# PhySol: Machine Solver for Textually Represented Physics Problem Implementing RNN-based Information Extraction

A Thesis Proposal  
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

Besabe, Lester Y.  
Camero, Jan Andrew S.

September 2017

# TABLE OF CONTENTS

[TITLE PAGE i](#_Toc492854664)

[TABLE OF CONTENTS ii](#_Toc492854665)

[LIST OF FIGURES iii](#_Toc492854666)

[LIST OF TABLES iv](#_Toc492854667)

[LIST OF NOTATION v](#_Toc492854668)

[CHAPTER 1 The Problem and Its Background 1](#_Toc492854669)

[1.1 BACKGROUND OF THE STUDY 1](#_Toc492854670)

[1.2 STATEMENT OF THE PROBLEM 3](#_Toc492854671)

[1.3 THEORETICAL/CONCEPTUAL FRAMEWORK 4](#_Toc492854672)

[1.3.1 THEORETICAL FRAMEWORK 4](#_Toc492854673)

[1.3.2 CONCEPTUAL FRAMEWORK 8](#_Toc492854674)

[1.4 SIGNIFICANCE OF THE STUDY 9](#_Toc492854675)

[1.5 SCOPE AND LIMITATION OF THE STUDY 10](#_Toc492854676)

[1.6 OPERATIONAL TERMS 10](#_Toc492854677)

[CHAPTER 2 Review of Related Literature 12](#_Toc492854678)

[2.1. RELATED LITERATURE and STUDIES 12](#_Toc492854681)

[2.2. SYNTHESIS OF THE STUDY 32](#_Toc492854682)

[CHAPTER 3 Research Methodology 33](#_Toc492854683)

[3.1. RESEARCH DESIGN 33](#_Toc492854687)

[3.2. SOURCES OF DATA 33](#_Toc492854688)

[3.3. INSTRUMENTATION 34](#_Toc492854689)

[3.3.1 SOFTWARE/HARDWARE TOOLS 34](#_Toc492854690)

[3.3.1.1 SYSTEM ARCHITECTURE 34](#_Toc492854691)

[3.3.1.2 DEVELOPMENT DETAILS 36](#_Toc492854692)

[3.3.2 RESEARCH INSTRUMENT 37](#_Toc492854693)

[3.4. DATA GENERATION 37](#_Toc492854694)

[3.5. DATA ANALYSIS 38](#_Toc492854695)

[REFERENCES 41](#_Toc492854696)

[APPENDICES 43](#_Toc492854697)

[APPENDIX A: SAMPLE RESEARCH INSTRUMENT 43](#_Toc492854698)

# LIST OF FIGURES

[Figure 1. 1: Theoretical Flow of the System 4](#_Toc492845777)

[Figure 1. 2: Simple Architecture for an Information Extraction System 5](#_Toc492845778)

[Figure 1. 3: General Model for Feedforward Neural Network 6](#_Toc492845779)

[Figure 1. 4: General Model for Recurrent Neural Network 7](#_Toc492845780)

[Figure 1. 5: Model for a Simple Recurrent Network 7](#_Toc492845781)

[Figure 1. 6: Conceptual Framework of the Study 8](#_Toc492845782)

[Figure 2. 1: System Architecture of Word Problem Solver 18](#_Toc492845879)

[Figure 2. 2: Example of the Problem-Solving Process 20](#_Toc492845880)

[Figure 2. 3: DOL Tree Example 21](#_Toc492845881)

[Figure 2. 4: System Architecture of GWPS: Geometry Word Problem Solver 23](#_Toc492845882)

[Figure 2. 5: System Architecture of an RTE System 25](#_Toc492845883)

[Figure 3. 1: System Architecture of the System 34](#_Toc492845921)

# LIST OF TABLES

[Table 2. 1: Steps in Solving Computational Word Problem 13](#_Toc492846002)

[Table 2. 2: Example of Word Problem Grammar 17](#_Toc492846003)

[Table 2. 3: List of Schemas 19](#_Toc492846004)

[Table 6. 1: Raw Data Analysis Table 44](#_Toc494961494)

[Table 6. 2: Evaluated Data Analysis Table 46](#_Toc494961495)

# LIST OF NOTATION

|  |  |  |
| --- | --- | --- |
| **Acronyms:** | | |
| **IES** | | Information Extraction System |
| **NER** | | Named-Entity Recognition |
| **NLP** | | Natural Language Processing |
| **NLTK** | | Natural Language Toolkit |
| **POS** | | Part of Speech |
| **RNN** | | Recurrent Neural Networks |
|  | | | |
|  |  |  | |
|  |  |  | |
|  |  |  | |
|  |  |  | |
|  |  |  | |
|  |  |  | |
|  |  |  | |
|  |  |  | |
|  |  |  | |
|  |  |  | |
|  |  |  | |
|  |  |  | |

# CHAPTER 1 The Problem and Its Background

## BACKGROUND OF THE STUDY

Computational word problems have always been a part of the learning exercises of an individual. It is noticeable that this type of problem opens a wide variety of real-world solutions. The idea of having word problem solving as a challenging task is not entirely because of the person’s mathematical skills, in fact, many studies have shown that it is really reading comprehension that makes it difficult (Dela Cruz & Lapinid, 2014). Moreover, other factors can be viewed as a key component that adds up to the person’s resistance in problem solving, such as language barrier, slow learning pace, abstract reasoning, and perceptual deficits (Lopez, 2008).

A successful evaluation of a word problem requires two general things; first, is to translate the word problem into a mathematical equation with its parameters, and second, to solve the equation using an appropriate solution based on what is required in the given problem.

Imagine if a computer can successfully perform these things. Having a computer to automatically translate a textual representation of a computational problem into a mathematical notation, and have that equation to be automatically evaluated and solved. By the time a computer knows how to understand and interpret this kind of problem, then it will be a step higher for the computers to solve general problems.

It is known that since the first creation of computers, technology have exceeded human beings in both speed and accuracy when it comes to mathematical computation. Yet, even until today, computer still lacks an algorithm that can fully solve computational problems that are described in the natural language (Shi, Wang, Lin, Liu, & Rui, 2015). Studies in designing an algorithm that automatically solves mathematical word problems can even be dated back since the early 1960s (Bobrow, 1964). Different approaches of previous studies in this area can be classified into two categories: symbolic approach and statistical approach (Shi, Wang, Lin, Liu, & Rui, 2015). In symbolic approach, texts are analyzed using its grammatical structure by pattern recognition. Most studies that use symbolic approach suffers when it comes to the semantic level of analysis, but it is observed that the system can solve word problems accurately considering if the structure of the input matches the structure of the text corpora that is used. On the other hand, statistical approach makes use of probabilistic model to extract information in each input text. While this approach can be considered as a more flexible method than symbolic when it comes to semantic analysis, one major drawback of statistical approach is the ambiguity in parameter estimation and inference.

This study is designed to create a framework for computers to automatically evaluate, interpret, and solve a computational word problem, specifically in the domain of Physics, due to the reason that this domain reflects real-world scenarios. The system aims to extract important parameters in the word problem such as the given and the required. As well as to analyze the semantics of the text for the formulation of the appropriate solution to be applied. The system will be outlined to perform different disciplines of Natural Language Processing for syntax and semantics analysis. Additionally, Recurrent Neural Network will be used for the identification and extraction of the relevant parameters and problem classification. The use of Recurrent Neural Network in the field of information extraction can be considered as both symbolic and statistical approach, which is applicable for both syntactic and semantic level of analysis.

The study is expected to address the collective limitations of other related studies. In the paper of (Apostol & Orsolino, 2016) and (Leszczynski & Moreira, 2017), it is observable that the design of the system mainly focuses on using specific grammar for generating word problem, as a result, the word problems that should be entered is strictly advised to comply with their expected grammar. Additionally, the intent of this paper is to create a system that will identify all the important parameters considering that noise parameters exist in the input, likewise, the system will avoid arbitrary selection of parameters that is encountered in the paper of (Sundaram & Khemani, 2015). The same shortcoming can be recognized in the paper of (Kushman, Artzi, Zettlemoyer, & Barzilay, 2014), wherein in their experimental setup, the dataset, which came from a crowd-source tutoring website, still requires to be filtered for nonessential information such as word and phrases that are not necessary for the given word problem. To illustrate, a sample input is shown below.

*Input:* Jairus walks at a speed of 4 kmph, while his friend, Edil, walks at a speed of 6 kmph. How much time does Jairus will take to walk a distance of 20 km?

In the given example, it is noticeable that to be able to solve the word problem, the system must identify that the required parameter to be used is the value of the entity that the variable *Jairus* entails and not the entailment of the variable *Edil*. To be able to achieve this specific objective, the given word problem must be filtered automatically, extracting only those entities that are necessary in relevance with the required. This is where the RNN-based information extraction will take place. The system will be using two Recurrent Neural Network. The first neural network will be used to classify all the candidate entities that are under the domain of Physics. Then the second neural network will identify all the parameters needed based on what is required in the given problem.

The expected output of the system will be divided into two classification: the initial parameters and the end parameters. The initial parameters will be the Given and the Required while the end parameters will be composed of the Formula, Solution, and the Final Answer.

The purpose of this study is to design a system that will simulate how a human being solves a computational word problem. The study is expected to introduce a new method for a machine to solve these problems. By analyzing the challenges that a human being faces when it comes to problem solving, the researchers will align the different concepts and principles under Machine Learning to be able to make a machine reason and create logic almost the same way a human can.

## STATEMENT OF THE PROBLEM

The study aims to translate a Physics word problem into its corresponding numerical representation. As well as to extract important parameters in the given problem using RNN-based Information Extraction. Specifically, this study seeks to answer the following questions.

1. What is the level of accuracy of the system in implementing RNN-based Information Extraction in Physics word problem. In terms of the following parameters:
   1. Given
   2. Required
2. What is the level of accuracy of the system in evaluating and solving the given Physics word problem. In terms of the following parameters:
   1. Formula
   2. Final Answer

## THEORETICAL/CONCEPTUAL FRAMEWORK

## THEORETICAL FRAMEWORK

The study is consisted of different principles and concepts of Machine Learning, primarily Natural Language Processing and Artificial Neural Networks. The following are the concepts that support the research of the study.

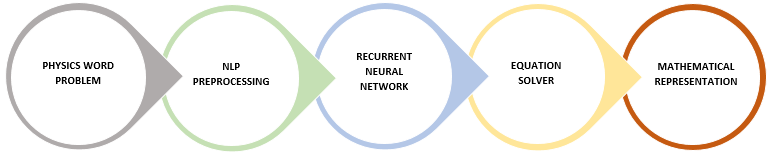


Figure 1. 1: Theoretical Flow of the System

* **Natural Language Processing**

Natural Language Processing (NLP) can be defined as the automatic processing of human language. It relates to formal language theory, compiler techniques, theorem proving, machine learning, and artificial intelligence (Copestake, 2004). To understand human language is to understand not only the words, but the concepts and how they’re linked together to create meaning (Kiser, 2016).

The main process that is going to be applied in the system under Natural Language Processing are the following: Tokenization, Part-of-Speech Tagging, and Named-Entity Recognition. These are the fundamental NLP disciplines that are substantial in Information Extraction. The NLP part in this study is done during the simplification and before the analysis phase. The aim of the simplification phase is to make the text amenable for analysis. This is done by breaking the stream of input texts into words that will be subjected to subsequent processing. For most situations, tokens are split based on a set of delimiters. These delimiters are frequently whitespace characters. Each word will then be labeled and tagged based on their grammatical disposition. POS Tagger assigns part of speech to each word and other tokens, and Named-Entity Recognition will locate and classify all the named entities available in the text. Figure 1.2 shows an architecture for a simple information extraction system.

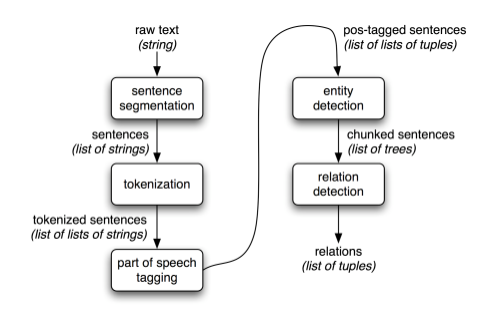


Figure 1. 2: Simple Architecture for an Information Extraction System

* **Neural Networks**

To be able to understand Recurrent Neural Networks, one must have to understand the basics of Feedforward Neural Networks. These networks are both named in relation to the way they channel information through a series of mathematical operations performed at the nodes of the network. One feeds information straight through, while the other cycles it through a loop.

In feedforward networks, the inputs are fed into the network, and with supervised learning, the output would be a label, a name applied to the input. In this way, input can be categorized or classified base on their patterns. Figure 1.3 shows a general model for a Feedforward Neural Network.

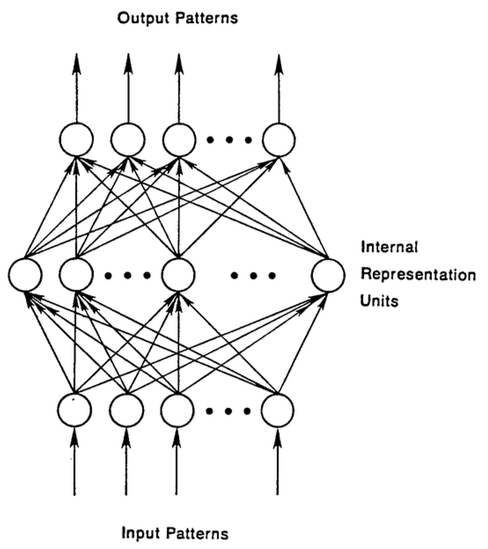


Figure 1. 3: General Model for Feedforward Neural Network

Recurrent Neural Networks, on the other hand, feeds the network with both current input and previously perceived input. In this concept, a Recurrent Neural Network’s decision at current time will affect the decision at one time later. Thus, RNN will always have two sources of input, the present and the recent past, in which in that case, the collective data will determine how the system will respond to new information. Feedforward networks and Recurrent networks almost process the same way, the only difference is that, the output in the Recurrent network will be stored as a memory to be used as a future input. The sequential information is preserved in the Recurrent network’s hidden state (Skymind, 2017). Figure 1.4 shows a general model for a Recurrent Neural Network.

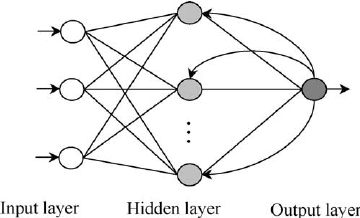


Figure 1. 4: General Model for Recurrent Neural Network

The concept of a Recurrent Neural Network can be visualized in the model of (Elman, 1990) that is shown in Figure 1.5. Where the BTSXVPE represents the input example in the current moment and the Context Unit represents the output of the previous moment. The forward arrow denotes a complete set of trainable connections from each sending unit to each receiving unit in the next pool. While the backward arrow from the hidden layer to the context layer denotes a copy operation.

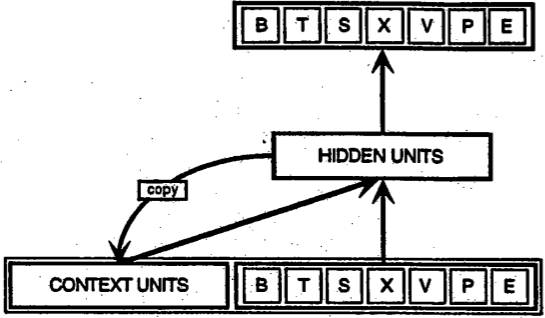


Figure 1. 5: Model for a Simple Recurrent Network

## CONCEPTUAL FRAMEWORK

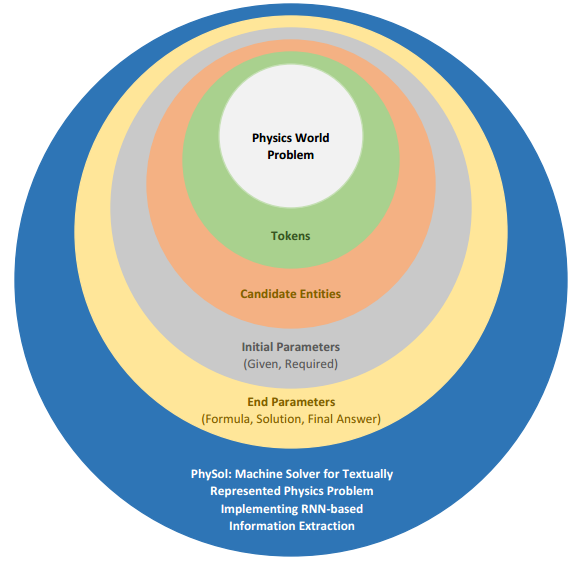


Figure 1. 6: Conceptual Framework of the Study

Figure 1.6 shows the relationship and flow of the variables in the system. The system will take a Physics word problem as an input, thus it will be the independent variable. The dependent variable, on the other hand, are the Overall Parameters, which are divided into two classifications, Initial Parameters and End Parameters. To be able to achieve the desired dependent variable, the input will undergo different methods such as NLP Preprocessing and RNN-based Information Extraction. As a result, these processes will produce the Initial Parameters, which includes the *Given* and the *Required*. Through an Equation Solver the Initial Parameters will be derived to produce the End Parameters, which is compose of the Formula, Solution, and the Final Answer.

## SIGNIFICANCE OF THE STUDY

The objective of the study is to develop a system that automatically evaluates and solves a Physics word problem. The overall outcome of the study will benefit the following individual that are involve in computational word problem, specifically in Physics.

Primarily, the main significance of this study will mostly affect the other existing knowledge in the area of machine learning. Solving computational word problem is just an example of the vast open problems in the field of computing. Computers, even until at the present time, still lacks an algorithm that can fully understand computational word problems that are describe in the natural language. As for machines who can only understand one’s and zero’s, information represented by the natural language are considered as data in an unstructured form. To sum it up, the study can add a significant contribution to those professionals that are inclined in the field of Computational Linguistics and Information Extraction, the study will be utilizing Recurrent Neural Network using both symbolic and statistical approach, in which as of now the two most effective method in machine learning.

In the success of this study, the application of the methodology may not only be beneficial for individuals in the field computer science.

In the academic perspective, the system can be beneficial for both students and professors for validating their evaluation of a certain Physics word problem. Similarly, it can act as a guide for those who requires additional effort in comprehending computational word problem in the domain of Physics.

In a similar manner, Linguists and Psychologists can also gain benefit in the study for the reason that the study can manifest some correlation between mathematical skills, grammatical structures, and reading comprehension in solving computational word problems.

## SCOPE AND LIMITATION OF THE STUDY

The main focus of the study is to solve and evaluate Physics word problems in a monolingual corpora, specifically English. These Physics word problem includes problems in Electricity, Forces, and Motions (Kinematics), which will be the initial scope. To be specific, the system is bounded with the following Physics area: Voltage, Current, Resistance, Power, Newton’s Law of Motion, Speed, Velocity, and Acceleration. The Physics word problem should only have not more than one (1) explicit *Required*.

Incorrect spelling of parameters and keywords can be accepted by the system, considering that the consistency of the incorrectness is observed. It is assumed that all input is grammatically correct.

The system will be using the Stanford NLP tools for the Named-Entity Recognition and POS Tagger. As well as TensorFlowTM to train the Recurrent Neural Network.

The extent of the study will only be in the Philippines. Likewise, the study will be conducted in the Poltechnic University of the Philippines, Sta. Mesa, Manila.

## OPERATIONAL TERMS

**Computational Word Problems** – A mathematical problem that is describe in the natural language in which it will be the input of the system.

**End Parameters** – The collective name for the *Formula*, *Solution*, and *Final Answer*, which are part of the dependent variables and the output of the system.

**Information Extraction** – The implementation of the Recurrent Neural Network in identifying the overall parameters.

**Initial Parameters** – The collective name for the *Given* and *Required*, which are part of the dependent variables and the output of the system.

**Natural Language Processing** – A preprocess of the system which aims to make the text amenable for analysis.

**Overall Parameters** – The collective name for the Initial and End Parameters. The complete output of the system.

**Physics** – The main domain of the input of the system.

**Recurrent Neural Network** – The process used to evaluate the Physics word problem, to be able to transform it into its corresponding mathematical representation.

# CHAPTER 2 Review of Related Literature



## RELATED LITERATURE and STUDIES

* + 1. **Computational Word Problem (Human)**
       1. **Reading Coaching for Math Word Problems**

According to Phyllis and David Whitin(2000), “Math is language, too”. Young students nowadays have challenges in comprehending math word problems, even though they know what mathematical operations to perform. The reason for this problem is they can’t comprehend what the problem is actually asking them to do, because young readers is distracted by everyday language, math words, or combination of both. Those who struggles in reading comprehension and the math computation faces the biggest challenge. The researchers make use of the fourth-grade math questions because that grade is a crucial part in student’s life in school, learning in this stage establishes the foundation of learning into upcoming school years.

It is challenging for students who are solving math word problems when they are facing or reading words that are unknown or unfamiliar to them.

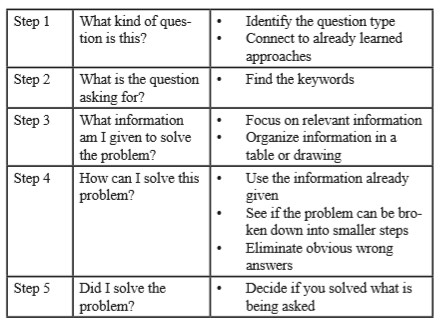
*“Haley swam 22 laps each day for 18 days. Then she swam 25 laps each day for 10 days. What was the total number of laps she swam over the 28 days?” (Massachusetts Department of Education, 2006)*

Some students may not know the word ‘swam laps’ which makes them have difficulties in solving this problem. The strategy is to ignore unfamiliar or confusing words and try solving the problem with the words the child knows (Edward, Maloy & Anderson, 2009). In above example question, a young individual can ignore the words swam and laps. The readers need to recognize that Haley did something 22 times for 18 days, and 25 times for 10 days then adding the product of these two then you have the final answer.

Word problems include proper names that may be unfamiliar for young readers. Proper names like people names may distract readers to the essential mathematical information in the problem. When young readers are solving word problems, if they encounter an unknown word, the easily classify this as a name. That’s why when readers get confused with these names, teachers encourage them to substitute them with the name of their parents, friends, brother, etc. Then it is easier now to comprehend the given word problem.

Another thing that young reader faces is the sentence structure and syntax, because word problems are written in compositional and not in conversational English. Mathematician George Polya’s (1973) classic problem solving approach use of a model that teachers can teach the students how to properly comprehend word problems.

Table 2. 1: Steps in Solving Computational Word Problem



Polya’s framework need the readers first to determine the type of problem, then the next thing they need to look is what is the problem trying to find. Then look for the given relevant variables in the problem, from this reflect how you solve the problem with the given data. Then finally ask if you have satisfied the problem is trying you to look for.

Math terminologies makes it difficult for young readers to solve the word problems because of they need to recall mathematical terms and how to solve them. For example, total, odd and even, greatest and least, etc. First strategy here is constantly teach the student how to solve these mathematical terms. Second strategy is to encourage students to formulate their own problems to familiarize them in math terms.

Another challenge is problems that have multiple math operations, which hinders students to understand how to formulate solution to the given problem, the strategy here is that change word variables with another name that a student can relate too (e.g. basketball points, scores). Charts, graphs and visual about the question also makes the problem for the students uncomprehensive, the solution is to make charts and graphs about different topics so that students can ask their friends and family.

Math word problems is not just about mathematical computation, it needs the use of literacy and proper comprehensive. Literacy coaches and teachers need wide-ranging strategies in order to support children as they improve their skills in reading and mathematics (Edward, Maloy & Anderson, 2009).

* + - 1. **Making Sense of Word Problems**

In the early academic institution, such as grade school, arithmetical word problems are being solved using “pure mathematical” approach, sometimes this can be unrealistic. In this book, the contrast between the student's traditional “pure mathematics” approach on solving word problems and on the other hand, the more realistic approach which is based on student’s daily experience and knowledge for ways of improving the method of solving word problems.

Criticism focuses on the fact that the current practice with word problems in school mathematics does not support student’s tendency to use their everyday life knowledge (Verschaffel, Greer and de Corte, 2000). Authors shows numerous examples of unrealistic solutions and answer to word problems from literatures and author’s observation. Some examples show nonsensical questions (How old is the captain?), while some have very unrealistic background (“How much to post a letter?), this sums that students’ school mathematics does not have any connection to their real-life experiences. Students tend to simply apply formulas and solutions sequentially or algorithmically with neither realistic considerations nor the use of use sense making. The paper tested two approach, they given questions that can be categorize as questions that can be solved by obvious arithmetic operations and questions that can be solved only by using common sense. The result shows that the general number of students does not apply the relationship to reality style. Researchers confirmed that due to personalization, solvers are more successful when the word problems relate to themes and context originated around familiar events, people or activities to the students. The main problem in the traditional approach is that this method is taught by the teachers every grading stage. The key proposal to this is to adopt a modelling perspective to allow students and teacher a rich interaction which will make a new approach in solving word problems.

* + - 1. **Students’ Difficulties in Translating Worded Problems into Mathematical Symbols**

This paper presented by (Dela Cruz & Lapinid, 2014) shows the different difficulties encountered by students in translating worded problems into mathematical equations. In the discussion of the study, it is stated that the failure of the student to translate the given problem can be rooted out in six different problems. The following are the major problems that makes this problem solving a challenging task: misinterpretation of the problem, lack of comprehension of the text, incorrect use of operation, carelessness when it comes to parameters, interchanging value of the variables, and difficulty with unfamiliar words.

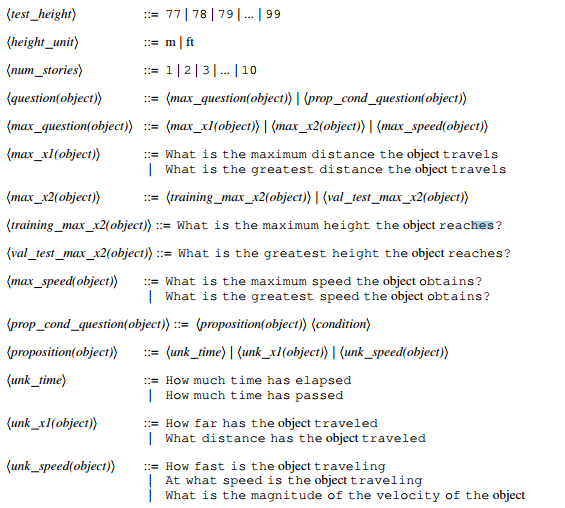
Having this paper into consideration, the study will be reflecting these aforementioned factors in designing the method that is going to be applied in the system. The system will be designed to simulate how a human being solves these computational word problems, thus taking these factors into account will generate a more favorable result.

* + 1. **Computational Word Problem (Machine)**
       1. **Machine Solver for Physics Word Problems**

A complete system architecture for a machine solver is presented by (Leszczynski & Moreira, 2017). The main focus of this paper is to automatically solves Physics word problems, namely classical mechanics of a point particle in free fall. Given a word problem as an input, the solver extracts the parameters and identifies the corresponding type of question involve in the input sentence. Two independently trained recurrent neural networks are used to complete these tasks. One neural network, referred to as the labeler, is used to identify all the parameters and locate the question within the problem. And the other neural network, referred to as the classifier, identifies the type of question. Then a numerical integrator is used to compute the evolution over time of the word problem.

A specific grammar is used to restrict the structure of the word problem as an input. It ensures that the training set is disjoint from the validation and test sets. It is vital in assessing the ability of the trained networks to generalize. An example of the grammar that is used is shown in Table 2.2.

Table 2. 2: Example of Word Problem Grammar



Such limitation in this study is observed, the domain of the word problem that is used is very centralize in a specific area in Physics, namely classical mechanics of a point in free fall. While it can be seen as effective in solving the word problem in this particular area, testing semantic ambiguities can’t be perform with the method that is used in the study. The problem when it comes to using a specific grammar in Natural Language Processing is that the input it very much restricted. It is a requirement for the input to be subjected to the particular grammar that is used, which means, failure of the input to comply with the structure of the grammar, will result to a rejected input, likewise, only specific inputs can be accepted.

* + - 1. **Natural Language Processing for Solving Simple Word Problems**

A study conducted by (Sundaram & Khemani, 2015) describes a system that solves simple arithmetic word problems. It takes a computational word problem that is described in the natural language, extracts information required for representation, and then perform derivation for the answer. The main focus of the study is the used of the different Natural Language Processing techniques to retrieve relevant information from the given word problem.

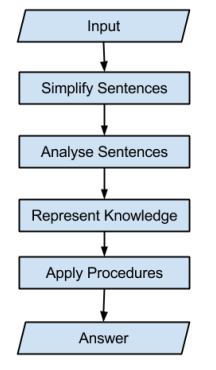
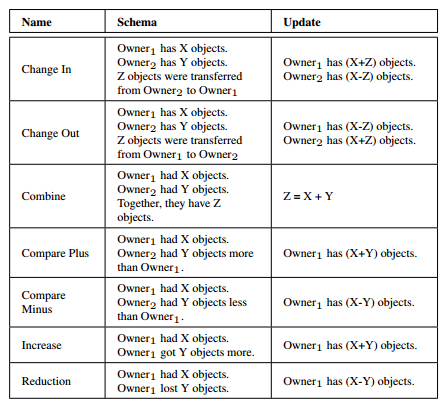


Figure 2. 1: System Architecture of Word Problem Solver

The architecture of the system is reflected in Figure 2.1. The computational word problem requires simplification such that it should be the case that only one verb should be contained in every single sentence. Then, information extraction will be performed, linguistic information is extracted from each sentence and passed on for the Knowledge representation. This knowledge representation makes use of a schemas, see Table 2.3 for example, to described how entities interacts. To elaborate, each sentence is examined sequentially until a keyword is encountered. Consider the following sentence, “John had 3 apples. He forfeited 1 apple. How many apples does he have now?” In this example, the sentence will trigger the “Transfer-in-Ownership” schema because of the keyword “forfeit” that maps to that particular schema. In the schemas that the researcher of the study had designed, no sentence is explicitly matched against a template. Whenever a keyword is encountered, the information is adjusted according to its corresponding procedure. The failure of the schema to be available in reference to a particular information will introduce a new variable. If the value that corresponds to that variable is seen later, it is replaced in all the expressions that contain the variable.

Table 2. 3: List of Schemas



During this phase, all the constituent sentences are taken into consideration wherein events are ordered according to time. Once all knowledge is extracted and has been represented, the problem will be solved using the procedures stored in the system, Figure 2.2 shows an example that depicts how the particular problem is solved.

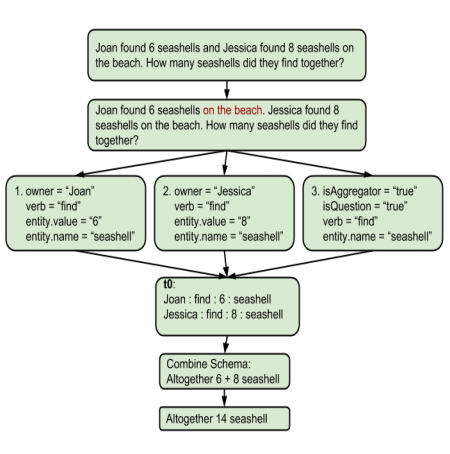


Figure 2. 2: Example of the Problem-Solving Process

Though the study obtains a result that is more than favorable, it is very limited when it comes to its context. One major limitation of the study is that the semantic analysis for the units are not well defined. Although this may seem to be a minimal concern in the context that the study used, other area in computational word problem requires analysis even when it comes to units. Take for example the unit “kmph” or “kilometer per hour”, in the area of kinematics, this unit can be analyze as a parameter which entails two values, a value for time and a value for distance.

* + - 1. **Automatically Solving Number Word Problems by Semantic Parsing and Reasoning**

This paper presented by (Shi, Wang, Lin, Liu, & Rui, 2015) reflects an automatic solver for mathematical word problems using semantic parsing and reasoning approach. The study implements a CFG parser based on 9,600 semi-automatically generated grammar rules. The computer system is entitled SigmaDolphin using a meaning representation language called DOL, abbreviation of Dolphin Language, it uses DOL as the structured semantic representation of the natural language text. A semantic parser is used to transform computational word problem into DOL trees, see Figure 2.3 for example. Then, a reasoning module is included to derive the mathematical expressions from DOL trees and to calculate the final answer.

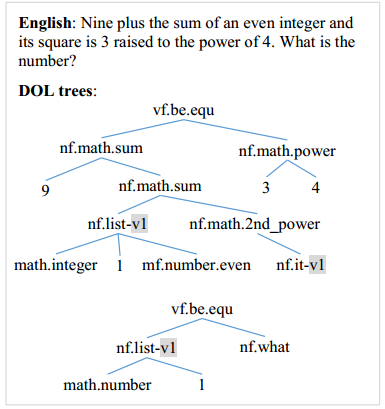


Figure 2. 3: DOL Tree Example

The study claims that the system showed improvements over previous symbolic methods in the following ways. One, is that they have introduce a systematic way of parsing natural language text, based on context-free grammar (CFG). And two, the evaluation is enhanced in terms of both data set construction and evaluation mechanisms.

To summarize the approach, the system architecture of the study contains three modules. A *meaning representation language* called DOL as the semantic representation of natural language text. A *semantic parser* which transforms natural language sentences of a math problem into DOL representation. And a *reasoning module* to derive math expressions from DOL representation.

The experimental results reported in the study are only number word problems. And the researchers of the study noted that general math word problems are beyond the capability of the system due to the approach’s entity types, properties, relations, and actions. The study suggests to extend the approach to general math word problems by expanding the coverage of the DOL language and the CFG grammar.

* + - 1. **GWPS: Geometry Word Problem Solver**

This paper presented by (Apostol & Orsolino, 2016), reflects a system that formulates equation, solution, and solves answer automatically given a mathematical word problem, specifically in the domain of Geometry.

The system architecture, shown in Figure 2.4, is composed of the following concepts; Natural Language Processing, Information Extraction, and the Equation Solver. The system contains a database in which it stores previous word problems that are already been solved. If the given problem is not present in the database, then it will proceed to the other processes. The main basis of the process is the named-entities that are existing in the text, which are the Geometry related keywords. Then the keyword will be used to identify the corresponding formula to be implemented.

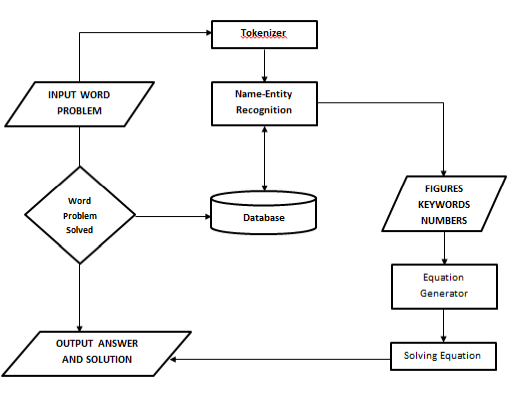


Figure 2. 4: System Architecture of GWPS: Geometry Word Problem Solver

The overall accuracy of the system in solving the word problem gains an 86.65% score. Although, the study has obtained a favorable result, it is observed that the study is very much limited when it comes to semantic analysis. The system is expected to function in a very limited context since the process in information extraction only involves NLP preprocesses, namely Named-Entity Recognition. Furthermore, the researchers of this study explicitly noted that the system can only accept specific structure of the grammar, in a more specific manner, it is a must to have the input to locate the parameters in a fixed position in the text, wherein all the parameters are required to be followed by the system’s defined keywords.

* + 1. **Natural Language Understanding**
       1. **Applying CRFs and SVM to Textual Entailment Recognizing**

Textual entailment is a useful for systems that includes inferencing over sentences in natural language. Entailment knowledge is important in any knowledge-based systems. The Recognizing Textual Entailment(RTE) in a simple sense is if you are given two text or phrase, the system should recognize if the either one of the text can be entailed, i.e. deriving or inferring one text from the other text. There are few challenges in developing a system that focused in RTE. One problem is the incompleteness of sentences, incorrect spelling, grammar error, and abbreviations, which will make a problem soon because these errors and abbreviations could mean something or anything else making it hard for the system to recognize entailments. Second if some parts of the text are written in the context but the references or sources of information is defined explicitly or implicitly in reference. dates, places, pertaining to the corpus. RTE defined as the direct relationship or connection between two given text (Yongmei & Hovy, 2014). Support Vector Machine aims to seek or make decision boundary to separate the given training data into two classes and their corresponding classification. It is actually based in the structural risk minimization principle of computational machine learning theory. The input of the SVM is a training examples and the machine will try to find the classification of every training data and maps them to classes. The structural risk minimization aims to find the hypothesis that the researchers can rest assure that the lowest probability bound for generalization error. While the Conditional Random Fields (CRF), it gets the strong independence assumptions of HMM and solving the problem of label-bias problem of the MEMM. To model that real-world data that the conditional probability of a sequence can depend on features of the observation sequence.

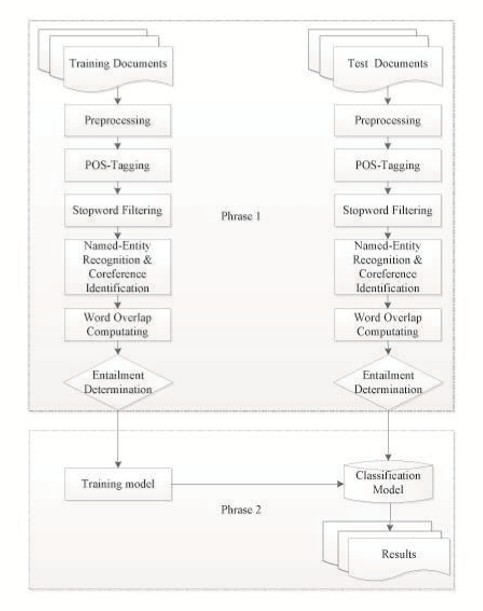


Figure 2. 5: System Architecture of an RTE System

There are preprocesses that are involve before performing the textual entailment recognizing. This includes converting uppercase words and letters to lower cases, removing stop words (the, to, etc.) because they have little lexical content, they also remove words with length 1, there are also named entity recognition which will give identification to words. After the preprocessing procedure, the classification starts. The use of machine learning methods in recognizing entailments between a text and a hypothesis is effective however the dataset’s nature will help to achieve higher accuracy as a feature in testing (Yongmei & Hovy, 2014).

* + - 1. **Combination of Rule-based and Machine Learning for Biomedical Event Extraction**

According to (Pham & Ho, 2014), Biomedicine has been attracting researchers in the field of natural language processing because of the demand of automatic processing of large volume of papers, test, literature and documents, and the main task here is biomedical entity recognition. These entities are proteins, genes or disease names (Saha et al 2012). There are other researchers that extracts information concern with protein-protein interactions, gene-disease and drug-protein relations. The proposed approach is a hybrid approach in extracting event from biomedical literature. This will make use of semantic rule-based and machine learning classification. The approach will be divided in two phases, first is with the use of linguistic information from syntax tree to create two kinds of rules. The second phase is reusing the features to know what is the shortest path between trigger and arguments, and again some important features as token contains both protein and trigger (Amami et al. 2012).

The given text will have text preprocessing, with the use of NLP tools available in the market the system could do tokenization, POS tagging and parsing, and this information will be used to recognize event trigger and for each candidate event, its argument will be identified (Pham & Ho, 2014). In event trigger detection, only words that have POS tags will be evaluated, with the use of simple dictionary from training data, this will determine if these tokens are triggers set. The next stage is the event detection, in this stage, (Pham & Ho, 2014) combined the rule-based semantics and machine learning classification in detection. The rule based task is used to extracts events by combining a trigger with appropriate arguments in the right context. There are two kinds of rules applied, the first kind of rule will be based from the training corpus, the second kind of rule is based on POS tagger. After this rules have been applied to extracts events next stage is to use machine learning, the researchers make use of Support Vector Machine (SVM). The SVM classifier ith a linear kernel is used to classify the relation between two objects (trigger/protein or trigger/event) in the sentence o given event class (Pham & Ho, 2014). The preprocessing will recognize complex events based on the detected events.

Overall, the system’s approach of combining semantic rule-based and machine learning achieves not so high but it encourages the efforts to use of hybrid approach.

* + - 1. **An Experiment in Semantic Tagging using Hidden Markov Model Tagging**

Understanding the meaning of a word has been a challenge to psychology and artificial intelligence research communities. The use of Hidden Markov Model in part of-speech tagging has been proven to be a useful approach in eliminating language ambiguity. The idea of semantic tagging is to add word sense markings so that it can be used to process in some automatic processor of languages to choose the proper meaning of words in a given context. Researchers can agree that semantic tagging is much more difficult than performing part-of-speech tagging. But because there are approaches that are proven to be effective in part-of-speech, the researchers experimented how these approaches behave when used in semantic tagging. They have used the WordNet’s 45 semantic tags, these got some advantages, first the size of the semantic tag is scalable, meaning the training data does not have to be large, secondly the semantically tagged corpora has been provided also. The part-of-speech tagging is easily understood than semantic tagging, it is because semantic tagging has no consensus or general representation or agreement while the POS tagging has general understanding and ways of interpreting. But despite the lack of consensus, the WordNet team takes the challenge to construct general semantic tagging scheme. These 45 semantic tags consist of 3 tags for adjectives, 1 for all adverbs, 26 tags for nouns and 15 tags for verbs. It is a fact that HMM approach in POS tagging is very promising, so the researchers try the HMM approach on semantic tagging if it will result to higher accuracy in tagging meanings to the context. If using the Hidden Markov Model, there are preparation activity. This preparation includes making a lexicon which contains all possible semantic tags. Also, the researchers need to prepare training corpus and a test corpus with the same size. After this compute the HMM model based on training sets by comparing the tags by semantic tagger and in the test corpus. They include three major test, the first test’s test corpus the overall accuracy was 86% and the accuracy over ambiguous tokens of 71%. The second test, they experimented with the use of POS pre-tagging, by using this, the number of ambiguity classes decreases, with this the overall accuracy of 89% but the accuracy over ambiguous tokens is identical at 71%. In the third test, they assigned the most common part of speech for each lexicon which got the overall accuracy of 90%. Researchers find it surprising that the processes and techniques to improve the accuracy of the POS tagger when applied to semantic tagger, the overall accuracy also improves.

* + - 1. **Representation of Knowledge in a Program for Solving Physics Problem**

Any word problem can be solved efficiently through the use of modelling the problem itself. The need for a machine that solves problems automatically are rising in today’s world, numerous than ever. A computer program that solves physics word problems that is stated in English has been made in this regard. The sentences of the physics word problem are transformed into semantic network form to generate objects that are from canonical object frames. The use of specialized representation using the procedural knowledge that is required to convert one representation into another because of some information is unspecified implicitly by the source representation, with this, it simplifies many of the processes which must be performed by the program. The program is called ISAAC, it is able to read, understand, solve, and generate objects based on the problem stated. The program first needs to understand the English sentence of the problem. This needs the parsing of various semantic parser network form, which helped by large number of semantic programs. One of the important semantic process is referent identification, this identify the relationship between objects based on developing model of the problem. The next major step is to identify the type of conceptual entity, such as the location of the object, its attributes, etc. The final step in the semantic process is the execution of the verb semantics, the arguments of the verb are represented as semantic frames, this causes the transfer of new information to existing objects in the internal model of the problem. After all the semantic processing, the next step is to process by the language-free internal model, this model makes the representation of the objects such as ladders, tables, ropes, the person, also its features, and their relationship. Then this objects are transformed into Canonical Object Frame, which is as idealization or abstraction of certain features of an actual object that is the behavior of real objects. This representation is used as a reference to make a geometric model that is a model for representing the positions of the objects in the problem to a common coordinate system. Combining geometric models of course needs the features or attachment to each other of the objects. The program solves the problem in the formal system and returns the result back in its original form. After the canonical objects have been generated from geometric models, equations are written based accordingly on physical laws. The last step is the generation of the picture and the diagram of the problem, this includes reasonable sizes in the drawing object. The program proven that the notion of canonical object frame is a powerful technique for constructing problem systems (Gordon, Novak). Also, the use of canonical object frame makes the problem search space smaller because the irrelevant information is not being accessed.

* + - 1. **LSTM Neural Networks for Language Model**

Any natural language needs a Language Model because this is the core component that incorporates syntactical and semantical constraints (Sundermeyer, Schluter, H. Ney, 2012). Feed-forward networks only predict a fixed length of words of a sequence, so the researchers use recurrent neural network LSTM or Long-Short Term Memory to solve this problem. Unfortunately, recurrent neural networks are hard to train using BPTT or Back Propagation through time. These can be improved the training by making efficient optimization algorithms.

Neural Network has a problem so called “vanishing gradient”, this is the situation where the gradient that is propagated through time either decays or grows exponentially. With this, the gradient will affect the RNN’s training by either dominates the next weight or effectively get lost. The basic principles behind neural network language model are: the input words are encoded 1-K, K being the number of words in vocabulary; the output layer uses softmax activation function; the training criterion is the maximum likelihood. According to the paper, they use sigmoid activation function, as a result they have achieve the perplexity lower than standard recurrent neural network by 8%.

They have also observed that using two concatenated sentences will have small improvement of the LSTM. The perplexity significantly increases when the context length exceeds a certain limit, with this smearing of inputs will obtain the best perplexities hen LSTM are combined with this projection.

* + - 1. **TEXTRUNNER: Open Information Extraction from the Web**

Traditionally, Information Extraction relied on human involvement in the form of hand crafted extraction rules or tagged training examples. Also, users are also asked to explicitly specify the relation of items. It will be possible for the users to query over heterogeneous corpora, IE must move away from architectures that require relations to be specified prior to query time in favor of those that aim to discover all possible relations in the text (Banko et al., 2017). The main problem with incorporating web as a platform for Information Extraction is that it has a vast corpus and the relations of interest are unanticipated.

In automating IE, the first step was to move knowledge-base into trainable systems that has hand-tagged instances (Riloff, 1996). Using the Corpus Heterogeneity on the web will make parsers and NER less accurate (Sundermeyer, Schluter, & Ney, 2012) because the corpus is different from training data. TEXTRUNNER are segmented into different working parts. First part is the Single Pass Extractor, which work is to make single pass on the given text and functions as POS tagger and noun-phrase chunker and then it acts as classifier whether or not to extract it. The classifier will be the second part of the TEXTRUNNER, in this part, the researchers will make the training data based on heuristic constraints and the classifier will use this training data and decides whether a sequence of POS tags are a correct extraction or not. The next part is the Resolver, which is assigned to make synonyms, because the extractor extracts many different strings with the same relation. The TEXTRUNNER also has a Query Interface which handles user inputs, if the User inputs “Newton” it will return (“Newton”, “invented”, “calculus”). Out of 9 million web documents, TEXTRUNNER successfully extracts 7.8 million well-formed tuples.

## SYNTHESIS OF THE STUDY

Different approaches of previous studies in the area of translating computational word problem into mathematical equation can be classified into two categories: symbolic approach and statistical approach. Both approach has a drawback in their respective method. Symbolic approach suffers when it comes to the semantic level of analysis, but it is observed that the system can solve word problems accurately considering if the structure of the input matches the structure of the text corpora that is used. On the other hand, statistical approach makes use of probabilistic model to extract information in each input text. While this approach can be considered as a more flexible method than symbolic when it comes to semantic analysis, one major drawback of statistical approach is the ambiguity in parameter estimation and inference.

One important key component of a system that translates computational word problem is its grammar. The particular language model that is going to be used will greatly affect the result of the other components. A very specific grammar is not applicable for system that accepts inputs in a vast domain. Likewise, a widely-designed grammar can cause ambiguous results due to its broadness in context.

In conclusion, this study will address the collective limitation of the methods of other studies. In particular, the study will design a method that uses the combination of the symbolic approach and statistical approach, to target the two advantage of the methods. And this method can be both utilize using Recurrent Neural Network. Furthermore, the framework of the system will be using a dynamic designed grammar. The study is proposed to design a language model that reflects the general structure of the natural language.

# CHAPTER 3 Research Methodology



## RESEARCH DESIGN

The researchers of this study generally used an experimental research design. After-Only without Control Design is going to be used in the system since the study attempts to know the significant impact of using Recurrent Neural Network based Information Extraction in solving Physics word problems. The Overall Parameters (Given, Required, Formula, Solution, Final Answer) are going to be the responding variables in which it will be dependent with the Physics word problem. The study proposed a test area of dynamic grammar model as the general model of the RNN to be used.

## SOURCES OF DATA

Physics is a language, in particular the language of a certain kind of worldview (Brown, 2013). Physics word problems can be acquired easily with the use internet sources with accessible websites and forums. However, the researchers will use published textbooks to obtain English physics word problems about electricity, forces, kinematics, work and energy, etc. See section 1.5 for the Scope and Limitations. These published textbooks are either from elementary physics or high school level physics.

Because of the reason that the study aims to comprehend and solve physics problems with no boundaries in the word problem structure, the physics word problem can be in any structure as long as the problem is within the given domain of physics and the problem is textually represented.

Building a dynamic system that interprets any given word problem in a certain domain will require the researchers to use probabilistic sampling technique, particularly stratified random sampling. In stratified sampling, the population is partitioned into regions or strata, and a sample is selected by some design within each stratum (Thompson, 2013). Stratified random sampling involves transforming heterogeneous population into homogeneous population, in our study, these heterogeneous population would be the 500 physics word problems and eventually this will be transformed to homogeneous domain-based partitions.

## INSTRUMENTATION

## SOFTWARE/HARDWARE TOOLS

### SYSTEM ARCHITECTURE

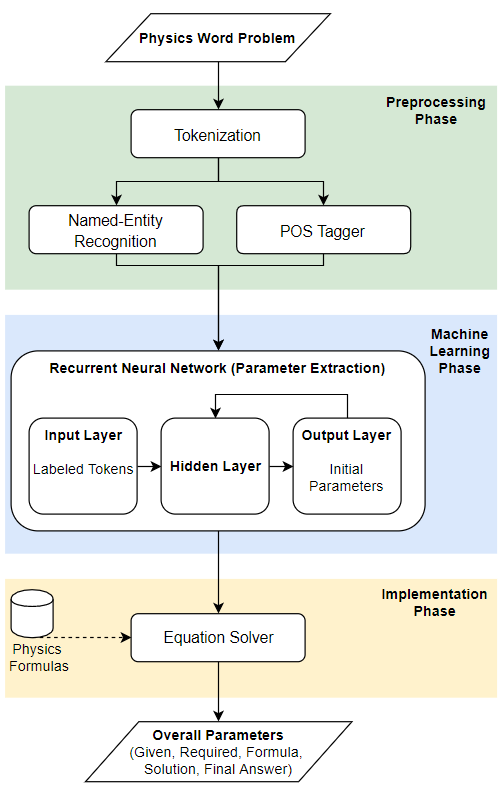


Figure 3. 1: System Architecture of the System

The input in the system is a Physics word problem.In the Preprocessing Phase, the input will undergo tokenization to segment the text into smaller units.The Named-Entity Recognition will classify all the named-entity that are present in the text, the NER will make use of the grammatical structure to statistically identify all entities. This process is done to make sure that the input is viable to be processed by the RNN.The POS-Tagger will tag all the words in text with their corresponding Part-of-Speech. The POS-Tagger will be used to aid the RNN to identify the relationship of each with words with their adjacent and related words.

The Machine Learning phase can be divided into two fundamental parts, the extraction for the *Given*, and the extraction for the *Required.* Recurrent Neural Networks will be used in both parameter extraction. The extraction of the *Given* will make use of all the named-entities that are identified in the preprocessing phase. These candidate entities are going to be tested in the RNN to evaluate whether the particular entity entails another entity. The RNN will take into account one entity at a single time and then the RNN will assign a probability to each word. The assignment of the probability is based on the grammatical structure of the sentence. The adjacent word that will obtain the highest probability will be the word that will be classified as the entailed entity. All the entities that will obtain an entailment will be denoted as a candidate parameter.The extraction of the *Required*will make use of the label of the POS Tagger that is done in the preprocessing phase. The system will only use the word whose label will be tagged as a “pronoun”, this is the common grammatical tag for basic keywords for questions. The failure to identify the *Required* in the “pronoun” tag will subsequently use the “verb” tag. All the words that receive the “pronoun” and “verb” tag will be a candidate parameter. These candidates are going to be tested in the RNN to evaluate whether the particular word can entail another word. The RNN will take into account one word at a single time and then the RNN will assign a probability to each word. The assignment of the probability is based on the grammatical structure of the sentence. The adjacent word that will obtain the highest probability will be the word that will be classified as the entailed word. The word pairing that will identified will be the *Required* parameter.

The implementation phase will make use of the identified *Required* as a basis on what candidate formula is going to be applied. A database of Physics formula will serve as a lookup table for this process.The *Formula* that is going to be used will be reflected base on what are the *Given* in relevance with the *Required*.Then a direct substitution will be applied to the *Formula*.The system will then derive the *Final Answer* by evaluating the values in the *Formula.*

The output of the system will be the Overall Parameter (Given, Required, Formula, Solution, and the Final Answer).

### DEVELOPMENT DETAILS

The researchers are going to be using Java programming language to develop the system. The database will be built in MySQL for storing dictionary, tokens and their corresponding POS tags and Named-Entity classifications, and Physics information. The proponents will be using Stanford NLP Group’s tools such as POS Tagger and Named Entity Recognizer to acquire the necessary information of the tokens to make it readable for the RNN to train the word problem. The researchers will be using Eclipse as a programming and testing platform for development process. The development of the tool is best partnered with Agile development model, where every module starts with planning, analysis, designing, developing then testing, after this, if there are necessary changes encountered in the testing phase, it will start in planning, this will continue until the module is ready for deployment and the tool is completely done. Also, using this development model will assure the quality of the tool and provide great communication among the proponents. The use of Recurrent Neural Network will require training stage for weights and biases of the network, the researchers will use TensorFlowTM to train the said neural network.

## 3.3.2 RESEARCH INSTRUMENT

The researchers will be using this experiment paper to show the statistical treatment of data result in every test. This paper will be used for measuring the accuracy in identifying the overall parameters in the given computational word problem. Furthermore, this paper will also be used to measure the performance of formulating the necessary formula for the problem and how the system will derive the problem systematically. The results of the experiment will be analyzed to assess the overall performance of the system.

## DATA GENERATION

The following steps will be performed in order to acquire the required data for the study before the research method will be implemented. The following includes the respective procedures:

1. The researchers will analyze the domain of the study, the problem, and the expected output.
2. The researchers will identify the corresponding concepts and theories to be applied in order to recognize the requirements for the study.
3. The researchers will be gathering important data for both testing and implementation phase.

* The Physics word problems that will serve as a training data for the system will be gathered from different academic sources mainly Textbooks and Workbooks.
* All the experimental input in the training process will be evaluated and solved manually by the researchers together with an expert in the domain of Physics for validation purposes.
* There will be no specific maximum number of datasets to be used but to make sure that the implementation of the training of the RNN will be utilized, at least 60 samples of Physics word problem will be implemented per specific Physics area.

## DATA ANALYSIS

The level of accuracy of the system in evaluating and solving the given Physics word problem will be measured using Precision, Sensitivity, Error Rate and Accuracy as the metrics. The metrics will be individually applied to the Overall Parameters of the system, such us *Given*, *Required*, *Formula*, and *Final Answer*, the parameter *Solution* is not going to be measured since it can already be represented by the *Final Answer*. The formulas for getting the Precision, Sensitivity, Error Rate, and Accuracy in identifying the Overall Parameters are as of follows.

**Precision** =

Where:

CP = total no. of correctly recognized parameters (Given, Required, Formula, Final Answer)

IP = total no. of incorrectly recognized parameters (Given, Required, Formula, Final Answer)

100 = the 100% ratio

**Sensitivity** =

Where:

CP = total no. of correctly recognized parameters (Given, Required, Formula, Final Answer)

UP = total no. of unrecognized parameters (Given, Required, Formula, Final Answer)

100 = the 100% ratio

**Error Rate** =

Where:

TP = total no. of parameters (Given, Required, Formula, Final Answer) to be recognized

IP = total no. of incorrectly recognized parameters (Given, Required, Formula, Final Answer)

100 = the 100% ratio

**Accuracy** =

Where:

100 = the 100% ratio

ER = Error Rate

**Overall Precision** =

Where:

x = Total Number of Physics Word Problem

P1 = Precision of Physics Word Problem 1

P2 = Precision of Physics Word Problem 2

…

Px = Precision of Physics Word Problem x

**Overall Sensitivity** =

Where:

x = Total Number of Physics Word Problem

S1 = Sensitivity of Physics Word Problem 1

S2 = Sensitivity of Physics Word Problem 2

…

Sx = Sensitivity of Physics Word Problem x

**Overall Error Rate** =

Where:

x = Total Number of Physics Word Problem

E1 = Error Rate of Physics Word Problem 1

E2 = Error Rate of Physics Word Problem 2

…

Ex = Error Rate of Physics Word Problem x

**Overall Accuracy** =

Where:

100 = the 100% ratio

OER = Overall Error Rate

# REFERENCES

Apostol, J. D., & Orsolino, A. Y. (2016). GWPS: Geometry Word Problem Solver. *Thesis*, 1-59.

Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., & Etzioni, O. (2017). Open Information Extraction from the Web.

Bird, S., Klein, E., & Loper, E. (2014). Natural Language Processing with Phython. In S. Bird, E. Klein, & E. Loper, *Natural Language Processing with Phython* (pp. 261-285). California: O'Reilly Media, Inc. Retrieved from NLTK.

Bobrow, D. G. (1964). Natural Language Input for a Computer Problem Solving System. *Thesis*, 1-127.

Copestake, A. (2004). *Natural Language Processing 2002-03.* Retrieved August 31, 2017, from University of Cambridge: https://www.cl.cam.ac.uk/teaching/2002/NatLangProc/revised.pdf

Dela Cruz, J. K., & Lapinid, M. R. (2014). Students’ Difficulties in Translating Worded Problems into Mathematical Symbols. *DLSU Research Congress* (pp. 1-7). Manila: De La Salle University.

Edwards, S. A., Waloy, R. W., & Anderson, G. (2009). Reading Coaching for Math Word Problems. *Literacy Coaching Clearinghouse*, 1-4.

Elman, J. L. (1990). Finding Structure in Time. 179-211.

Kiser, M. (2016, August 11). *Introduction to Natural Language Processing*. Retrieved August 31, 2017, from Algorithmia: https://blog.algorithmia.com/introduction-natural-language-processing-nlp/

Kushman, N., Artzi, Y., Zettlemoyer, L., & Barzilay, R. (2014). Learning to Automatically Solve Algebra Word Problems. *Association for Computational Linguistics* (pp. 1-11). Seattle: University of Washington.

Leszczynski, M., & Moreira, J. (2017). Machine Solver for Physics Word Problems. *ICLR 2017* (pp. 1-16). New York: IBM T.J. Watson Research Center.

Lopez, L. (2008). Helping At-risk Students Solve Mathematical Word Probles Through the Use of Direct Instruction and Problem Solving Strategies. *Master Thesis*, 1-147.

Mallinson, J. (2015). Modelling Syntactic and Semantic Tasks with Linguistically Enriched Recursive Neural Networks. *Thesis*, 4-18.

Morton, K., & Qu, Y. (2013). A Novel Framework for Math Word Problem Solving. *International Journal of Information and Education Technology*, 1-6.

Morwal, S., Jahan, N., & Chopra, D. (2012). Named Entity Recognition using Hidden Markov Model (HMM). *International Journal on Natural Language Computing*, 1-9.

Novak, G. S. (1977). Representations of Knowledge in a Program for Solving Physics Problems. *International Joint Conferences on Artificial Intelligence* (pp. 1-9). Cambridge: MIT.

Pham, X.-Q., & Ho, B.-Q. (2014). Combination of Rule-based and Machine Learning for Biomedical Event Extraction. *International Conference on Information Resources Management* (pp. 1-9). HoChiMinh: AIS Electronic Library.

Rillof, E. (1996). Automatically constructing extraction patterns from untagged text. *Proc. of AAAI*.

Segon, F., Schiller, A., Grefenstette, G., & Chanod, J.-P. (n.d.). An Experiment in Semantic Tagging using Hidden Markov Model Tagging. 1-4.

Shi, S., Wang, Y., Lin, C.-Y., Liu, X., & Rui, Y. (2015). Automatically Solving Number Word Problems by Semantic Parsing and Reasoning. *Conference on Empirical Methods in Natural Language Processing* (pp. 1-11). Lisbon: Association for Computational Linguistics.

Skymind. (2017). *A Beginner's Guide to Recurrent Networks and LSTMs*. Retrieved August 31, 2017, from DeepLearning4J: https://deeplearning4j.org/lstm.html

Sundaram, S. S., & Khemani, D. (2015). Natural Language Processing for Solving Simple Word Problems. *Thesis*, 1-9.

Tan, Y., & Hovy, E. (2012). Applying CRFs and SVMs to Textual Entailment Recognizing. *Journal of Information and Computational Science*, 1-9.

Verschaffel, L., Greer, B., & de Corte, E. (2000). *Making Sense Of Word Problem.* Lisse: Swets & Zeitlinger.

Yates, A., Carafella, M., Banko, M., Etzioni, O., Broadhead, M., & Soderland, S. (2007). TEXTRUNNER: Open Information Extraction on the Web.

# APPENDICES

## APPENDIX A: SAMPLE RESEARCH INSTRUMENT

**Experiment Paper of “PhySol: Machine Solver for Textually   
Represented Physics Word Problem”**

**Objectives:**

The following is the list of the objectives for conducting the experiment paper:

* To determine the level of accuracy of the system in implementing RNN-based information extraction in Physics word problem. In terms of the following parameters:
  + Given
  + Required
* To determine the level of accuracy of the system in evaluating and solving the given Physics word problem. In terms of the following parameters:
  + Formula
  + Final Answer

**Materials/Equipment:**

* Laptop with at least 2 GB RAM
* Experiment Paper
* Pencil or Pen

**Procedures**

1. Gather samples of the Physics word problem in which it will serve as the training data.
2. Feed in the training data into the system.
3. Once the system is already done, input the Physics word problem.
4. Fill in the table with a valid information based on the overall parameters that the system had generated.
5. The experiment is complete. The results are now ready for analysis, presentation, and interpretation.

**Guidelines:**

* For each item, complete the table below, table 6.1, based on the overall parameter that the system will generate.

Table 6. 1: Raw Data Analysis Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sample Number:**  1 | **Total No. of Parameters** | **Total No. of Correctly Recognized**  **Parameters** | **Total No. of Incorrectly Recognized**  **Parameters** | **Total No. of Unrecognized Parameters** |
| **Given** |  |  |  |  |
| **Required** |  |  |  |  |
| **Formula** |  |  |  |  |
| **Final Answer** |  |  |  |  |
| **Sample Number:**  2 | **Total No. of Parameters** | **Total No. of Correctly Recognized**  **Parameters** | **Total No. of Incorrectly Recognized**  **Parameters** | **Total No. of Unrecognized Parameters** |
| **Given** |  |  |  |  |
| **Required** |  |  |  |  |
| **Formula** |  |  |  |  |
| **Final Answer** |  |  |  |  |
| **Sample Number:**  3 | **Total No. of Parameters** | **Total No. of Correctly Recognized**  **Parameters** | **Total No. of Incorrectly Recognized**  **Parameters** | **Total No. of Unrecognized Parameters** |
| **Given** |  |  |  |  |
| **Required** |  |  |  |  |
| **Formula** |  |  |  |  |
| **Final Answer** |  |  |  |  |
| **Sample Number:**  4 | **Total No. of Parameters** | **Total No. of Correctly Recognized**  **Parameters** | **Total No. of Incorrectly Recognized**  **Parameters** | **Total No. of Unrecognized Parameters** |
| **Given** |  |  |  |  |
| **Required** |  |  |  |  |
| **Formula** |  |  |  |  |
| **Final Answer** |  |  |  |  |
| **Sample Number:**  5 | **Total No. of Parameters** | **Total No. of Correctly Recognized**  **Parameters** | **Total No. of Incorrectly Recognized**  **Parameters** | **Total No. of Unrecognized Parameters** |
| **Given** |  |  |  |  |
| **Required** |  |  |  |  |
| **Formula** |  |  |  |  |
| **Final Answer** |  |  |  |  |

**…**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sample Number:**  x | **Total No. of Parameters** | **Total No. of Correctly Recognized**  **Parameters** | **Total No. of Incorrectly Recognized**  **Parameters** | **Total No. of Unrecognized Parameters** |
| **Given** |  |  |  |  |
| **Required** |  |  |  |  |
| **Formula** |  |  |  |  |
| **Final Answer** |  |  |  |  |

* Complete the table for each samples of computational word problems covering each Physics areas (Voltage, Current, Resistance, Power, Newton’s Law of Motion, Speed, Velocity, and Acceleration)
* Each samples will be evaluated using the metrics in section 3.5. Fill in the table below, table 6.2, for the data analysis.

Table 6. 2: Evaluated Data Analysis Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sample Number:**  1 | **Precision** | **Sensitivity** | **Error Rate** | **Accuracy** |
| **Given** |  |  |  |  |
| **Required** |  |  |  |  |
| **Formula** |  |  |  |  |
| **Final Answer** |  |  |  |  |
| **Sample Number:**  2 | **Precision** | **Sensitivity** | **Error Rate** | **Accuracy** |
| **Given** |  |  |  |  |
| **Required** |  |  |  |  |
| **Formula** |  |  |  |  |
| **Final Answer** |  |  |  |  |
| **Sample Number:**  3 | **Precision** | **Sensitivity** | **Error Rate** | **Accuracy** |
| **Given** |  |  |  |  |
| **Required** |  |  |  |  |
| **Formula** |  |  |  |  |
| **Final Answer** |  |  |  |  |
| **Sample Number:**  4 | **Precision** | **Sensitivity** | **Error Rate** | **Accuracy** |
| **Given** |  |  |  |  |
| **Required** |  |  |  |  |
| **Formula** |  |  |  |  |
| **Final Answer** |  |  |  |  |
| **Sample Number:**  5 | **Precision** | **Sensitivity** | **Error Rate** | **Accuracy** |
| **Given** |  |  |  |  |
| **Required** |  |  |  |  |
| **Formula** |  |  |  |  |
| **Final Answer** |  |  |  |  |

**…**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sample Number:**  x | **Precision** | **Sensitivity** | **Error Rate** | **Accuracy** |
| **Given** |  |  |  |  |
| **Required** |  |  |  |  |
| **Formula** |  |  |  |  |
| **Final Answer** |  |  |  |  |
|  | | | | |
|  | **Overall Precision** | **Overall Sensitivity** | **Overall Error Rate** | **Overall Accuracy** |
| **Result** |  |  |  |  |