and Python Programming
- Basic MATLAB Programming
- Basic Knowledge of Adobe Flash ,and Photoshop
- Proficient in Microsoft Office

ORSOLINO, ALBERT Y.
#34 Lupha Phase 1 Letre Road Tonsuya Malabon City
Contact no. 09297685661
Email address : albert.orsolino31@gmail.com

Educational Background

College: Polytechnic University of the Philippines

(BS in Computer Science)

Anonas, Santa Mesa, Manila, 1016 Metro Manila, Philippines

S.Y 2012-Present

Secondary: Macalaya National High School

Macalaya, Castilla, Sorsogon

S.Y 2007-2011

Elementary: Macalaya Elementary School

Macalaya, Castilla, Sorsogon

S.Y 2001-2007

Skills

- Basic HTML Web Design, Java Programming
- Basic C++,Visual Basic and Python Programming
- Basic MATLAB Programming
- Basic Knowledge of Adobe Flash ,and Photoshop
- Proficient in Microsoft Office

CERTIFICATE OF COMPLETION

This is to certify that the thesis work entitled **GWPS: Geometry Word Problem Solver**

by John David O. Apostol and Albert Y. Orsolino

was proofread and edited by the undersigned.

This certification is being issued for whatever legal purpose it may serve.

Signed by:

Ms. Jill iris O. Apostol
(ABE graduate; 2009 LET passer)

Date:

April 11, 2016