2018

# Artificial exam scorer for efficient marking and grading of short essay tests

Edmond O. Menya
*Faculty of Information Technology (FIT)*
*Strathmore University*

Recommended Citation

Menya, E. O. (2018). *Artificial exam scorer for efficient marking and grading of short essay tests* (Thesis). Strathmore University. Retrieved from https://su-plus.strathmore.edu/handle/11071/5997

# Artificial exam scorer for efficient marking and grading of short essay tests

**Edmond Menya**

**Thesis Submitted to the Faculty of Information Technology in partial fulfillment of the requirements for the award of a degree in Masters of Science in Information Technology of Strathmore University, Nairobi Kenya.**

**Date of Submission: May 2018**

## Declaration

I declare that this work has never been submitted for examination in any university. To the best of my knowledge, this thesis contains no research work previously published with exclusion of the referenced work included.

**Signature:** …………………………………… **Date:** …………………………………

**Name:** Edmond Menya

**Registration Number:** 73516

## Approval

I certify that this work is being submitted for examination with my approval

**Signature:** ………………………………………… **Date:** ………………………………

Prof. Ismail Ateya Lukandu, D. Sc.

Faculty of Information Technology

Strathmore University

# Abstract

Learning is an integral aspect to the development of students as well as progressing of a society. The process is always marked with milestones from class work to semester projects and eventually examinations. Students are always required, as a standard, to sit for an instructor set exam paper. The grade and scores that the student garners indicator of progress, amount of knowledge acquired as well as whether or not the student is qualified for the next academic level. Exams are thus an imperative aspect in the academic life cycle and a critical one for that matter. However, the examinations marking and grading process has been marred with inefficiencies, irregularities and unethical practices over the years. This study aimed at achieving the automation of the exam marking process. This approach seeks to introduce efficiencies cutting down time and cost involved in examinations marking in addition to eliminating human bias in the marking process. Research objectives were centered around studying accuracy levels of past exam papers marked by human instructors, reviewing challenges linked to the examination marking process, reviewing existing models, frameworks, architectures and algorithms that have tried exam marking automation, to develop an improved algorithm-based solution that is efficient for the marking problem and performing of experiments to validate the algorithm. The research engaged experimental research experimenting the relation between keywords, synonyms and their related words involvement in artificial marking and marking accuracy. The outcome is an algorithm that mines related words and counts between scheme and student answer to mark exams. The findings were that the model achieves an improved marking accuracy by a margin of 16% from 73% to 89%. The model achieved more accuracy when grading lower mark answers achieving 99.9% when marking 1-mark answers.

# Table of Contents

# List of Figures

# List of Equations

# List of Tables

# Abbreviations and Acronyms

| | | |
|---|---|---|
| **ADT** | - | Abstract Data Type |
| **AES** | - | Automated Essay Scoring |
| **BBIT** | - | Bachelor of Business Information Technology |
| **CCT** | - | Classical Test Theory |
| **CUE** | - | Commission for University Education |
| **GCSE** | - | General Certificate of Secondary Education |
| **IRT** | - | Item Response Theory |
| **KAT** | - | Knowledge Analysis Technologies |
| **KBS** | - | Kenya Bureau of Statistics |
| **KNEC** | - | Kenya National Examinations Council |
| **LSA** | - | Latent Semantic Analysis |
| **ML** | - | Machine Learning |
| **MoEST** | - | Ministry of Education Science and Technology |
| **MoP** | - | Ministry of Planning |
| **NLP** | - | Natural Language Processing |

# Definition of Terms

**Abstract Data Type** A collection of primitive data units together with operators that manipulate them. ADT is the foundation of Data Structures (Clifford, 2010).

**Algorithm** A mathematical function $f(x)$ that takes data inputs applies a series of steps to manipulate the data and returns a specific output in form of processed information (Clifford, 2010).

**Computational Linguistics** Another name for Natural Language Processing. An A.I branch that creates models that learn patterns from unstructured data (Jurafsky & Martin, 2017).

**Corpus** A body of text under study in computational linguistics (Jurafsky & Martin, 2017).

**Lexicon** A collection of Vocabulary in a dictionary kind of a setting used in NLP (Jurafsky & Martin, 2017).

**Natural Language Processing** A branch of Artificial Intelligence that aims to give machines the ability to process human language (Jurafsky & Martin, 2017).

**Machine Learning** A branch of Artificial Intelligence that studies model algorithms that allow machines to learn a general function from unstructured data (Russell & Norvig, 2010).

**Supervised Machine Learning** Learning achieved through labeled data used to learn the function that maps inputs to outputs. Includes classification (Russell & Norvig, 2010).

**Unsupervised Machine Learning** Unlabeled data set used to build an intelligent model. Includes clustering (Russell & Norvig, 2010).

**Test Data** The data set used to test the accuracy of an intelligent model earlier constructed using training data set (Jurafsky & Martin, 2017).

**Training Data** Data set whose mined characteristics develops an intelligent model (Jurafsky & Martin, 2017).

# CHAPTER ONE: INTRODUCTION

## 1.1. Background to the Study

Perhaps one of the most trending and imperative topic in contemporary society is *Education*. The hang-in and fall-off the headlines of education related topics have been rife. From the floors of parliaments into the streets, inside Lecture halls of institutions of higher learning to both the print and electronic media, this has all unfolded. Education finds a special position in current times given its key role in professions and achieving development goals. In Kenya, for instance, the constitution legalizes access to education as a basic (Kenya Law, 2008). In addition, education is incorporated in the social pillar of the Kenyan Vision 2030 as an aid to the achievement of the mid-income economy that Kenya intends to achieve by 2030 (Ministry of Planning, 2008).

At the heart of any educational system are exams. Examinations have been the standard procedure for years to qualify students to the next academic level or completion of an academic program. Examination questions are set by instructors in form of either multiple choice or short-essay questions. Students are expected to answer questions using their *own words* to bring out the *meaning*, which later during marking, the student-provided answers are compared to the instructor provided marking scheme for correct mark awarding.

Major challenges and inefficiencies have crippled the examination marking process. In 2010, MoEST reported that examiners who mark student's scripts are susceptible to *bias* and *subjectivity*, they consume much *time*, the process is *costly* and *inaccurate* with grades having been based on different ways of interpreting meaning in the student provided answers. Observations have been made of the now common case of two examiners marking the same script to give different scores. Both *Kipatanui* and *Ministry of Education* in 2011 and 2014 respectively, affirms that fraudsters of examination codes have long taken advantage of these loopholes. Given the grave nature of these maladies, a solution has been sought for years to no avail, with the many challenges facing examination administration. As Frost (2008) concludes in his paper, a computer-based solution demonstrates the capability to mark and grade multiple student papers with great levels of *accuracy*.

In other disturbing reports, other than bias marking, cases of exam cheatings have been reported spanning a long period. This has paused a great challenge to the credibility of the education system and examinational integrity. In the Kenyan case, in a detailed official letter sent to media houses, Education cabinet secretary stated that in 2015 alone over **15,000** cases of exam malpractices were reported and the 2016 University audit, reveals a major rise in malfunctions in how examinations and grading are carried out in Universities (Commission for University Education, 2017). This calls for urgent solutions given the delicate nature of academia in measuring progress.

The aim of this study is to review existing and related algorithms, models and frameworks of Artificial Intelligence and computational linguistics with an aim of realizing a competent algorithm to automate the marking process. The solution aims to achieve efficiencies and restore integrity in the manner in which exams are conducted and grades are awarded.

## 1.2. Problem Statement

Exams are the heart and soul of any education system. The process of marking exams has been mired with inefficiencies impeding the process (Commission for University Education, 2017). Marking and grading of exams done by human instructors has been susceptible to *bias* and *inaccuracy* (Ministry of Education Science & Technology, 2014). In addition, report by Kiptanui et al. (2011) has shown that the process is *time consuming* and *economically taxing*.

The situation, as it stands, is in sheer jeopardy. Cooperate organizations and the employment community have lost their trust in the current education system given that students possess grades which is no true reflection of academic progress (George, 2011). Poor examinations administration and marking is to blame for this crippling of the grade-awarding process. Various tested checks and balances have been to no avail with some even widening the problem. In 2015 and successive years, MoEST has admitted that an urgent solution is needed and current explored solutions are non-efficient.

## 1.3. Research Objectives

i. To study accuracy levels of past exam papers marked by human instructors

ii. To review challenges of the examination marking process

iii. To review existing models, frameworks, architectures and algorithms that can automate the exam marking process

iv. To develop an improved algorithm-based solution that solves the problem

v. To perform experiments testing validity of the developed algorithm

## 1.4. Research Questions

i. How have human markers and graders been carrying out the marking process?

ii. What current algorithms, frameworks and models exist that simulates the marking process and related techniques?

iii. How can an *extended*-algorithm that counters the marking problem be developed?

iv. In what ways can an experiment be performed to validate the developed Algorithm?

## 1.5. Scope of research

This research focuses on proposing an algorithm that automates marking and grading of short essay answers written in English and presented electronically. The developed algorithm was not tested on analysis of text in other languages considered official and academic, spoken across differing cultures and nationalities. In addition, the handling and conversion of hard-copy exam papers into digital text and images are not studied as part of this research. Data collected was on Information technology test done by Kenyan University students.

## 1.6. Justification of study

Inefficiencies in the examinations marking and grading process has been a thorn in the academic flesh for years. Education (2014), strived to seek immediate and lasting solutions to the endless problems admitted also by Ministry of Education Science & Technology (2015). Marking and grading of large amounts exam papers within the shortest time possible while achieving high levels of accuracy is a highly desired achievement. This can exponentially cut down cost and save teacher time spent in marking assignments and exams allowing them to focus more on research and quality teaching.

A rogue and deprived education system leads to paralysis in every sector in the economy. The rife cases of cooperate institutions and employers complaining about the quality of graduates released into the market is wanting. George (2011), affirms that the amount of finances spent to train such graduates is costing companies millions and the problem can be traced back to non-compliance to examination standards set by CUE. Exam marking automation can re-introduce back trust and integrity in grading of students.

# CHAPTER: TWO: LITERATURE REVIEW

## 2.1. Introduction

A plethora of researchers and academicians have formulated and attempted to better the examinations process by proposing technical solutions. The basal agenda of this chapter is to present such studied works. It commences with existing formal theories followed by challenges encountered in the exam-marking environment. A discussion on various technical ways pre-tried in solving the challenges follows. These are mainly computational linguistics and machine learning approaches in conjunction with their existent algorithms, models and frameworks. Also highlighted are their shortcomings.

## 2.2. Contemporary Formal theories of examinations assessments

Traditional test theory and Item Response theories are the two classical mathematical theories that forms the basis for the modern-day approach to examinations assessments. Alastair and Gill (2004), describe the distinction between the two theories normally considered synonymous.

### 2.2.1. Classical Test Theory (CTT)

This theory is based on assessing students with regards to their scores. Pointed out by Alastair et al. (2004), the theory is widely used today for examination marking, it holds the following classical equation:

$$X_o = X_t + e$$

**Equation 2. 1: Classical Test Theory**

**Where:**

$\boldsymbol{X_o}$ = Observation made by marker

$\boldsymbol{X_t}$ = True observation based on marking scheme

$\boldsymbol{e}$ = Error committed during observation

The ***X's*** can either be item or test scores with the test scores being a summation of all item scores. The equation points out that in the process of marking, the human examiner is prone to committing error, which varies given various circumstances such as *bias* and *mood* of the examiner.

### 2.2.2. Modern test theory

Modern Test Theory is an improvement on the traditional test theory. Collectively, all the modern test theories are studied under **Latent Trait Theories** from where the *Item Response* and *Rasch Models* are distinguished.

#### *2.2.2.1. Item Response Theory (IRT)*

IRT model has its origin in psychometrics and is concerned with the building of models for testing certain abilities. Xinming and Yung (2014), states that IRT is concerned not only with accurate test scoring but also with development of test items. Test items are set dependent on the aim of the examiner whether they seek to test abilities, traits or behavior in the examinee. The most widely used version of IRT is the Rasch Model.

#### *2.2.2.2. Rasch Models*

Rasch Model is a binary model classifying student's responses as either correct (1) or incorrect(0), which is also identified as a Dichotomous Response Model (Linden, 2010). The model's mathematical definition measures the Probability of a Correct Response using the equation:

$$Pr_{(x_{ij}\,=\,1)} = \frac{e^{n_i - \alpha_j}}{1 + e^{n_i - \alpha_j}}$$

**Equation 2. 2: Probability of Correct Response**

With the assumptions that we have $j$ binary items ranging, $X_i \dots\dots..X_j$, and **1** is the binary value for correct, **0** for incorrect response.

$n_i$ is the ability of subject $i$

$\alpha_j$ is the difficulty parameter of item *j*

This equation signifies that the probability of a correct response is determined by the *items difficulty* and the *subjects' ability* (Xinming & Yiu, 2014). The ability is directly proportional to the probability of correct response. The item characteristic curve below illustrates this relationship:

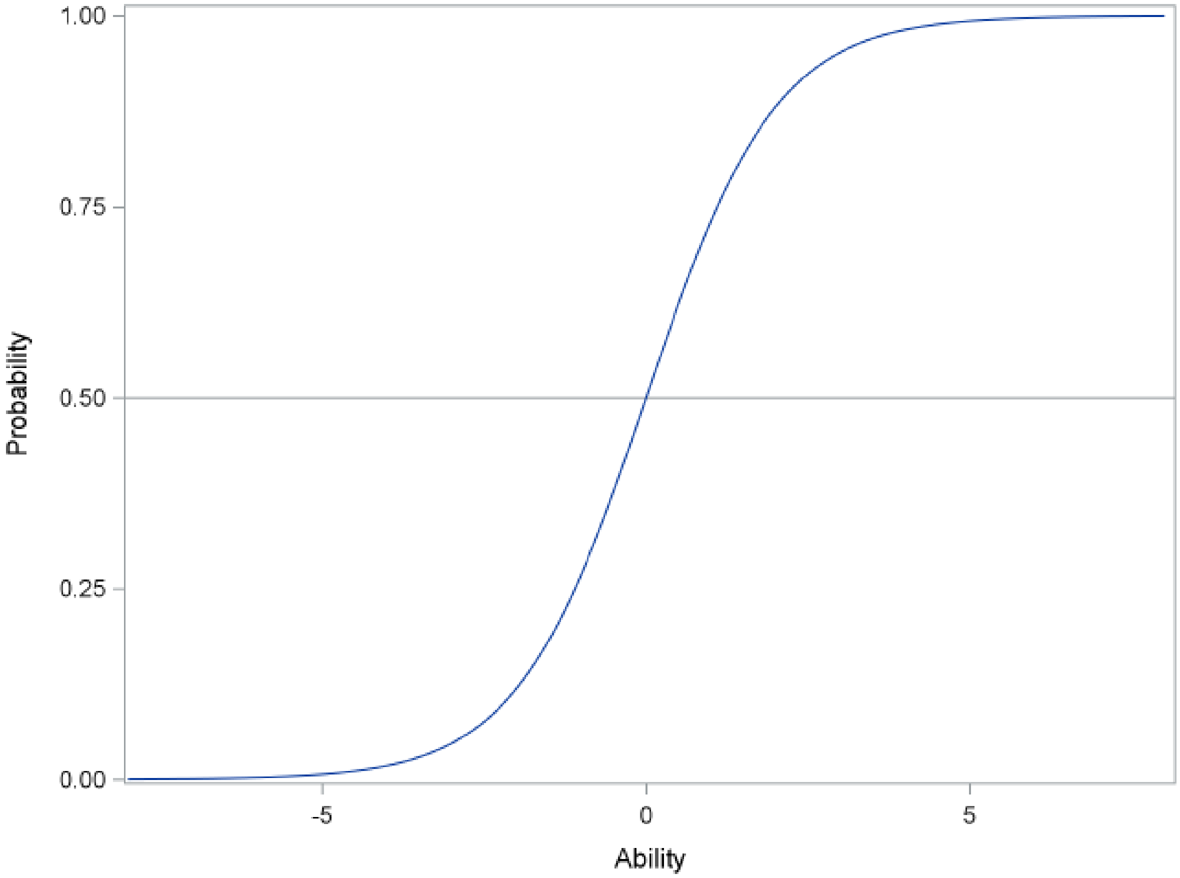


**Figure 2. 1: Item Characteristic Curve (Xinming & Yiu, 2014)**

The curve illustrates an exponential increase in probability with an increase in ability meaning that as the subjects ability increases also the probability of a correct response increases in practice.

Xinming and Yiu (2014) further point out a rather unique case when the subject's ability is equal to the value of difficulty parameter, which measures the level of difficulty in answering and item correctly. This case gives a probability of correct response as $\mathbf{0.5}$ which in practice is the measure for average respondents.

In one profound example for the average case, Xinming and Yiu (2014), maps three Item Characteristic Curves (ICC) with difficulty parameters of $-2, 0\ and\ 2$ respectively illustrating that the location of the ICC is determined by the value of the difficulty parameter.

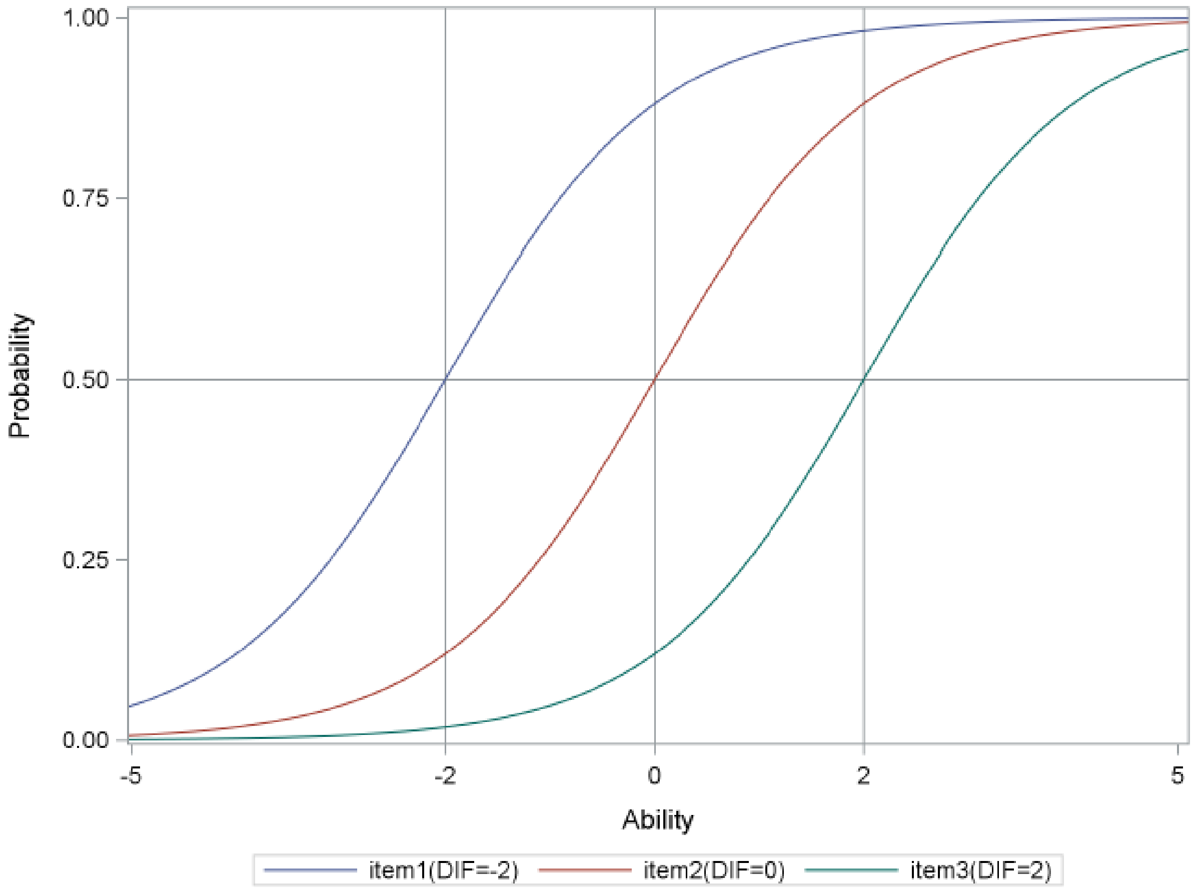


**Figure 2. 2: Item Characteristic Curve with Difficulty parameter (Xinming & Yiu, 2014)**

Respondents with abilities $-2, 0\ and\ 2$ have a $0.5$ probability of correct response.

### 2.2.3. Classical Test theory vs. Item Response Theory

The major drawback of Classical Test theory is the lack of distinction between item or test difficulty and the respondent's ability. CTT is based on observations, measuring which ones are true and those which are non-true. However, in real setting, respondents possess varying abilities in answering tests given their level of academia, information, exposure etcetera. Item Response Theory, improves on the Traditional but including parameters for measuring ability of subject and difficulty of answering a question by determining the probability of correct response (Xinming & Yiu, 2014).

### 2.2.4. Type of questions and related type of answers in the examinations environment

#### *2.2.4.1. Question verb keywords*

Academia is a wide field with many subjects. Differing subjects administer their exams with different question formulation see Appendices B and C. However, there are common keywords in the examinations environment as listed in Table 2.1 below. Each keyword dictates the type of expected answer. It is important to note that the keywords are mostly *verbs;* a part of speech whose significance is stressed more in section 2.6.3 a. knowledge of the question keyword can help the prediction of the depth of considerable synonyms suggested by the algorithm in this study presented in section 2.6.3.

**Table 2. 1: Key Verbs found in Exam questions**

| KEYWORD | EXPECTED ANSWER |
|---|---|
| | |
| **ANALYSE** | Break an issue down into its component parts; discuss them and show how they interrelate. |
| | |
| **ARGUE** | Make a case, based on appropriate evidence and logically structured, for and/or against some given point of view. |
| | |
| **ASSESS** | Estimate the value or importance of something, paying attention to positive and/or negative aspects. |
| | |
| **COMPARE** | Look for similarities and differences between. |
| | |
| **CONTRAST** | Set in opposition in order to bring out differences. |
| | |
| **CRITICISE** | Give your judgment about the merit of theories or opinions or about the truth of facts, and back your judgment by a discussion of the evidence. |
| | |
| **DEFINE** | Set down the precise meaning of the word or phrase, giving sufficient detail so as to distinguish it. |
| | |
| **DESCRIBE** | Give a detailed or graphic account. |

| | |
|---|---|
| **DISCUSS** | Investigate or examine by argument; sift and debate giving reasons for and against. |
| | |
| **EXPLAIN** | Tell how things work or how they came to be the way they are. |
| | |
| **IDENTIFY** | Pick out what you regard as the key features of something, perhaps making clear the criteria you use in doing so. |
| | |
| **ILLUSTRATE** | Use a figure or diagram to explain or clarify, or make it clear by the use of concrete examples. |
| | |
| **JUSTIFY** | Express valid reasons for accepting a particular interpretation or conclusion. |
| | |
| **OUTLINE** | Indicate the main features of a topic or sequence of events, possibly setting them within a clear structure or framework to show how they interrelate. |
| | |
| **PROVE** | Demonstrate or establish the truth or accuracy, giving evidence or a logical sequence of statements from evidence to conclusion. |
| | |
| **RELATE** | Explain how things are connected to each other and to what extent they are alike or affect each other. |
| | |
| **REVIEW** | To make a survey of, examining the subject critically. |
| | |
| **STATE** | Present in brief, clear form the main points. |
| | |
| **SUMMARISE** | Give a concise account of the chief points or substance of the matter, omitting details and examples. |
| | |
| **TRACE** | Follow the development or history of a topic form some point of origin. |

#### *2.2.4.2. Verb keywords, required answers and expected marks*

The type of answer required from a student is dictated by the question keyword and this in turn determines the amount of marks to be awarded. Some keywords require mostly one word, or one sentence answer, others require a paragraph or mini-essay. Other keywords require an answer in tabular form, while other questions require diagrammatic answers. Below, Table 2.2 summarizes the above showing proposed depth of synonyms to be considered as part of correct answer. (More description at section 2.6.3 c.)

**Table 2. 2: keyword-answer relationship**

| Question Keywords | Expected answer and marks | *Depth n* |
|---|---|---|
| **State, Name, List, Identify, Outline, Mention, Define** | Short answer ranging between one word and one sentence. Mostly one mark per answer | $n_1, n_2$ |
| **Explain, Discuss, Describe, Justify** | One paragraph answer with a number of statements that are interrelated. May contain examples. Mostly more than two marks the marks being dependent on the required statements | $n_2, n_x$ |
| **Compare and Contrast, Differentiate and Distinguish** | Tabulated answers with one side stating a proposition and the other offering a counter proposition. For a mark to be awarded, both proposition and its counter must be true. Mostly binary mark for each answer | $n_1, n_x$ |
| **Demonstrate, Show** | Answers required in diagram form or even equations. Mark awarded usually higher than the above per unit question | $n_1$ |

#### *2.2.4.3. Special subjects Questions and Answers*

Subjects such as mathematics, sciences, some arts and foreign languages (non-English) require special answers to their questions (shown in Appendix B). Most mathematical and scientific questions are answered using *equations*, *notations*, *symbols* and *diagrams*. In addition, these answers are normally either correct or not (non-fuzzy answers) given the standardized nature of these fields, a mathematical formula is always a standard with no variations. Such answers rarely require consideration of synonyms as part of answer thus a proposed depth of $n_1$. Special keywords for mathematical questions are summarized below:

**Table 2. 3: Mathematical question key verbs**

| **Some common Mathematical Question key verbs** |
|---|
| **Evaluate** |
| **Expand** |
| **Find the value of** |
| **Calculate** |
| **Solve** |
| **Simplify** |
| **Determine** |
| **Construct** |
| **Express** |

#### *2.2.4.4. Compound keywords and expected Answers*

In most exam questions, keywords exist in compounded form. This changes the required answer format, the expected marks and the depth of $n$ of synonyms to be considered. Some compounded form examples are *state and explain, list and discuss, explain with examples.*

#### *2.2.4.5. Type of answers*

Depended on question keyword and expected marks to be awarded, we have different type of answers. Different answers are displayed in differing formats and marked in differing ways shown in Table 2.4.

**Table 2. 4: Answer classes description**

| Answer Type | Description |
|---|---|
| **Statement answers** | One word/statement answers. They are the easiest to grade with the list depth of $n$. one sentence answers have keywords hidden in language creativity requiring a higher depth of $n$ |
| **Symbolic answers** | Mostly scientific and mathematical answers. Require formulae, notations, chemical equations and symbols. Mostly depth $n_1$ |
| **Tabulated answers** | Answered using tables with both side rationally connected |
| **Diagrammatic answers** | Mostly demonstrations in diagrams. Academic diagrams are mostly not artistic in nature but require mere shapes, connectors, tables, nodes and graphs. Mostly they carry a depth of $n_1$ or $n_2$ since they most either match the scheme diagram or closely related |
| **Foreign Language answers** | Answers written in other natural languages apart from English. Some alphabets require special symbols many of which have UTF encodings |

## 2.3. Challenges & Inefficiencies in Manual Exam Marking and Grading

Marking and grading of examinations is a timely, costly and a quite engaging process. Examiners, instructors take months to mark, tally and present national examination results. After result release, students dissatisfied by their results are allowed a chance to petition; numbers have been growing. Such challenges render the process unreliable and at times not trusted in its entirety (Gari, 2017).

### 2.3.1. Total tally challenge

Grading a student paper is non-immune from bias and even after marking each answer, summing up the total seems to be a challenge mostly to an exhausted examiner. Cases have been rife on the total tally owed to every candidate paper. In some cases, it is an under-count leading to the candidate gaining a lower score than the legit one, in other cases it is an over count the candidate gathering a higher score than deserved as reported by Ratcliffe (2014). A peculiar case is reported where a non-prepared student who scored an **A-grade** in English and after a remark on the paper it was noted that it hardly deserved an **E-grade** (the lowest grade). Such cases have led to the rise in number of rejection of exam results and requested remarking of test papers Gari (2017), in addition to a large number of resists (Ratcliffe, 2014).

An article published in the Guardian Vasagar (2012), reports of such cases in the national GCSE and A-Levels examinations where examiners were suspended due to tally mistakes. Such errors, the report concludes, are serious since they affect University Placement and course choice for students (Ratcliffe, 2014). The reported marking process was characterized by uncertainty resulting in termination of contract of **78** examiners with over **200** cases of marking irregularities observed.

### 2.3.2. Time challenge

The time it takes to mark, grade and present final results is a consuming inefficient process. Majorly, examiners are given a standard timeline to mark, grade and submit final results. It is of major concern that this time is given with a sole consideration to *equality* rather than *equity*; instructors with fairly high number of scripts to mark are assigned the same amount of time with those ones with a relatively lower number of scripts. In other cases, reports Ratcliffe (2014), an examiner was asked to mark hundreds of extra papers past the deadline.

The time challenge is correlational to the other challenges. For instance, since an instructor is rushing against time, probability of bias and under-tallying is higher. In addition, since instructors are normally paid as per the number of scripts they mark they tend to set individual goals to aim at marking more at the expense of accuracy (Kenya National Examinations Council, 2016).

### 2.3.3. Cost challenge

Hiring human examiners for the marking and grading process is relatively costly. This is due to the fact that quite a number of examiners need to be involved in the process; *supervisors*, *markers*, *observers* and *statisticians* are commonly involved to track the process. An estimate number of **17, 000** personnel were involved in the marking process of Kenyan Secondary level exams last year (Kenya National Examinations Council, 2016).

The cost of securing the marking center is also significant and increases in relation to the number of existent marking centers, for the Kenyan national 2016 examination, **33** in total. A marking center is expected to have in place a number of measures aimed at securing the integrity of the marking process. Challenges of barring unwanted personnel and performing background checks on human examiners has proven difficult and costly.

The marking of Kenyan national exams was budgeted at an estimated **sh. 1.7 Billon** (USD 17 Million) in 2016 (Budget Highlights 2016/2017, 2016). On the candidate's side, the cost of remarking has also been estimated to be too high for most to afford or consider (Ratcliffe, 2014).

### 2.3.4. Challenge in Detecting cases of cheating

Cases of cheating in exams have been rampant and increasing. In Kenya exam teaching rose by **60**% in 2015 and **70**% in 2016 with results of about **5000** candidates being nullified (Chemweno, 2016). Such cases have been attributed to the widespread of leakage that is tied to corruption in the country, a condition proving difficult to track. However, cases of dabbing, though not the leading cheating challenge, have been observed. Dabbing is common in continuous assessment test (CATs) and semester assignments, which are at times not regarded as serious as final examination yet they have a significant weight in the final semester grade in the University grading standards.

When two different scripts written by two candidates with dabbing evidence is marked by different examiners, detection of dabbing becomes a challenge. The use of electronic marking can help detect dabbing cases among students.

## 2.4. Exam Scoring Using Computational Linguistics

### 2.4.1. Natural Language Processing (NLP)

Natural language refers to any language that is spoken by human persons in a spontaneous way. NLP, also styled computational linguistics, is thus an artificial intelligence subdomain that studies linguistics rules to enable computer to have capacity to process natural languages relayed either in speech or written data formats.

Linguistics as a science of languages, studies *phonology* (sounds), *morphology* (word formation), *syntax* (sentence structure), *semantics* (meaning) and *pragmatics* (understanding) (Abhimanyu et al., 2013). In linguistics, there exists two phases of analysis: ***high level*** which corresponds to speech analysis and ***low level*** that corresponds to natural language processing. Speech recognition consists of five levels namely; *acoustic signals, morphemes, phones, words* and *letter strings*. Jurafsky and Martin in 2017, argued that NLP is divided into two major camps; the classical approach which majorly focuses on *morphemes, words, meaning in context, meaning out of context, phrases* and *sentences*, and the statistical approach that relies on statistical machine learning approach and algorithms.

Jurafsky and Martin (2017), summaries classical NLP in five key steps: *morphological and lexical analysis, syntactic analysis, semantic analysis, discourse analysis* and *pragmatic analysis*. Earlier in 2011, Kumar held the same idea giving more details on each aspect.

Morphology analyses the structure of words, how they are formed according to their components which are referred to as ***morphemes***. Lexical analysis entails looking up a word and all its related meanings, synonyms and word types (parts of speech) from a provided knowledge base. Syntax studies the relationships that exists between words, phrases and sentences, also grammar rules are analyzed. Semantics analyses meaning that is extracted from sentence syntax thus knowledge representation occurs at this stage. Pragmatics and discourse analysis seeks to extract intent from a sentence based on the context in which words are said.

### 2.4.2. Automated Essay Scoring (AES)

Automated Essay Scoring (AES), is the application of computer technology in evaluating and scoring written prose in academia (Dikli, 2006). AES has been an actively researched NLP area inspired by its capability to save on time and cost as well as its ability to give feedback to students after grading, a task that has proven rather difficult and time consuming for instructors and examiners. Mahwah et al., (2010), pointed out that the main advantage of AES is that such systems can provide a student with a score as well as feedback within seconds. Automated Essay scoring is a money and time saver and reduces the teachers' paper load, he concluded.

Research on Automating Essay Scoring is as old as Natural Language Processing. There are four main success applications in this area of study: Project Essay Grader (PEG), Intelligent Essay Assessor (IEA), E-Rater and IntelliMetric (Dikli, 2006).

### 2.4.3. Project Essay Grader (PEG)

This is the classical first successful Automated Essay Scoring project. Developed by Ellis Page in the mid 60's, the system automates the scoring processes using machine learning approach. Sample of essays are used in training the system, proxy variables are selected and entered in a prediction equation. Scores are assigned by computing beta weights from the training stage (Dikli, 2006).

The system major drawback was its concentration on the surface structures of the essay Burstein et al. (1998) ignoring imperative semantic aspects such as the elements of art and creativity that are very common in writing.

### 2.4.4. Intelligent Essay Assessor (IEA)

Created by psychologist *Thomas Landauer, Peter Foltz,* and *Darrell Laham,* IEA uses Latent Semantic Analysis (LSA) which is a modern successful computer linguistic technology. The approach enables the system to analyze and score essays using semantics analysis methodology. IEA runs on Pearson's Automated Knowledge Analysis Technologies (PKT) (Pearson, 2017).

#### *2.4.4.1. Latent Semantic Analysis (LSA)*

Latent Semantics Analysis, also called Latent Semantic Indexing (LSI) by Mahwah et al., (2010), refers to a statistical model of word usage that permits comparison of the semantic similarities between pieces of textual information defined by Dikli (2006). LSA draws its success based on the fact that the meaning of an entire passage is dependent on its constituent words.

$$Passage\ Meaning\ =\ meaning\ of\ word1 +\ meaning\ of\ word\ 2\ +\ \ldots\ldots\ldots\ldots + meaning\ of\ word\ n$$

**Equation 2. 3: Semantic Essay Equation (Mahwah et al., 2010)**

LSA represents words, sentences and paragraph in a high dimensionality matrix known as “semantic space”. Mahwah et al. (2010) demonstrated LSA’s capability to emulate human cognitive abilities inclusive of developmental acquisition of recognition vocabulary to word-categorization, sentence-word semantic priming, discourse comprehension, and judgments of essay quality a work began by Launder et al., in 1998.

#### *2.4.4.2.LSA Steps*

i. **Step 1:** Representation of text as a matrix, unique words forming the rows while text passage and context forms the columns. Each individual cell records the frequency of which each word appears in the text.

ii. **Step 2:** Preliminary transformation is performed on each cell. Each cell frequency is weighted by a function that expresses both the word’s importance in the particular passage and the degree to which the word type carries information in the domain of discourse in general (Launder et al., 1998).

iii. **Step 3:** Singular Value Decomposition (SVD) is performed to the matrix constructed in (b) above. Jurafsky and Martin (2007) define SVD as *a method for finding the most important dimensions of a data set, those dimensions along which the data varies the most*. This method was first applied by Deerwester (1998) to the task of generating embeddings from term document matrices.

    The process involves decomposition of the rectangular matrix into the product of three other matrices. One component matrix describes the original row entities as vectors of derived orthogonal factor values, another describes the original column entities in the same way, and the third is a diagonal matrix containing scaling values such that when the three components are matrix-multiplied, the original matrix is reconstructed.

### 2.4.5. IBM 805 Test Scoring Machine

The IBM 805 Machine is a classical Examinations marker and grader designed by Reynold Johnson and Benjamin Woods. As captured by International Business Machines (2017) the test scoring machine was designed to simplify the laborious task of grading student assessments by both Johnson and Benjamin who were teachers by profession. The first model graded students by detecting pencil marks using electrical conductivity property of graphite. Students were required to mark correct answers from provided multiple-choice list then their various selections would be analyzed by the machine and graded.

The major drawback of the system was that the level of electricity conducted varied greatly with the intensity of the pencil mark thus inaccuracies in grading were rampant especially in the case where a student applied a fainter pencil mark. Later research replaced pencil mark- method with high-resistant units raising resistance to the point where pencil marks no longer mattered. Student answers were treated as a classification problem of two classes: “right” and “wrong” (International Business Machines, 2017).

### 2.4.6. Write to Learn: PEARSON Knowledge Analysis Technologies (KAT)

Write to Learn is a Pearson Education Intelligence Project. Running as a web based tool for summary, essay writing and automated assessment services. Write to Learn measures student assessments using Knowledge Analysis Technologies developed by Pearson Write to Learn (2017). KAT engine is a unique automated assessment technology that evaluates the meaning of text, not just grammatical correctness or spelling.

Write to Learn is currently the *worlds most advanced scoring engine*, it uses Latent Semantic Analysis (LSA) and Pearson’s Intelligent Essay Assessor (PIEA) rankings done by (Automated Scoring, 2017).

## 2.5. Exam Scoring Using Machine Learning and Semantic Parsing

Machine Learning is a multidisciplinary field of study that aims to learn a mathematical function $\boldsymbol{f}(\boldsymbol{x})$ that maps inputs $\boldsymbol{x}$ to observed outputs from historical data. Semantic parsing, on the other hand, is an NLP branch that aims at graphical representation of sentence meaning for purposes of information extraction.

Application of statistical Machine Learning (M.L) techniques in automated test scoring treats marking and grading as a classification problem focusing on the supervised learning approach. Two existent classes; *correct* and *incorrect* are assumed, and a statistical model is built to classify student answers into one of the predefined classes. On the contrary, an unsupervised learning approach can be applied using clustering. Two main clusters are chosen and each student answer is clustered into one of the assumed clusters. Both training and test data are necessary for any Machine Learning model construction thus need to be defined in the development of an artificial examiner.

### 2.5.1. Artificial Intelligent Marker

Frost (2008) applies the use of pre-marked exam answers as training data obtained from biology GCSE test attempted by students and graded by instructors between $\mathbf{2001}$ and $\mathbf{2003}$. Frosts' model uses two machine learning algorithms, Decision Trees and Naïve Bayes Classifier to learn from the training data set. Four main stages are followed by the model to achieve the marking process: *Semantic parsing, featurization, data preparation* and *training.*

### 2.5.2. Accuracy of Machine Learning Artificial Examiner

Artificial Examiner achieved through use of machine learning algorithms achieved an average accuracy of $\mathbf{67.07}\%$, tested using *10-fold cross-validation*, using $\mathbf{90}\%$ of data for training and the remaining $\mathbf{10}\%$ for testing, technique shown in Fig 2.3 below, (Frost, 2008). Decision tree algorithm performed worse recording a $\mathbf{5}\%$ down performance thus Naïve Bayes was used as the main classifier. The model developed by Frost (2008), compares three different approaches; keywords approach, purely semantic approach and the machine learning approach to developing and artificial examiner.

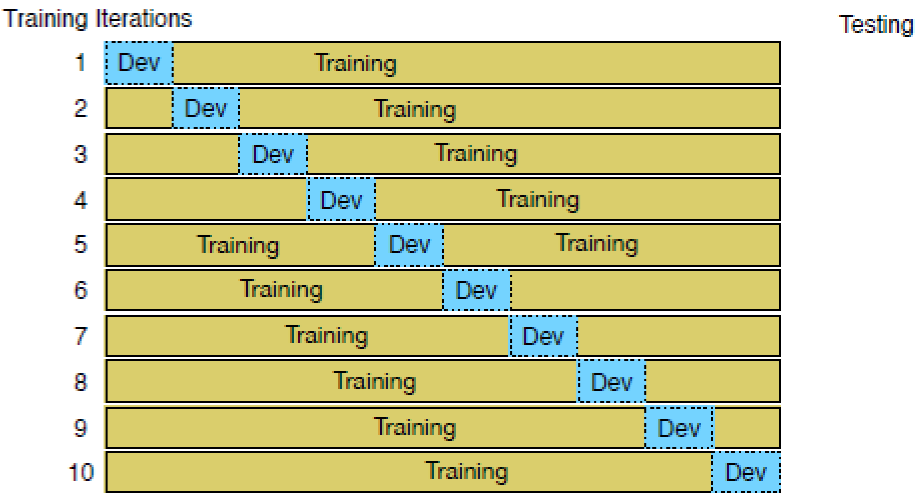


**Figure 2. 3: 10-fold cross validation iteration (Jurafsky & Martin, 2017)**

### 2.5.3. Challenges of Machine Learning Artificial Examiners

As is the challenge with the whole field of machine learning, developing a model for artificial examiner suffers the question *how much data is enough to have a near accurate model*? Russell and Norvig (2010), state that this question has been pondered on for years. Frost (2008) summarizes by starting the challenge to his approach remained the limited amount of training data stating that the model is likely to improve with more data. Russell and Norvig (2010) further prove that statistical Machine Learners improve with additional training data, the more the training data the closer hypothesis $h$ is closer to the true target function $f$. but since the true function $f$ can never be reached by statistical learners, the achieved model is termed *Probably Approximately Correct* (**PAC** Algorithm). The question remains how much pre-marked question does the model require in order to achieve high level accuracies?

In addition, the machine learning approach includes pre-marked exam papers for training data. This pauses a greater challenge in that pre-marked exam papers themselves contain elements of *bias*, grammatical errors and increases the probability of having cases of questions not previously encountered by the classifier algorithm. Due to this, many new questions set by examiners and attempted by students end up not being graded by the artificial examiner. Furthermore, additional processes, are needed by the intelligent marker to correct grammatical errors such as spelling.

Treating test scoring as a classification problem lead to binary marking where a student answer is either right or wrong. This method offers no place for awarding a *half-mark* or related scores where a student has a near answer.

Machine learning models require massive computer resources; both processing power and memory. Storage, data mining and processing the training data set and testing the model is computationally taxing.

## 2.6. Exam Scoring Using Text Mining and Keyword Extraction

One specific aspect of natural language communication is the use of text to pass on message. Text mining is the analysis of unstructured text data to discover patterns and extract meaningful information (Charu & Cheng, 2012). The field is interested in paradigms such as: *Information extraction, text summarization, keyword extraction,* and *opinion mining*.

A number of algorithms have been developed for information retrieval using keyword extraction, the mostly used ones are: *Text-Frequency Inverse-Document Frequency (TF-IDF), TextRank* and *Rapid Automatic Keyword Extraction (RAKE*). Kogan and Jacob (2010) argued that RAKE algorithm performs best given its ability of extracting key phrases rather than keywords and its efficiency in using computational resources economically compared to ***TF-IDF*** and ***TextRank***.

### *2.6.1.* Rapid Automatic Keyword Extraction: The RAKE Algorithm

RAKE algorithm is an unsupervised, domain and language independent methodology of keyword extraction (Kogan & Jacob, 2010).

In 2010, Kogan and Jacob summarized the steps of RAKE algorithm as:

**i.** Remove all stop words from the text. Stop words are considered to have little or no meaning in statistical text analysis. They include words like *the, for, and, are* etc.

**ii.** Create an array of candidate keywords which are set of words separated by stop words.

**iii.** Calculate the frequency of the words in the key phrases. Further calculations lead to keyword scores

**iv.** Find the degree of each word in the key phrase

**v.** For every candidate keyword find total frequency and degree by summation of word scores

**vi.** Degree and frequency gives the score for being a keyword or key phrase

### 2.6.2. Accuracy of Artificial Exam Marker Implemented using keyword extraction

Frost (2008) compared his model with keyword extraction approach on intelligent artificial examiner. The keyword extraction model achieved average accuracy of **73%, 3%** more than the semantic approach which achieved **70%** accuracy, and **6%** more than the machine learning approach which achieved **67.07%**. Evidently, keyword approach proved to perform better compared to the other two models. However, the use of pre-marked questions as the basis for grading introduces some major challenges that leads to the poor performance of the keyword model.

This research proposes the keyword approach and usage of marking scheme supplied by instructors who set exams as the basis for authoritative answers taking into account Frost (2008) recommendation that manually specifying marking scheme is superior to the machine learning approach. In addition, rather than relying on keywords, this study proposes the mining of keyword synonyms, assigning probabilities of replacement to each synonym and using them to *fuzzy-mark* student answers.

### 2.6.3. Challenges of keyword extraction technique

Keyword extraction technique achieve high levels of accuracy in marking, however, challenge remains on the instructor side in that they have to include as many *variations* of likely answers as possible, this may be a time-consuming process. Such an intelligent marker aims at solving problems experienced in marking and grading but not those encountered in the exam setting process.

In addition, ignoring semantics in look out of answers may lead to the awarding of a grade to answers that are not logical in their syntactical structure provided a keyword or phrase is detected. E.g. if the answer required is *the cat chased the mouse* (keywords: Cat, Chase, Mouse) also the different meaning answer; *the mouse chased the cat* shall be awarded full grade by the artificial grader.

## 2.7. Exam Scoring using Keywords Depth Technique

Given the advantages of keyword extraction technique, and seeking to solve existent challenges, this research proposes the application of keywords synonym depth approach. Keyword extraction shall be used as the main technique used by the artificial examiner for answer grading where keyword extraction achieves accuracy of $\mathbf{79.11}\%$ compared to $\mathbf{76.86}\%$ achieved by semantic parsing approach (Frost, 2008).

For prose answers, semantic parsing approach achieved accuracy of $\mathbf{58.69}\%$ against $\mathbf{58.25}\%$ achieved by keyword approach as reported by Frost (2008). The keyword approach percentage is expected to rise with the use of instructors marking scheme as the authoritative source of keyword mining and marking.

### 2.7.1. Advantages of keyword depth technique

Use of keyword and key-phrases and their *variations* allows for grade awarding for students who may not have the correct answer but still have an idea similar to the answer.

Extraction of keywords and phrases is key since students are not necessarily seeking to *cheat* the automated grading system but psychologically in the examinations environment, students strive to give their best of answers. This prompts it *unnecessary* to check for cases where the student only include keywords and phrases devoid of proper grammatical sentences. Frost (2008) points this important point quoting *Prof. Sargur Srihari*.

High accuracy is an added advantage given the use of instructor-supplied marking scheme as opposed to pre-marked questions as the authoritative answer source.

> *Professor Srihari asked human examiners to grade 300 answer booklets. Half Of the graded scripts were then fed into the computer to "teach" it the grading process. The software identified key words and phrases that were repeatedly associated with high grades. If few of these features are present in an exam script it generally receives a low grade ... Professor Stephen Pullman at the University of Oxford has identified another potential pitfall in Professor Srihari's approach. "You can't just look for keywords, because the student might have the right keywords in the wrong configuration, or they might use keyword equivalents.*
>
> ***John Frost***

### 2.7.2. Keyword Variations Using Lexicon Dictionaries

#### *2.7.2.1. WordNet*

The advancement in NLP for English language had a major boost when Princeton University launched the WordNet project. WordNet is a lexical dictionary and database for English words that clusters words that are closer in meaning and normally used in relation to the other, these are called ***synsets*** a similitude of the English **synonyms**. Miller (1995) terms WordNet as a database that *links English nouns, verbs, adjectives, and adverbs to sets of synonyms that are in turn linked through semantic relations that determine word definitions.*

Lexical relations that exists between synonyms are used to construct *synsets* based on specific sense of words rather than word forms and strings. Elsevier (2006) states that each of WordNet's $\mathbf{117,000}$ synsets is linked to other synsets by means of a small number of *conceptual relations* thus forming meaningful relations among words and phrases that are essential for NLP and text-mining. Elsevier (2006), further states that Word forms with several distinct meanings are represented in as many distinct synsets. Thus, each form-meaning pair in WordNet is unique.

Due to its free availability and continual development as a Princeton University project, WordNet is a stable NLP dictionary and can be used by any software that aims at processing text for meaning.

#### *2.7.2.2. Grady Ward's Moby Lexicon Project*

The Moby project pioneered in 1996 and has been of interest to date as far as words and their synonyms are concerned. The project consisted of compiling an extensive thesaurus where words can be search for their synonyms. The project has since been released into the public domain.

The extensive thesaurus also termed; the largest English Thesaurus by Sheffield (2000) contains: $185,000$ entries fully hyphenated, Word lists in five of the world's great languages. $230,000$ entries fully described by part(s) of speech, listed in priority order, $175,000$ entries fully International Phonetic Alphabet coded, The complete unabridged works of Shakespeare, $30,000$ root words, $2.5\,million$ synonyms and related words, and $610,000+$ words and phrases, the largest word list in the world (Sheffield, 2000).

# CHAPTER THREE: RESEARCH DESIGN AND METHODOLOGY

## 3.1. Introduction

This study aimed at developing an algorithm solution to the examinations marking and grading problem. The research questions under study were:

i. How have human markers and graders been carrying out the marking process?
ii. What current algorithms, models exist that simulates the marking process and related techniques?
iii. How can an extended-algorithm that counters the marking problem be developed?
iv. In what ways can an experiment be performed to validate the developed Algorithm?

The above questions were addressed through experimental research design. This chapter presents the nature of data collected, studied and analyzed in the study.

## 3.2. Research Design

Experimental research design was engaged. This research design involves identification of variables that are of interest and are to be tested, manipulated or measured to observe the consequences. The independent variable under scrutiny was: number of keywords and synonyms contained in student answer. Dependent variables on the other hand, was: accuracy of the answer. The null hypothesis was the claim that artificial exam marking is equal to or greater than 90% accurate if the AES algorithm considers *keywords*, *synonyms* and *other-nyms* relationships between marking scheme and student provided answers.

$$H_0\ Accuracy\ \geq 90\%$$

$$H_a\ Accuracy\ < 90\%$$

Empirical steps were carried out to analyze the effectiveness and accuracy of the algorithm in the marking and grading process. This was run on the sample data that served as control groups consisting of various questions from differing subjects done by a wide variety of students and set by distinctive examiners.

## 3.3. Data Collection

This study was facilitated by the collection of both qualitative and qualitative data. The method to be used in data collection were observation and document studies. The study process involved the collection of secondary data readily available. This was due to the fact that exam grading is a highly sensitive process and cannot be delegated to an algorithm under study in the real examination environment. Pre-marked test papers, pre-set exams and pre-attempted examination papers formed the source of secondary data.

### 3.3.1. Document Studies

Document studies is the method that aided in scrutiny of collected test papers and marking schemes. Studying how marks are distributed and awarded by examiners as well as differing ways in which students answered various questions.

### 3.3.2. Sampling

Due to the many number of academic subjects written in English language, and the numerous number of pre-attempted and pre-graded papers, this study sampled the population under study. Population refers to the whole domain set of elements of interest in a study. For instance, in this study the population is all possible exam papers written in English and their marking schemes.

Random sampling technique was employed by this research. This method ensures the equal chance of every element in the population to be included in the sample and obeys the law of statistical regularity, which states that *if the sample chosen is random then the sample will have the same characteristics as the population*.

Sample size was determined using the re-arranged standard $\boldsymbol{z - score}$ table formula (standard normal table) with assumed $\mathbf{90}\%$ confidence level and $\mp 1$ margin of error. This formula was used with the assumption that data is uniformly distributed.

$$n = \left[\frac{Z\alpha/_2\ \sigma}{W}\right]^2$$

**Equation 3. 1: z-score sample size**

**Where**

$Z\alpha/_2 =$ the critical value, the positive Z value that is at the vertical boundary for the area of $^{\alpha}/_2$ in the right tail of the standard normal distribution.

$\sigma =$ population standard deviation.

$n =$ sample size

$W =$ amount of error allowable on interval estimate

A sample size of 151 $answers$ was arrived at with a 90% confidence interval with $\sigma = 10$ and mean mark allowable range of $\bar{x} \mp 1$ ∴

$$n = \left(\frac{1.645 * 7.47}{1}\right)^2$$

$$= \mathbf{150.9986} \approx \mathbf{151}\ \boldsymbol{answers}$$

## 3.4. Data Analysis

This study used EXCEL software as tool for data analytics for quantitative data. The collected secondary data would amount to a number of calculations that Excel software can perform analysis of the desired parameters within the stipulated time frame with accuracy achieved.

## 3.5. Description of Algorithm to be studied and developed

This study was interested in the study and development of an algorithm to automate the marking process. The key areas of interest in the algorithm model were: The main AES marking engine, an exams database, result manipulation and retrieval engine. Data was collected with the aim of achieving such a system.

## 3.6. Research Quality Assurance and Validity

Validity of data collected was carried out through pre-test exercises. Instructors and examiners were involved to validate the algorithms marking process. Active as well as non-active teaching personnel, instructors and examiners formed a panel and scrutinized the algorithm steps and the marking technique suggested by the study.

## 3.7. Ethical Considerations

Ethics studies the morality of free human actions regarding how good or evil they are (Hamilton, 2012). Moral human acts lead to attainment of the good while immoral acts lead to the encounter of evil. Ethical issues were identified in the collection of secondary data. An ethical dilemma ensued as data collected displayed grades of students that are normally deemed private. Privacy, data ownership were the main ethical issues. The dilemma was addressed using the best utility ethical framework that provides the following framework (Hamilton, 2012):

i. **Identification and description of the situation at hand:** the identified situation all surrounded the examination administering, marking and grading processes

ii. **Definition of the ethical dilemma and key values in question:** issues of privacy, data ownership and user consent were in question

iii. **Stakeholders identification:** the stakeholders were the exam candidates, examiners and institution or exam body.

iv. **Identification of alternative actions and their benefits:** actions to be taken were listed down regarding the above

v. **Listing consequences of each alternative action to be taken:** each repercussion of alternative action taken were identified.

It was resolved to collect data from examiners with student consent.

# CHAPTER FOUR: ARTIFICIAL EXAM SCORER MODELLING

## 4.1. Introduction

The aim of this project was to develop an algorithm that introduces efficiencies into the exam marking of short essay answers. This chapter presents the techniques applied in the modelling of the automated exam scorer. Each attribute in the examinations environment deemed relevant to this project is modelled and presented.

## 4.2. Artificial Examiner Design

The artificial examiner design was achieved through unified modelling language (UML) diagrams and Object-Oriented design. The design diagrams included use case and domain model for demonstrating the domain of interest, entity relationship diagram to model relational database that was used, data flow diagrams to show flow of data in the system, interaction diagram achieved through a sequence diagram.

The main system flow is captured in the below fully dressed use case narrative and then by use case diagram 4.1:

**Actors: Examiner, Student**

| **Main Success Scenarios** | **System Responsibilities** |
|---|---|
| **1.** Examiner sets a new exam | **2.** Create new exam identifier |
| **3.** Examiner writes questions | |
| **4.** Examiner write marking scheme | **5.** Read marking scheme |
| **6.** Student attempts and exam | |
| **7.** Student provides answers to questions | **8.** Read student answer |
| | **9.** Mark student answer |
| **10.** Student accesses result | **11.** Presents results analytics |
| **12.** Examiner accesses results | |

### 4.2.1. Use case

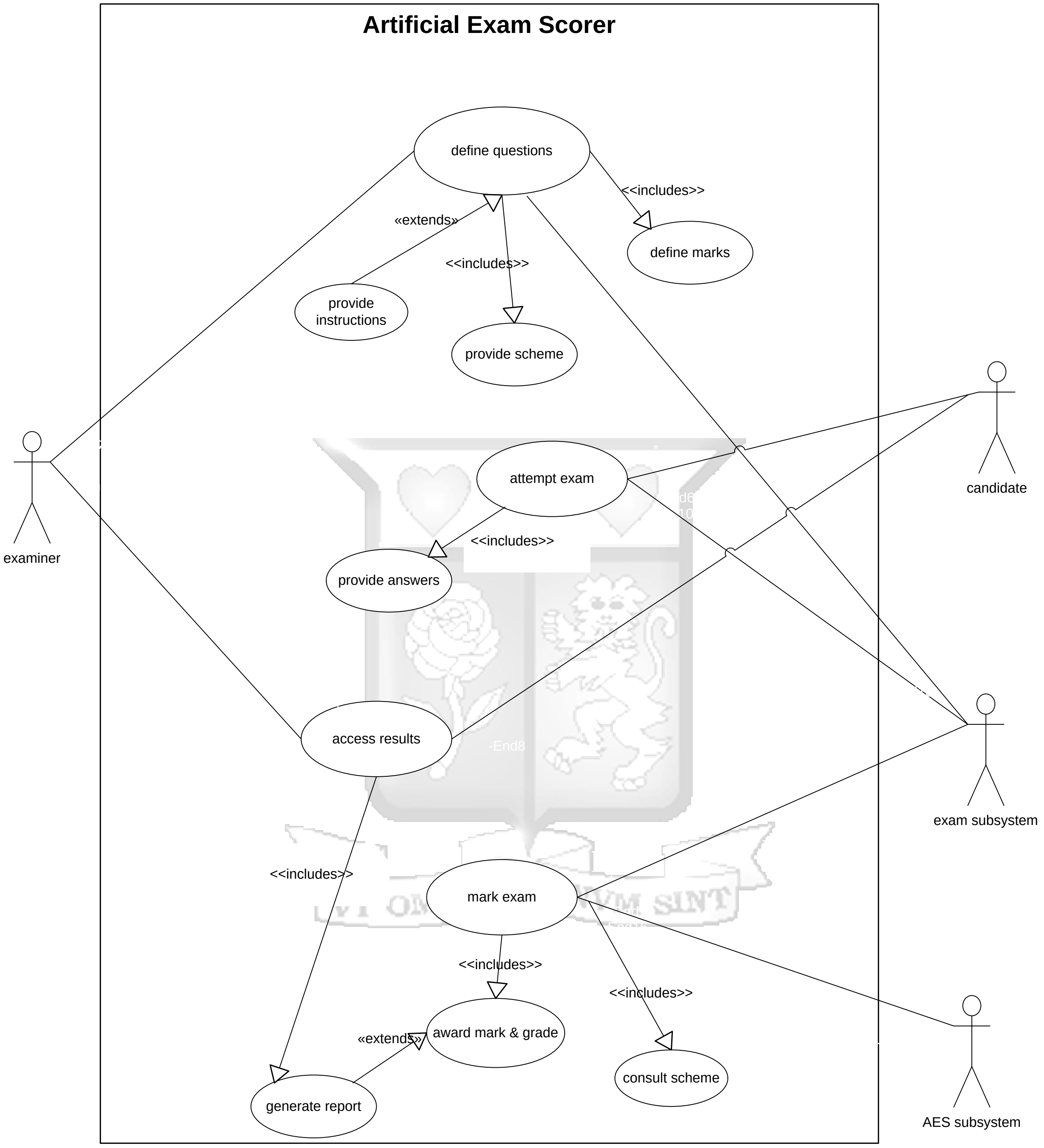


**Figure 4. 1: Artificial Examiner Use Case Diagram**

### 4.2.2. Domain model

A domain model UML diagram is used to model a given domain of interest on which a system is expected to function. This was used to model the domain of interest for the AES system. A total of 12 namely *examiner, candidate, marking scheme, exam, mark engine, total mark, student answer, result view, student grader, exam question, AES system* and *exam description* entities were identified and modeled as shown in figure 4.2 below:

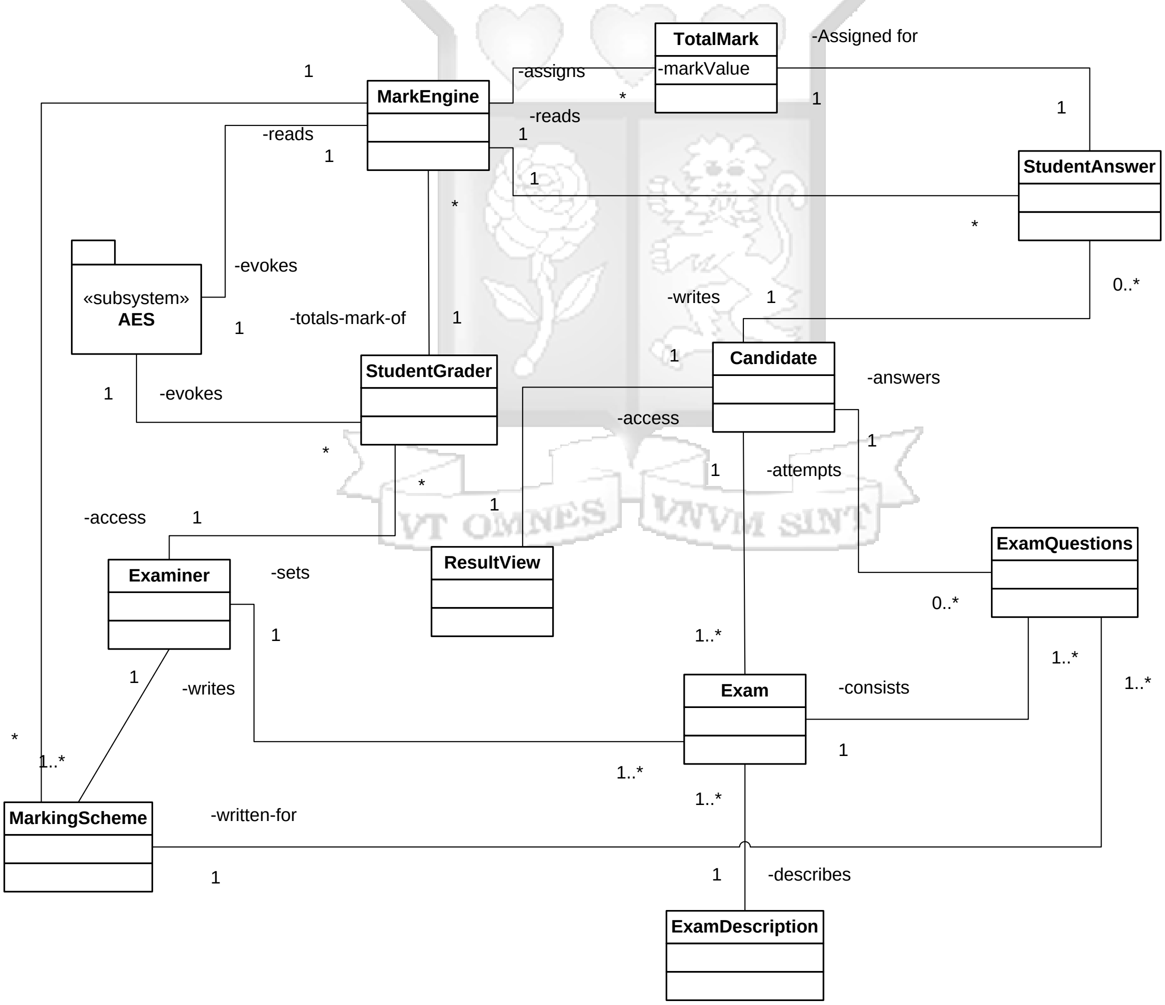


**Figure 4. 2: Artificial Exam Scorer Domain of Interest**

## 4.3. Data Model

Any computer system takes data as inputs for a process and outputs information thus data is a key element in any computing system. There are two classes of data; data at rest and data in transit. To model data at rest, Entity Relationship Diagram was used which was later developed into a relational database tables. The ERD is shown in below figure 4.3 and consists of a total 11 relational entities.

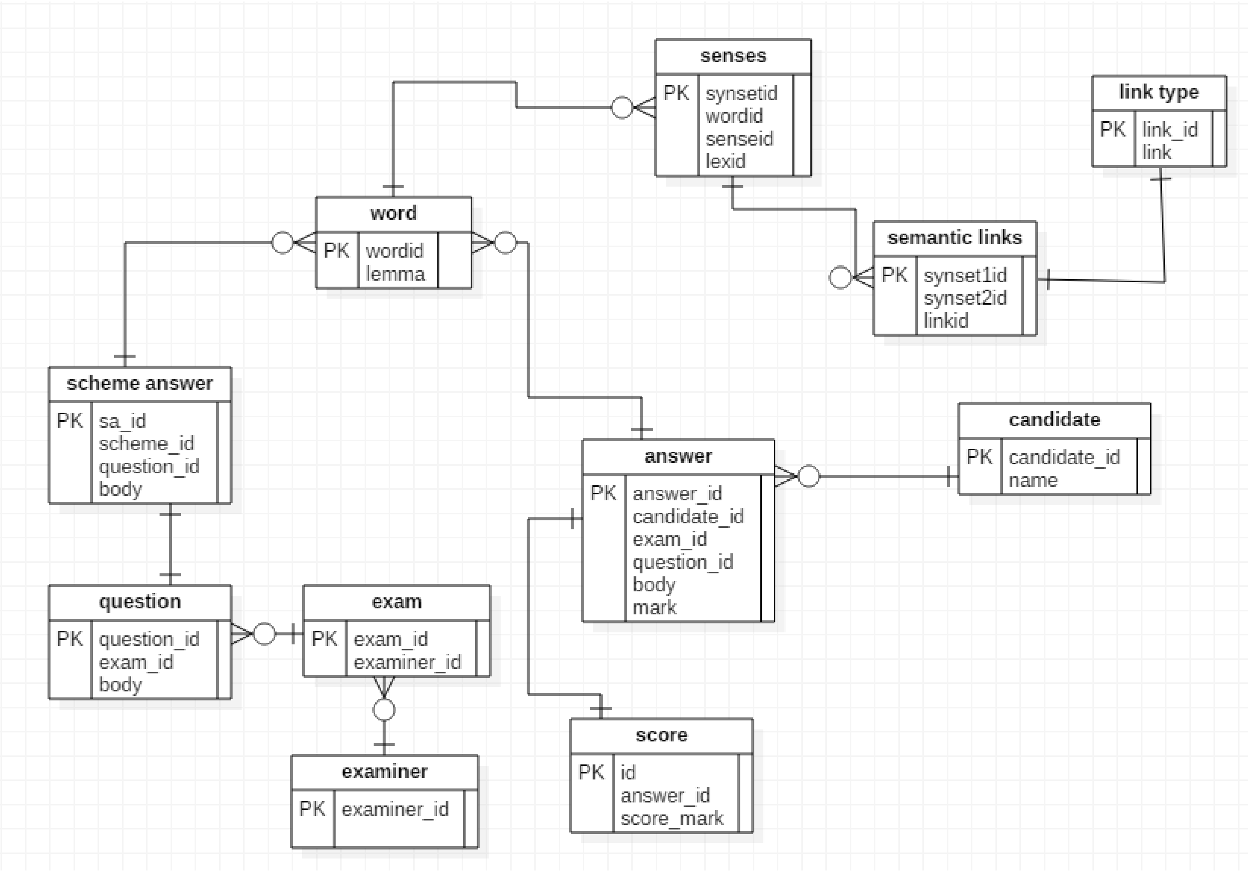


**Figure 4. 3: Entity Relational Model**

## 4.4. Process Modelling the Exam Scorer

Modelling data in transit in relation to data at rest was achieved using Data Flow Diagram starting with the context diagram level zero. The level 1 DFD shows data stores while levels to and three shows detailed processes.

### 4.4.1. Context diagram

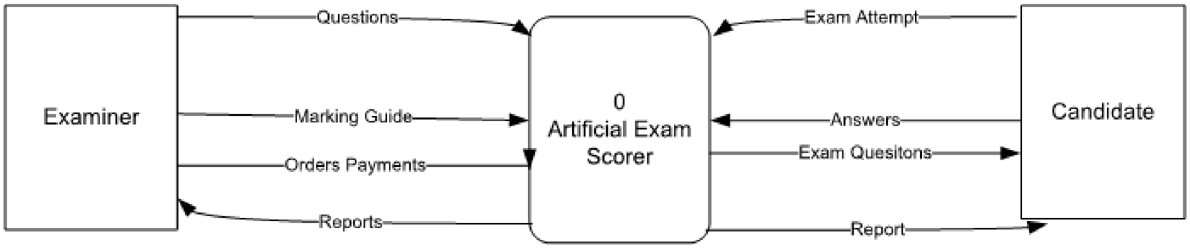


**Figure 4. 4: Artificial Exam Scorer Context Diagram**

### 4.4.2. Level 1 Data Flow

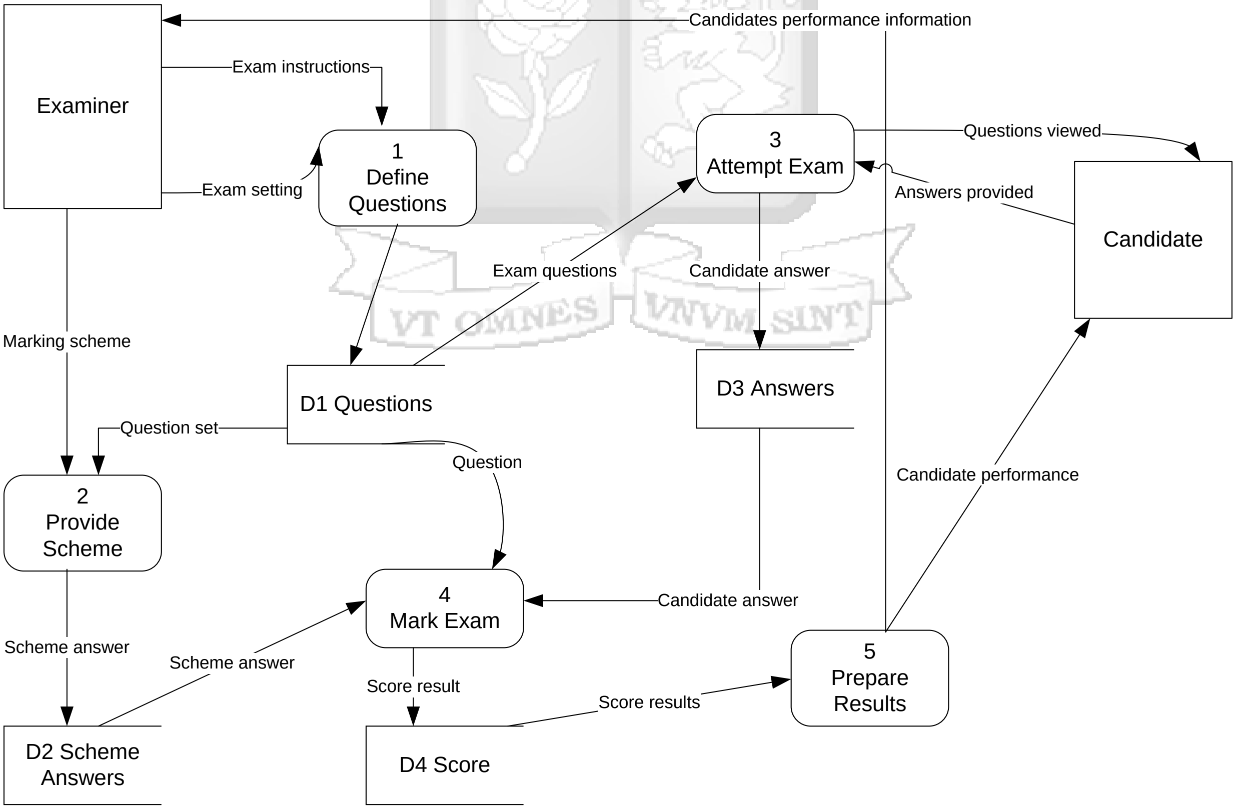


**Figure 4. 5: Level one Data Flow with Data Stores**

### 4.4.3. Level 2 Data Flow

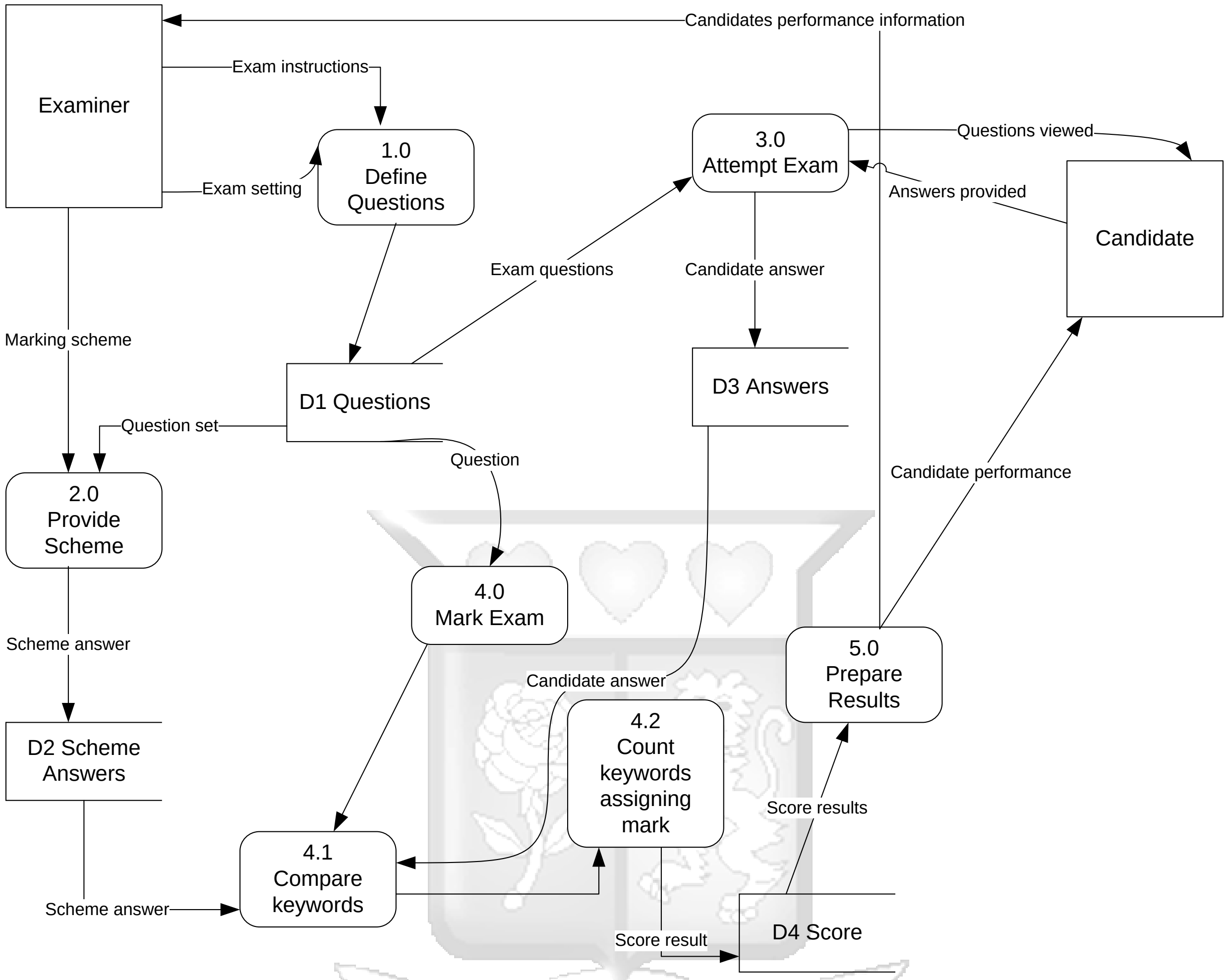


**Figure 4. 6: Level 2 Data Flow**

### 4.4.4. Level 3 Data Flow

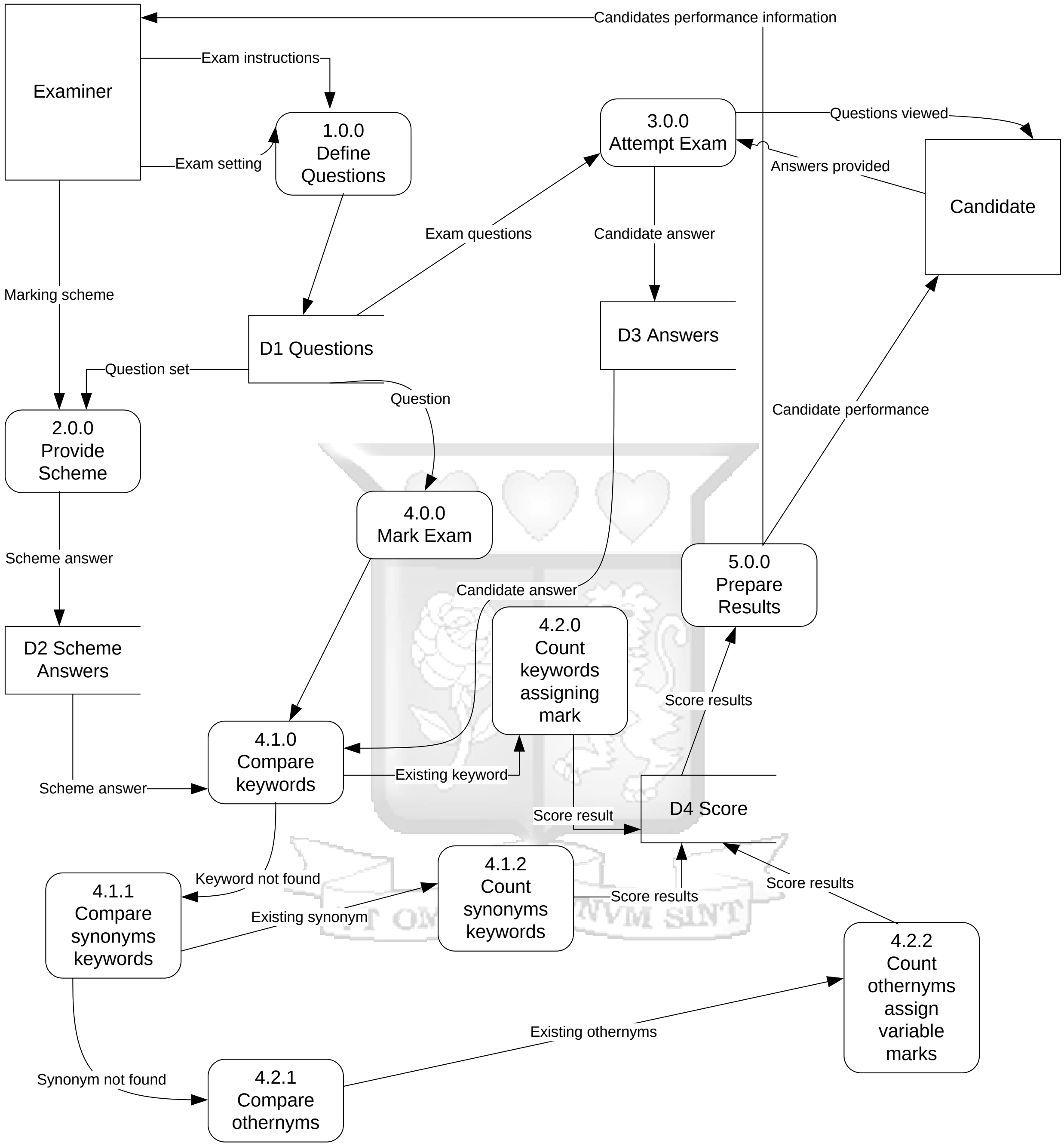


**Figure 4. 7: Level 3 Data Flow with Additional Processes**

## 4.5. System Interactivity

System interactivity models the dynamic nature of system objects. Two UML diagrams i.e. sequence and collaboration diagrams can be used to model the objects. A sequence diagram show below in figure 4.8 is shown below for the main AES engine.

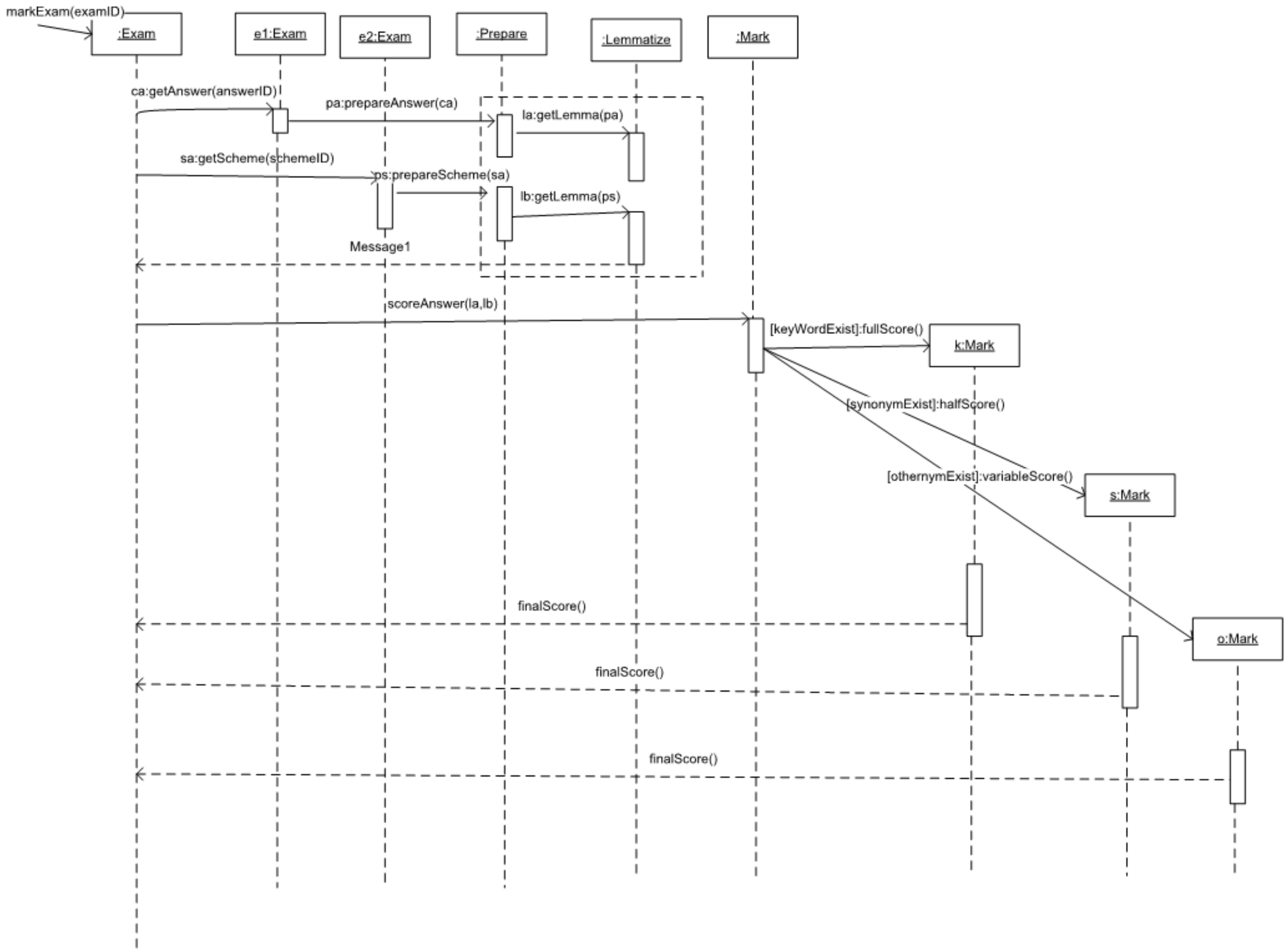


**Figure 4. 8: Artificial Exam Scorer Sequence Diagram**

# CHAPTER FIVE: ARTIFICIAL EXAM SCORER IMPLEMENTATION AND TESTING

## 5.1. Introduction

This research was centered around the efficient marking and grading of short essay answers through automated means. This is practically achieved by the development of an algorithm. Prior tried methodologies and algorithms were studied and an improved algorithm was developed. This chapter presents the tools, frameworks and methodology used in the implementation of the algorithm and the system surrounding its operation.

## 5.2. Automating the grading of exams

To automate the grading of exams the following steps were adopted by the developed algorithm:

### 5.2.1. Automated Marker Algorithm Key Steps

i. Prepare the marking scheme and student answer by POS tag the marking scheme then drop all *non-key* parts of speech i.e. *conjunctions, articles* and *interjections.* Lemmatize and Stem to find root of words for Noun *singular/plurals* and verb *tenses* remain with potential keyword parts of speech i.e. *Nouns, Verbs, Adjectives*

ii. Define depth for every keyword. Depth 1 if for synonyms, depth 2 for hyponym, depth 3 for hypernym checks respectively

iii. Build a *reduced* synonyms depth tree, without repetition, for every keyword. Child nodes must be of the same POS tag as parent node

iv. Grade the answer based on existent keywords, synonyms, hypernyms or hyponyms and depth levels by traversing the tree ADT using BFS traversal algorithm

v. Terminate algorithm when keyword or synonym found though entire tree has not been traversed or when the entire tree is traversed and neither keyword nor synonym is found

## 5.3. Algorithm Implementation

### 5.3.1. Part of Speech Tagging – Penn Treebank Tagger

Part of speech tagging is the process of identifying classes of words forming a sentence. The English language has nine parts of speech *noun, verb, pronoun, adjective, preposition, adverb, conjunction, article* and *interjection.* Each of the nine parts of speech presents a class defining the function of a given word in a sentence also called semantic tendencies.

(Jurafsky & Martin, 2017) formally describe each of the nine-word classes. The *Noun* class names people, places or things. *Verbs* refer to action and processes. *Pronouns* are words that are used in substitute of nouns. *Adjectives* present terms for properties or qualities that describe a noun or pronoun. An *Adverb* describes a verb, an adjective or another adverb. *Prepositions* occur before nouns to show semantical relations, a *Conjunction* joins two phrases, clauses or sentences, while *Interjections* express emotions or mood. All the nine classes have sub-classes e.g. class noun is subdivided into proper nouns, collective nouns and concrete nouns, and class adverb is divided into adverbs of time, place, degree and manner.

Different word classes have different importance and have differing propensity in determining keywords in a given sentence. Jurafsky and Martin (2017), expounds further on this by subdividing the nine-word classes into two major categories; open class and closed class. Closed class categories are those of fixed membership that rarely change with the ever-transitioning nature of human languages. They include *propositions, determinants, pronouns, conjunctions, auxiliary verbs, participles* and *numerals*. On the contrary, the open-class category includes word classes that are constantly receiving new members. They are *nouns, verbs, adjectives* and *adverbs*. Most keywords in a sentence (exam question and answers) belong to the open class category words as demonstrated in section 2.1.4.

Creativity in language breeds ambiguity that leads to challenges in computational linguistics. Complex word classes like *phrasal verbs* (combination of verb and participle) and sentence formations like *idiomatic expressions* possess a deeper or cultural meaning, rather than the surface meaning. In addition, some word classes have a higher probability of usage than others. For instance, interjections are rarely used while articles, mostly *the*, are highly used.

The Penn Treebank tagset, presented by Jurafsky and Martin (2017) and invented by Marcus et al. (1993) is commonly used for English corpora. It consists of 45 tags an extension of the classical nine-word classes.

**Table 5. 1: Penn Treebank part-of-speech tags (Jurafsky & Martin, 2017)**

| Tag | Description | Example | Tag | Description | Example |
|---|---|---|---|---|---|
| **CC** | coordin. conjunction | and, but, or | **SYM** | symbol | +,%, & |
| **CD** | cardinal number | one, two | **TO** | “to” | to |
| **DT** | determiner | a, the | **UH** | interjection | ah, oops |
| **EX** | Existential ‘there’ | there | **VB** | verb base form | eat |
| **FW** | foreign word | mea culpa | **VBD** | verb past tense | ate |
| **IN** | preposition/sub-conj | of, in, by | **VBG** | verb gerund | eating |
| **JJ** | adjective | yellow | **VBN** | verb past participle | eaten |
| **JJR** | adj., comparative | bigger | **VBP** | verb non-3sg pres | eat |
| **JJS** | adj., superlative | wildest | **VBZ** | verb 3sg pres | eats |
| **LS** | list item marker | 1, 2, One | **WDT** | wh-determiner | which, that |
| **MD** | modal | can, should | **WP** | wh-pronoun | what, who |
| **NN** | noun, sing. or mass | llama | **WP$** | possessive wh- | whose |
| **NNS** | noun, plural | llamas | **WRB** | wh-adverb | how, where |
| **NNP** | proper noun, sing. | IBM | **$** | dollar sign | $ |
| **NNPS** | proper noun, plural | Carolinas | **#** | pound sign | # |
| **PDT** | predeterminer | all, both | **“** | left quote | ‘ or “ |
| **POS** | possessive | ending ’s | **”** | right quote | ’ or ” |
| **PRP** | personal pronoun | I, you, he | **(** | left parenthesis | [, (, f, < |
| **PRP$** | possessive pronoun | your, one’s | **)** | right parenthesis | ], ), g, > |
| **RB** | adverb | quickly, never | **,** | comma | , |
| **RBR** | adverb, comparative | faster | **.** | sentence-final punc | .! ? |
| **RBS** | adverb, superlative | fastest | **:** | mid-sentence punc | : ; ... – - |
| **RP** | particle | up, off | | | |

*Example from marking scheme see Appendix C*

*From the answer to the question regarding the John Van Neumann computing model component number one:*

***Answer; Inputting: is*** *The processes of entering data and instructions into the computer*

Example POS tagging Marking Scheme answer is shown below:

The Penn Treebank tagged (tags in bold) version of the answer is:

The**|DT** processes**|NNS** of**|IN** entering**|VBG** data**|NNS** and**|CC** instructions**|NNS** into**|IN** the**|DT** computer**|NN**

From Penn Treebank table;

**DT:** determinant, **NNS:** noun, plural, **IN:** preposition, **VBG:** verb, **CC:** conjunction, **NN:** noun

### 5.3.2. Text Lemmatization

A number of words may have surface differences but share the same root, a difference brought about by adding suffixes, prefixes in common count nouns when defining plurals, or defining the tenses in verbs. Lemmatization is the task of determining that two words have the same root, despite their surface differences (Jurafsky & Martin, 2017). The root of the word is called *lemma*. E.g. *shelves*, *shelfing, shelfed* all have the lemma *shelf*. Word stemming is simplified lemmatization where affixes are chopped off to get the root of the word.

Lemmatization is imperative to match all keyword variations that originate from the same keyword. Morphological parsing with finite state transducers is used to achieve lemmatization.

Example from marking scheme Appendix c:

***Answer; Inputting: is*** *The processes of entering data and instructions into the computer*

*The lemmatized answer is: The process of enter data and instruction into the computer*

### 5.3.3. Synonyms, depth *n* tree and Probability Value of Replacement

Synonyms are words or phrases that mean exactly or nearly the same as another word or phrase in the same language (Miller, 1995). In this study, the term synonym is used on a broader sense to mean all related word to a given keyword i.e. hypernyms, hyponyms also called *othernyms*.

Keywords in a given sentence can be replaced by synonyms without altering the semantic structure of the sentence. Most correct student answers are closer to the instructors' marking scheme since they contain synonyms of keywords or terminologies. The closeness between a students' answer and the marking scheme can be measured by the presence of synonyms to keywords meaning that the student is providing the correct answer as dictated by the scheme, a statement alluded by the conclusion of (Frost, 2008).

Synonym depth refers to a data mining process to return as many synonyms as possible to a given keyword from a lexicon dictionary. Every given keyword has a list of synonyms, in the synonym tree this is defined as depth $n_2$ with the keyword forming the root which is depth $n_1$**.** To mine for more synonyms, a synonym of the individual synonyms at depth $n_2$ are mined and returned forming depth $n_3$**.** This process can be iterative leading to a vast synonym tree of up to depth $n_x$**.**

Below is an example from appendix c marking scheme, words which are strictly required to be part of the answer with no synonym consideration are given a depth of $n_1$.

*Q b. Clearly explain the functionality of each of the* ***5 components*** *of the Van Neumann model of computing.* ***(5 marks)***

***Marking Scheme***

***Award 1 mark for correct point out of a component and 1 mark for explanation***

***i. Inputting (depth*** $n_1$***):*** *The processes of entering data and instructions into the computer*

***ii. Storing (depth*** $n_1$***):*** *Saving data and instructions making them available for processing and output*

***iii. Processing (depth*** $n_1$***):*** *Performing arithmetic operations* ***(addition, multiplication, subtraction and division)*** *or logical operations* ***(OR, AND, NAND & NOR)*** *on data to transform them to information*

***iv. Outputting (depth $n_1$):** Producing useful information in a desired form such as audio, video or printed copies*

***v. Control (depth $n_1$):** Directing/coordinating the manner in which all the above operations are carried out*

*Synonyms Depth tree for answer **(i)***

***Answer: Inputting (depth $n_1$):** The processes of entering data and instructions into the computer*

***Normalized answer with depths:** Input ($n_1$) process ($n_3$) enter ($n_3$) data ($n_1$) instruction ($n_3$) computer ($n_1$)*

***Input, data, computer** (have depths of $n_1$ thus no depth tree constructed)*

Example of assigning depth n is demonstrated on figure 5.1.

We get a depth $n_3$ tree demonstrated partly by the non-binary tree data structure below and in detail by table of the lemmatized keyword *process.*

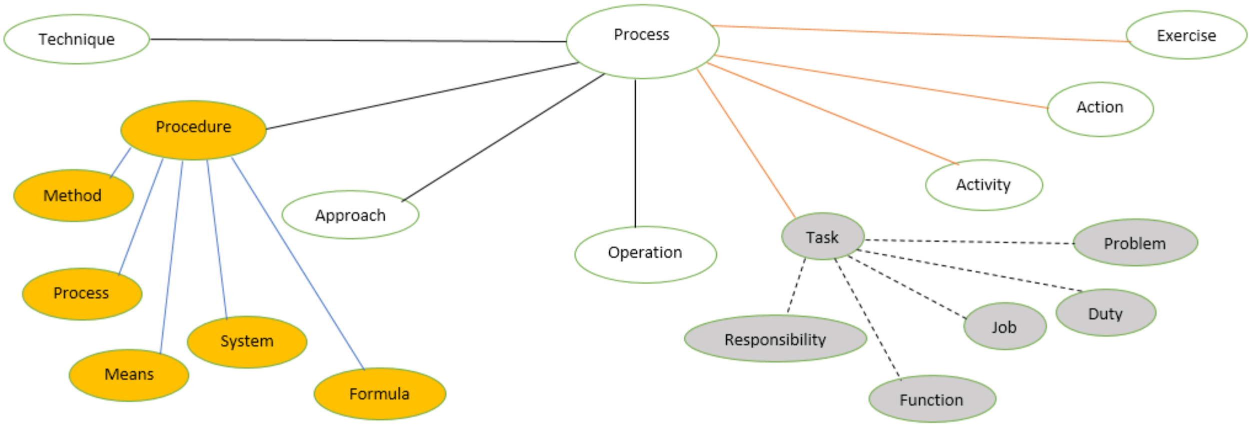


**Figure 5. 1: Synonym depth 3 Tree for keyword process**

**Table 5. 2: Synonym Distribution depth 3**

| Depth $n_x$ | Root Word - **Process** (POS tag: Noun) | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| $n_1$ | | | | | | | | | |
| $n_2$ | Procedure | Technique | Operation | Method | Approach | Action | Means | Task | Practice |
| $n_3$ | plan,<br>method,<br>system,<br>strategy,<br>way,<br>approach,<br>formula,<br>mechanism,<br>methodology,<br>technique,<br>means,<br>measure,<br>process,<br>proceeding,<br>operation,<br>agenda;<br>routine,<br>drill,<br>practice | method,<br>approach,<br>procedure,<br>process,<br>system,<br>way,<br>manner,<br>mode,<br>fashion,<br>style,<br>means,<br>strategy,<br>tack,<br>tactic,<br>line;<br>routine,<br>practice | action,<br>activity,<br>exercise,<br>affair,<br>business,<br>undertaking,<br>step,<br>enterprise,<br>task,<br>job,<br>process,<br>procedure,<br>maneuver,<br>campaign | formula,<br>process,<br>means,<br>medium,<br>mechanism;<br>tack,<br>approach,<br>way,<br>line<br>route,<br>road;<br>strategy,<br>tactic,<br>plan,<br>recipe,<br>rule<br>procedure,<br>technique,<br>system,<br>practice,<br>routine | attitude,<br>slant,<br>perspective,<br>viewpoint,<br>outlook,<br>method,<br>procedure,<br>process,<br>technique,<br>style,<br>strategy,<br>stratagem,<br>way,<br>manner,<br>mode,<br>tactic,<br>tack,<br>path,<br>system,<br>means; | activity,<br>movement,<br>work,<br>effort,<br>exertion,<br>operation<br>deed,<br>act, activity,<br>move, gesture,<br>undertaking,<br>exploit,<br>maneuvers,<br>achievement,<br>accomplishment,<br>venture,<br>enterprise,<br>endeavor, effort,<br>exertion, work,<br>handiwork,<br>doing,<br>creation,<br>performance,<br>behavior,<br>conduct;<br>reaction,<br>response | method,<br>way,<br>manner,<br>mode,<br>measure,<br>fashion,<br>process,<br>procedure,<br>technique,<br>expedient,<br>agency,<br>medium,<br>instrument,<br>mechanism,<br>channel,<br>vehicle,<br>avenue,<br>course | job,<br>duty,<br>chore,<br>charge, labor<br>assignment,<br>function,<br>commission,<br>mission,<br>engagement,<br>occupation,<br>undertaking,<br>exercise,<br>business,<br>responsibility,<br>errand,<br>detail,<br>endeavor,<br>enterprise,<br>venture,<br>quest,<br>problem,<br>burden | application,<br>exercise,<br>use,<br>operation,<br>implementation,<br>execution,<br>enactment, action,<br>doing<br>profession, career,<br>business,<br>work,<br>pursuit,<br>occupation,<br>following<br>custom,<br>procedure, policy,<br>convention,<br>tradition, fashion,<br>habit,<br>wont,<br>method, system,<br>routine,<br>institution,<br>way,<br>rule,<br>business |

**Table 5. 3: Synonym Distribution depth 3 with expected lower probability**

| **Depth** $n_x$ | Process (Noun) | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| $n_1$ | | | | | | | | |
| $n_2$ | Undertaking | Activity | System | Exercise | way | Job | affair | business |
| $n_3$ | pledge, agreement, promise, oath, covenant, vow, word bond, commitment, guarantee, assurance, warrant, contract, compact enterprise, venture, project, campaign, scheme, plan, operation, endeavor, effort, task, assignment, charge, activity, pursuit, exploit, job, business, affair, procedure, process, transaction; mission, quest, exploration, expedition | action, liveliness, movement, life, stir, animation, commotion, flurry, tumult, hubbub, excitement, agitation, fuss, whirl pursuit, occupation, venture, undertaking, enterprise, project, scheme, business, job, affair, task, campaign; interest, hobby, pastime, recreation, diversion, entertainment; act, action, deed, doing, exploit, maneuver | structure, organization, order, arrangement, complex, apparatus, network; administration, institution attack, means, way, manner, mode, framework, scheme, plan, policy, programme, regimen method, methodology, technique, process, procedure, approach, practice, line | movement, exertion, effort, work training gymnastics aerobics, step aerobics, jogging, running aquarobics, callisthenics, isometrics, eurhythmics, keep-fit, dancercise, bodybuilding problem, assignment task | process, procedure, technique, system, plan, strategy, scheme, means, mechanism, routine, manner, approach, route, road, method practice, habit, custom, characteristic, policy, convention, fashion, use, routine, rule; trait, attribute, mannerism, peculiarity, idiosyncrasy, oddity; conduct, behavior, manner, style, nature, personality, temperament, disposition | position, post, situation, place, appointment, posting, placement, occupation, profession, trade, career, work | event, incident, happening, occurrence, phenomenon, eventuality, episode, interlude, circumstance adventure, experience, case, matter, business, proceedings | Work line, occupation, profession, career, employment, job, position, pursuit, vocation, calling, field, sphere concern, affair, responsibility, province, preserve, duty, function, task, assignment, obligation, problem, worry, lookout |

The table is further reduced by pruning repetitions and zero probability entities. The unique depth $n_3$ synonyms can then be used to mine depth $n_4$ synonyms. This iteration continues to depth $n_x$.

**Table 5. 4: Reduced Synonym Distribution Table**

| Depth $n_x$ | Root Word - **Process** (POS tag: Noun) | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| $n_1$ | | | | | | | | | |
| $n_2$ | Procedure | Technique | Method | way | Approach | System | Means | Operation | Undertaking |
| $n_3$ | plan,<br>strategy,<br>formula,<br>mechanism,<br>methodology,<br>measure,<br>proceeding,<br>agenda,<br>routine,<br>drill, | manner,<br>mode,<br>fashion,<br>style,<br>tack,<br>tactic,<br>line | medium,<br>route,<br>road,<br>recipe,<br>rule | scheme,<br>habit,<br>custom,<br>characteristic,<br>policy,<br>convention,<br>use,<br>trait,<br>attribute,<br>peculiarity,<br>idiosyncrasy,<br>oddity,<br>conduct,<br>behavior,<br>nature,<br>personality,<br>temperament,<br>disposition | attitude,<br>slant,<br>perspective,<br>viewpoint,<br>outlook,<br>path | structure,<br>organization,<br>order,<br>arrangement,<br>complex,<br>apparatus,<br>network;<br>administration,<br>institution<br>attack,<br>framework,<br>programme,<br>regimen | expedient,<br>agency,<br>instrument,<br>channel,<br>vehicle,<br>avenue,<br>course | step,<br>enterprise,<br>maneuver,<br>campaign | pledge,<br>agreement,<br>promise,<br>oath,<br>covenant,<br>vow, word<br>bond,<br>commitment,<br>guarantee,<br>assurance,<br>warrant,<br>contract,<br>compact<br>enterprise,<br>venture,<br>project,<br>campaign,<br>operation,<br>endeavor,<br>effort,<br>assignment,<br>charge,<br>pursuit,<br>exploit<br>transaction,<br>mission,<br>quest,<br>exploration,<br>expedition |

### 5.3.4. Pragmatics, context and rational synonyms

The pragmatics and discourse nature of natural language dictates the rationality of meaning within the context in which communication takes place. Some synonyms are deemed suitable than others to replace a given keyword in a given context. This study proposes the calculation of probability value of replacement for every given synonym that is to replace a keyword.

***Probability value of replacement:*** *The probability that a synonym can logically replace a root word in the synonym depth tree within a given context without altering the sentence meaning.*

Example from marking scheme appendix c, in the answer; Inputting is *The processes of entering data and instructions into the computer*, the keyword **process**, (lemma of processes), has the following synonyms mined from the lexical dictionary of synnates:

*Procedure, Technique, Operation, Method, Approach, Action, Means, Task, Practice, Undertaking, Activity, System, Exercise, Way, Job, Affair, Business*

Within the context of the answer, the first nine synonyms are intuitively suitable in replacing the keyword Process e.g. the answer reading as:

***Inputting is*** *The <u>procedure/technique</u> of entering data and instructions into the computer,*

*Sounds an acceptable answer. However, the last eight synonyms seem less probable in intuitively presenting a correct answer within the given context e.g. if the answer reads:*

***Inputting is*** *The <u>business/job</u> of entering data and instructions into the computer,*

*Sounds a half correct answer if not a wrong answer.*

Since in the examination environment students strive to write correct answers, some synonyms to keywords are less likely to occur than others. This presents a concern on synonym probability calculation to determine which synonyms can correctly be considered to replace the root word and accepted as a constituent part of the correct answer.

This study proposes a computation of *probability value of replacement* ($P_{vr}$) drawn from the probability multiplication rule of independent events (Jaynes, 1995). This is calculated as a joint probability of the existence of the *root word* $\exists_{rw}$as a synonym in the **synonym of synonyms** at depth $n_x$ $x \neq 1$ of the synonyms tree where the tree contains depths $n_1, n_n, \dots., n_x$ with depth $n_1$ containing the **root word denoted** $rw$ and each successive depth $n_2, n_3, \dots, n_x$ containing $i$ synonyms denoted $S^{depth}_{position} S^x_i$ $starting\ at\ i = 1$ at depth of label $x$.

$$P_{vr}\ (S_i^x) = \mathrm{P}(\exists_{rw} \text{ and } \exists S_i^{x+1})$$

$$= \mathrm{P}(\exists_{rw}) * \mathrm{P}\left(\exists S_i^{x+1}\right)$$

$$= \frac{C(\exists_{rw})}{C(rw)} * \frac{C\left(\exists S_i^{x+1}\right)}{C(S_i^x) - 1}$$

**Equation 5. 1: Joint Probability of Replacement**

**Table 5. 5: Probability tree nodes and synonym members. Root node contains keyword**

| DEPTH | NODE MEMBERS |
|---|---|
| $\boldsymbol{n_1}$ | $rw$ |
| $\boldsymbol{n_2}$ | $S_1^2, S_2^2, \dots., S_i^2$ |
| … … … … | … … … … |
| $\boldsymbol{n_x}$ | $S_i^x, S_{i+1}^x, \dots., S_{i+m}^x$ |

Probability is calculated by counting (denoted as $C$) the existent synonyms $S$ at depth label $x + 1$ that form the tree nodes at depth label $x$ and dividing by the actual number of synonyms, less the current synonym in consideration, at a defined depth of label $x$.

From appendix c scheme, we calculate the following probabilities of replacement for the keyword **process** for some depth $n_2$ synonyms using Eq. 5.1:

$$P_{vr}\ (\text{Procedure}) = \frac{1}{1} * \frac{8}{16}$$

$$= 0.5$$

$$P_{vr}\ (\text{Technique}) = \frac{1}{1} * \frac{6}{16}$$

$$= 0.375$$

$$P_{vr}\ (\text{Operation}) = \frac{1}{1} * \frac{9}{16}$$

$$= 0.56$$

$$P_{vr}\ (\text{Method}) = \frac{1}{1} * \frac{7}{16}$$

$$= 0.437$$

Other synonyms have a $P_{vr}$ of $0$ since their $P(\exists_{rw}) = 0$ leading to a multiplication by $0.$

$$P_{vr}\ (\text{Business}) = 0$$

$$P_{vr}\ (\text{Affair}) = 0$$

$$P_{vr}\ (\text{Job}) = 0$$

Table below shows the $P_{vr}$ for all depth $n_2$ synonyms

**Table 5. 6: Probability value of replacement table**

| $i$ | 1 | 2 | 3-6 | 7 | 8 | 9 | 10-17 |
|---|---|---|---|---|---|---|---|
| **Synonyms** $\boldsymbol{S_i^x}$ $x = 2$ | Operation | Procedure | Method/Way/<br>System/Undertaking | Technique | Approach | Means | Task/Action/<br>Exercise/Practice/<br>Activity/Job/<br>Affair/Business |
| $\boldsymbol{P_{vr}}\,(S_i^x)$ | $\mathbf{0.56}$ | $\mathbf{0.5}$ | $\mathbf{0.437}$ | $\mathbf{0.375}$ | $\mathbf{0.312}$ | $\mathbf{0.25}$ | **0** |

The algorithm thus marks any of the following as a correct answer:

> ***Inputting is*** *The **process**/operation/procedure/method/way/system/undertaking/technique/approach/means of entering data and instructions into the computer*

Example of marking with keyword probability is shown below:

Synonyms with P $(\exists_{rw}) = 0$ can be ignored since they are less related to the root word $rw$ thus much rational probability can be computed by modifying Eq. 5.1:

$$P_{vr}\,(S_{i'}^{x'}) = \frac{C(\exists_{rw})}{C(rw)} * \frac{C\left(\exists S_{i'}^{x'}\right)}{C\left(S_{i'}^{x'}\right) - 1}$$

**Equation 5. 2: Joint Probability of replacement with non-zero probability entries**

Where $S_{i'}^{x'}$ is a subset of $S_i^x$ where $\exists_{rw}$ is always true at depth $n_{x+1}$. Since $\exists_{rw}$ is always true, equation $P_{vr}$ can be simplified as:

$$P_{vr}\,(S_{i'}^{x'}) = \frac{C\left(\exists S_{i'}^{x'}\right)}{C\left(S_{i'}^{x'}\right) - 1}$$

**Equation 5. 3: Simplified Eq. 5.2**

$P_{vr}$ Values calculated from Eq. 2.5 are more accurate as shown in table below:

**Table 5. 7: Modified Probability value of replacement**

| $i'$ | 1 | 2-4 | 5-6 | 7 | 8-9 |
|---|---|---|---|---|---|
| **Synonyms**<br>$S_{i'}^{x'}$<br>$i' = 9$ | Procedure | Technique/Method/<br>Way | Approach/System | Means | Operation/Undertaking |
| $P_{vr}\,(S^j)$ | $\mathbf{0.875}$ | $\mathbf{0.75}$ | $\mathbf{0.625}$ | $\mathbf{0.5}$ | $\mathbf{0.312}$ |

The keyword ***procedure*** is more likely to be a correct answer substitute to keyword ***process*** than word ***undertaking***. The much correct answer from the algorithm is now:

> ***Inputting is*** *The **process/** procedure/technique/method/way/approach/system/means/operation/undertaking of entering data and instructions into the computer*

### 5.3.5. Tree Data Structure

A tree data structure provides way to organize data in a recursive fashion presenting a way to perform key operations such as insertion, search and sorting in a computationally efficient manner. Any given tree is a collection of $k$ **nodes** and $k - 1$ **edges**. The origin node is known as the **root node** say $r$ from where other **subtrees**, $T_1, T_2, \ldots \ldots., T_X$ arise. The root of every subtree is a **child** of $r$ and $r$ is the **parent** of each subtree root. Every node, except the root node, has one parent.

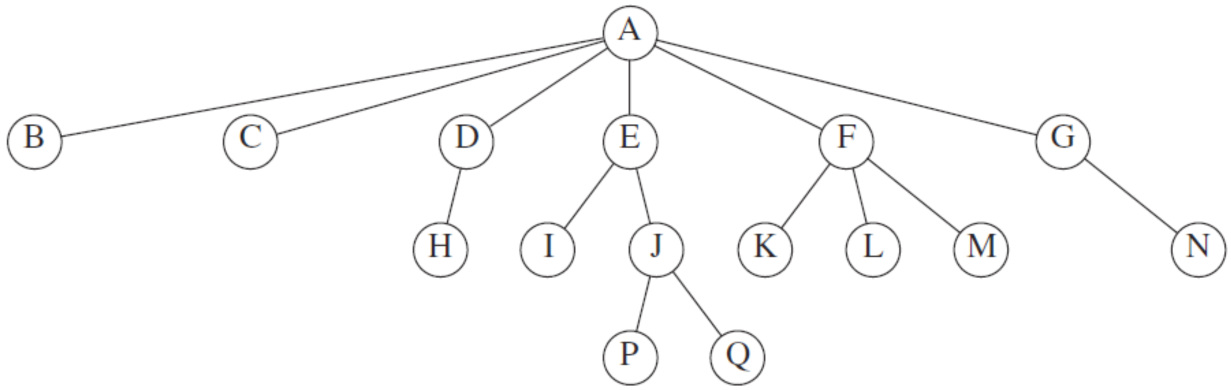


**Figure 5. 2: Tree ADT**

From the above tree adopted from (Weiss, 2014), Node $F\ has\ A$ as a **parent** and $K, L, and\ M$ as **children**. **Leaf Nodes** are those with no children i.e. $B, C, H, I, P, Q, K, L, M, and\ N$. **Siblings** are Nodes with the same parent $K, L, and\ M$ are all siblings.

The tree can be traversed using *in-order*, *preorder* or *post-order* algorithms to visit every node in search of existent keywords.

**a. Pre-order Traversal Algorithm**

In pre-order traversal, the algorithm visits Node *before* visiting its children. For the above tree, the algorithm will output:

*A, B, C, D, H, E, I, J, P, Q, F, K, L, M, G, N*

**b. Post-order Traversal Algorithm**

Post-order traversal visits Node *after* child. The algorithm will output:

*B, C, H, D, I, P, Q, J, E, K, L, M, F, N, G, A*

**c. In-order Traversal Algorithm**

An in-order traverser first visits the left child, including its entire subtree, then visits the node, and finally visits the right child, including its entire subtree. This algorithm is complex for non-binary trees. The traverser outputs:

*B, C, A, H, D, I, P, Q, J, E, K, L, F, M, G, N*

### d. Breadth First Search Algorithm

BFS algorithm visits depth $d$ before visiting depth $d_i$. This traversal algorithm is implemented in the algorithm.

*A, B, C, D, E, F, G, H, I, J, K, L, M, N, P, Q*

### 5.3.6. Depth *n* probability synonym Algorithm Pseudocode

The synonyms depth tree is constructed from synnates depth as defined in a lexicon dictionary. For every given keyword (depth $n_1$), synnates are mined to form depth $n_2, n_3, \dots .., n_x$. Every successive depth $n_x$ after $n_2$, is when ***other-nyms*** are mined from the same lexicon.

```
START

INPUTS: StudentAnswer, NormalizedMarkingScheme, SynonymsDictionary

OUTPUTS: StudentAnswerScore

PROGRAM <ScoreStudentAnswer>:

teacherKeywords←depthTag(NormalizedMarkingScheme)

FOR EACH keyword k in teacherKeywords with depth d:

     allKeywordVariations←getSynates(k,d,SynonymsDictionary)//tree ADT

ENDFOR
```

```
FOR answer $a \in$ StudentAnswer:

$UnitMark = 0$

FOR EACH keywordVariation $\boldsymbol{kv}$ of depth $\boldsymbol{d}$ && probability $p$ in allKeywordVariations

        IF $\boldsymbol{kv}$ $exist\ in\ a$:

                $UnitMark = UnitMark + \ (d^{-1} * p)$

        ENDIF

ENDFOR

$RawMark = \frac{UnitMark}{Count(allKeywordVariations)}$

$PriorScore \ = \frac{2}{1 + \ e^{-2(RawMark)}} - 1$

    IF $ore \ > 0.46$:

                $StudentAnswerScore \ = T$

        ELSE IF $PriorScore \ \geq 0.3$ && $\leq 0.46$:

                $StudentAnswerScore \ = \ 0.5 * T$

        ELSE

                $StudentAnswerScore \ = \ 0$

    ENDIF

ENDFOR
```

```
return StudentAnswerScore

END



FUNCTION depthTag(NormalizedMarkingScheme)

OpenClass = nouns, verbs, adjectives, adverbs

ClosedClass = propositions, determinants, pronouns, conjunctions, auxiliary verbs,
participles, numerals

    FOR EACH keyword k in NormalizedMarkingScheme

IF POStag ∈ ClosedClass && POStag != numerals

    Delete k

         ELSE IF POStag ∈ OpenClass && POStag = nouns

              depth = 2

         ELSE IF POStag ∈ OpenClass && POStag != nouns

              depth = 3

         ENDIF

    ENDFOR

return (k,depth)

END

FUNCTION getSynates(k,d,SynonymsDictionary)

    FOR range(1,d)

         allDepthdSynates,d ← Scan  SynonymsDictionary  return  all
synonyms of k
```

```
    END FOR

    FOR EACH synates $s$ in allDepthdSynates of depth $< d$

        $P_{vr}(s) = \frac{Count(synates\, s' \in alldepth\, d-1Synates\,)}{Count(alldepth\, d-1Synates)\, -1}$

        Depthntree ← for $P_{vr}(s) \neq 0$ add new Node n to the left if $P_{vr}(s) > \forall(P_{vr}(s))$ to the right otherwise

    return(Depthntree,d, $P_{vr}(s)$)

END
```

## 5.4. Grading answers with marks distribution to keywords of depth $n_x$

The depth $n$ keyword approach considers keywords presence as the interpretation of a correct answer. The relationship between question key-verbs, required answers and expected marks can be used to mark a given answer (see section 2.1.4). This research proposes a mark distribution approach to achieve traditional marking described in the formal theories of section 2.1.

### 5.4.1. Fuzzy mark approach

Any given scheme answer $A_i$ contains a series of keywords$\kappa_1, \kappa_2, \dots, \kappa_t$. Each of these keywords are assumed to have the same *weight* in determining the value of correctness of an answer, thus they are given a *unit weight* of 1.

> **The *keyword equal weight level $n_1$ assumption:***
>
> *Every filtered keyword $\boldsymbol{\kappa_t}$ has an equal weight in determining the correctness of its attached answer $\boldsymbol{A_i}$. Rephrased as; Answer ideas are uniformly distributed among its constituent keywords.*

To arrive at an awarded mark $m$, the following formulae is applied:

$$m = \frac{C\left(\exists \kappa_t^i\right)}{C\left(\kappa_t^i\right)} * T$$

**Equation 5. 4: Awarded Mark Equation**

All the existing student keywords $\exists \kappa_t^i$ are counted and divided by the total number of scheme keywords $\kappa_t^i$ and the result is multiplied by the *total possible mark* denoted as $T$.

### 5.4.2. Binary mark approach with hyperbolic tangent activation function

The above approach leads to award of marks on a *continuous scale*. This means that irrational number and none-classical fraction markings such as $0.35, 0.6666, 1.257$ are possibilities and common to be awarded. On the contrary, the classical approach in marking has been to award rational number marks such as $1, 0.5, 2, 5$. To maintain the *classical marking* *approach*, the equation can be modified to make use of a *scaled sigmoid curve*, the hyperbolic tangent, as an activation function to fire binary like markings.

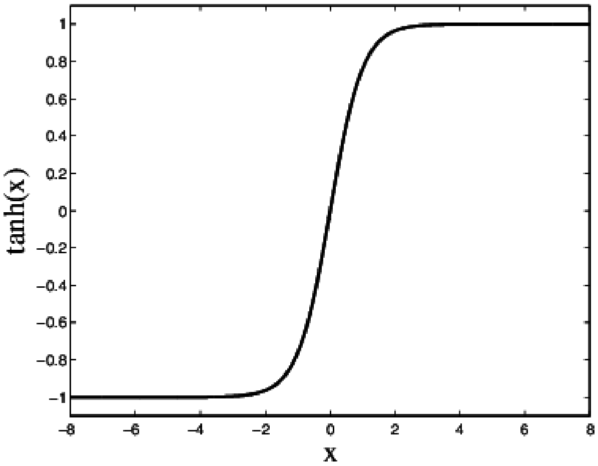


**Figure 5. 3: Hyperbolic tangent curve (Jaynes, 1995)**

The tanh curve is used within the bound of (0,1) which distributes marks within the range(0,0.7615).

**Table 5. 8: Hyperbolic tan mark distribution example (0-5) null-mark, (6-9) half-mark, (9-18) full-mark**

| $C(\exists\kappa_t^i)$ | **0** | **1** | **5** | **6** | **9** | **13** | **18** |
|---|---|---|---|---|---|---|---|
| $l = \frac{C(\exists\kappa_t^i)}{C(\kappa_t^i)}$ | 0 | 0.055 | 0.277 | 0.333 | 0.5 | 0.722 | 1 |
| $\tanh l$ | ***0*** | ***0.055*** | ***0.271*** | ***0.3215*** | ***0.460*** | ***0.618*** | ***0.7615*** |

The above table is inspired by contemporary grading scheme shown in Table 5.9.

**Table 5. 9: Contemporary grade percentage distributions**

| Grade | Percentage Boundary |
|---|---|
| **A** | 70-Above |
| **B** | 60-69 |
| **C** | 50-59 |
| **D** | 40-49 |
| **E** | 0-39 |

This modifies Eq. 5. 4 to:

$$m' = \tanh m$$

$$m' = \frac{2}{1 + e^{-2m}} - 1$$

$$m' = \frac{2}{1 + e^{-2\left(\frac{C(\exists\kappa_t^i)}{C(\kappa_t^i)}\right)}} - 1$$

**Equation 5. 5: Award Mark Equation with tanh activation function**

A range can be set such as:

$$m' > 0.46 \equiv 1 \text{ (full mark)}$$

$$0.3 \leq m' \leq 0.46 \equiv 0.5 \text{ (half mark)}$$

$$m' < 0.3 \equiv 0 \text{ (null mark)}$$

From the above limits, if the mark to be awarded $m'$ is greater than 0.46 *fire* a full mark $(e.g. 1,2 \; or \; 5)$, if it's between 0.3 $(inclusive)$ $and$ 0.46 $(inclusive)$ award the answer a half mark, else when it is less than 0.3 award zero mark.

***Example:** Awarding marks to a student answer that contains **6 keywords** out of possible **18 scheme answers keywords** for the normalized scheme answer of Question **2a. appendix c** which has a total award of **2 marks i.e.** $\boldsymbol{T = 2}$.*

***Normalized scheme answer:***

*circuit turn use model binary logic binary number system two state 1 on 0 off simplify digital logic*

*Eq. 2.8 Marks the answer as:*

$$m \ = \frac{6}{18} * 2$$

$$= 0.33333333333 \ldots \ldots . * 2$$

$$= 0.66666 \ldots \ldots.$$

$$m' = \frac{2}{1 + \ e^{-2(0.3333\ldots)}} - 1$$

$$m' = 0.3215$$

$$m' \ > 0.3$$

∴ *a half mark awarded i.e.* $\frac{1}{2} * 2 = 1 \, mark$

The above equation Eq. 5.5 marks depth $n_1$ keywords. For successive depths, the *equal weight assumption* cannot assign a unit weight of **one** since a synonym answer is only a *variation* of the true scheme answer. Thus, unit weight need to be adjusted for every successive depth; $for\ n_1\ it\ was\ 1, for\ n_2\ it\ transforms\ to\ 2^{-1}, n_3\ to\ 3^{-1}, \ldots., n_x\ to\ x^{-1}$.

In addition to depth consideration in keyword weight, the probability that a synonym can replace a keyword within context $P_{vr}$ is key in calculating synonym weight. Logically, different synonyms have different chances of replacing a given keyword thus the change in weight in relation to this probability measure. The calculation for a keyword weight is given by:

$$W_k = \ x^{-1} P_{vr}(\kappa_t^i)$$

This is the inverse of the depth $x$ multiplied by probability of keyword. This translate Eq. 2.7 to a series of additions depth, probability products for every keyword. Since depth $n_1$ $x = 1$ and a keyword has a probability of 1 to replace itself, the unit keyword assumption still holds at **one** for this *special root* depth. Eq. 5.6 numerator $C\left(\exists\kappa_t^i\right)$ transforms to the below series for keyword $\kappa_t$ with keyword depth of $x_{\kappa_t}^{-1}$:

$$C\left(\exists\kappa_t^i\right) = \ [x_{\kappa_1}^{-1} P_{vr}(\kappa_1)] + \left[x_{\kappa_2}^{-1} P_{vr}(\kappa_2)\right] +, \dots \dots, + \left[x_{\kappa_t}^{-1} P_{vr}(\kappa_t)\right]$$

$$= \sum_{t=1}^{t} x_{\kappa_t}^{-1} P_{vr}(\kappa_t)$$

**Table 5. 10: Weight distribution table**

| **Depth**<br>$n_x$ | $x^{-1}$ | **Example from table 2.10** | $W_k$ | | | |
|---|---|---|---|---|---|---|
| **1** | $1^{-1}$ | Process | $1^{-1} \ * 1 = 1$ | | | |
| **2** | $2^{-1}$ | Procedure<br>Technique<br>Means<br>Undertaking | Procedure | Technique | Means | Undertaking |
| | | | $2^{-1} * 0.875$<br>$= 0.4375$ | $2^{-1} * 0.75$<br>$= 0.2187$ | $2^{-1} * 0.5$<br>$= 0.25$ | $2^{-1} * 0.312$<br>$= 0.156$ |

## 5.5. Tools of Implementation

PHP 7 was the main programming language used to develop the automated engine. Object Oriented Programming approach was used thus classes and objects were defined. The implementation included seven classes as per the design of the class diagram. These were the *Exam*, *Mark*, *Synonym*, *User*, *DB*, *Prepare* and *Accuracy* classes.

Relational Database Management System was used for data storage written in MySQL query language with XAMPP being the main engine. The database consisted of 11 tables namely *exam, examiner, question, answer, candidate, synnate, score, senses, semlink, linktype* and *word* as displayed in the entity relationship diagram of chapter four.

Hypertext MarkUp Language (HTML - 5) and CSS 3 were used for scripting and styling front-end webpage. The used entities includes tables, div containers, headings and paragraphs. All the above are summarized in Table 5.11.

**Table 5. 11: Tools of implementation summary**

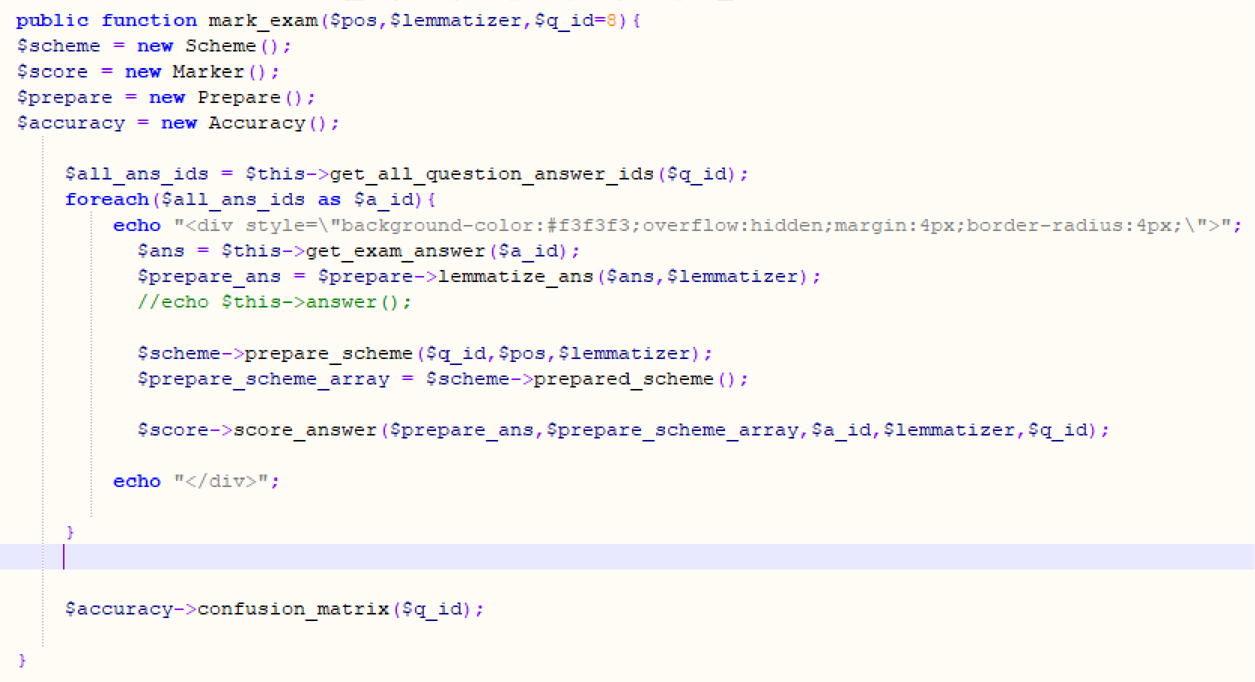
| TOOL | DEFINITION | PURPOSE |
|---|---|---|
| **PHP 7** | Programming language mainly used in back-end web application development | Main programming language |
| **MySQL** | RDBMS query language variation of SQL | Main database query language |
| **XAMPP** | Freely available apache web server | Main web server |
| **HTML - 5** | Mark Up language used for front-end web development | Scripting language |
| **CSS 3** | Styling language used to achieve front-end user experience in web development | Styling language |
| **Notepad ++** | Lightweight freely downloadable Text editor | |

## 5.6. Data Cleansing Procedure

Data set used was per chapter 3 specifications. Pre-marked and graded exams together with their respective marking schemes were sampled, studied, cleaned and fed into the system by the researcher. Gold standard label were obtained through human review of the mark awarded to each question and resolving bias. The clean data set was then manually typed into the system for processing by the AES engine and storage into the database.

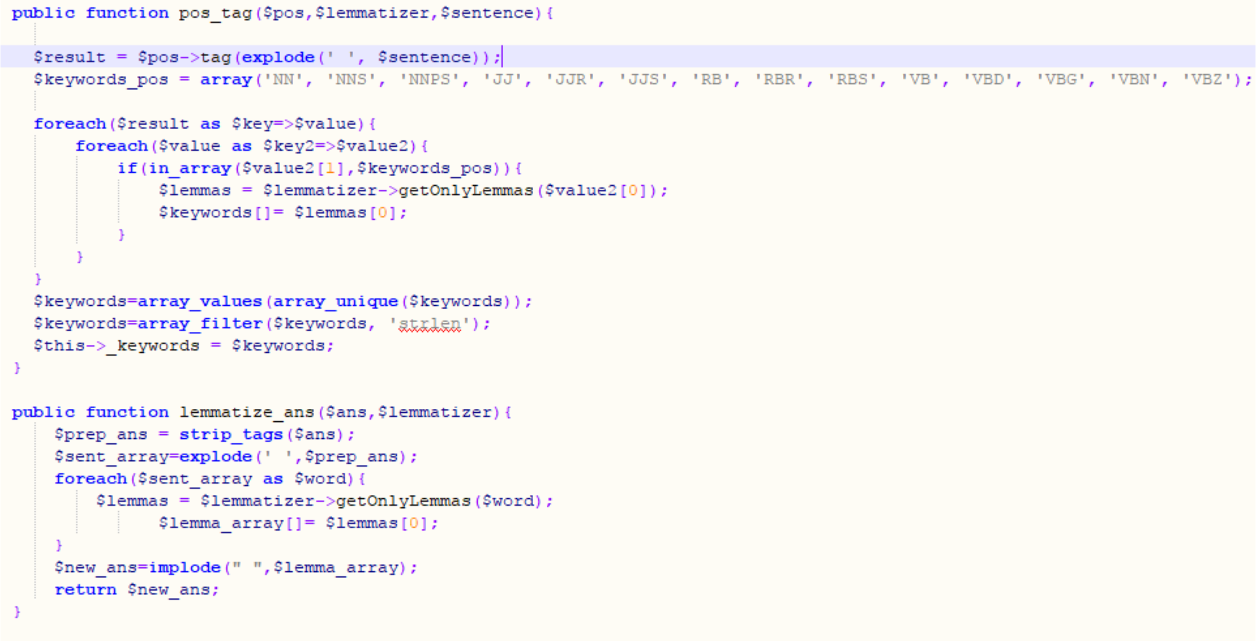
## 5.7. Artificial Exam Scorer Engine

This formed the main implementation of the defined algorithm. Marking and grading was automated in this engine written in PHP-7. The main method first engaged in the engine is method mark_exam() which exists in class Exam that takes in the question_id to be marked as a parameter with instances necessary for the preparation of the question being featured as shown below.



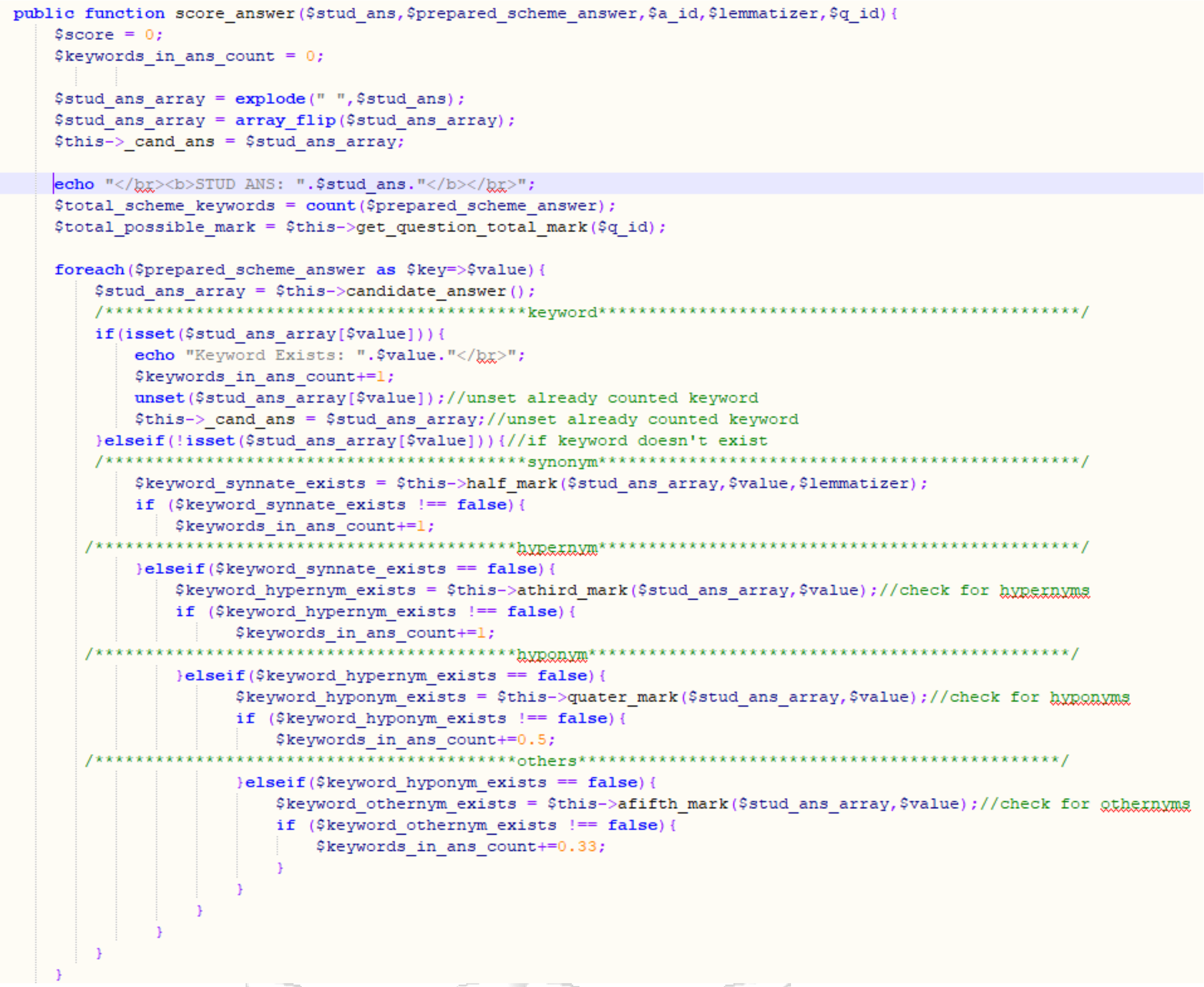
```php
public function mark_exam($pos,$lemmatizer,$q_id=8){
$scheme = new Scheme();
$score = new Marker();
$prepare = new Prepare();
$accuracy = new Accuracy();

    $all_ans_ids = $this->get_all_question_answer_ids($q_id);
    foreach($all_ans_ids as $a_id){
        echo "<div style=\"background-color:#f3f3f3;overflow:hidden;margin:4px;border-radius:4px;\">";
          $ans = $this->get_exam_answer($a_id);
          $prepare_ans = $prepare->lemmatize_ans($ans,$lemmatizer);
          //echo $this->answer();

          $scheme->prepare_scheme($q_id,$pos,$lemmatizer);
          $prepare_scheme_array = $scheme->prepared_scheme();

          $score->score_answer($prepare_ans,$prepare_scheme_array,$a_id,$lemmatizer,$q_id);

        echo "</div>";

    }


    $accuracy->confusion_matrix($q_id);

}
```

The score_answer() method found in the Mark class is the second method engaged. This method takes five parameters inclusive of the prepared candidate and scheme answers. answer preparation is done by POS tagging the answer and marking scheme in order to filter out *conjunctions*, *articles* and *interjections* remaining with *nouns*, *verbs*, *adjectives* and *adverbs* which are then lemmatized to find root words creating uniformity by eliminating affixes *(suffixes, prefixes)* that define plural for *nouns* and tenses for *verbs*. These open-class word then form keywords.

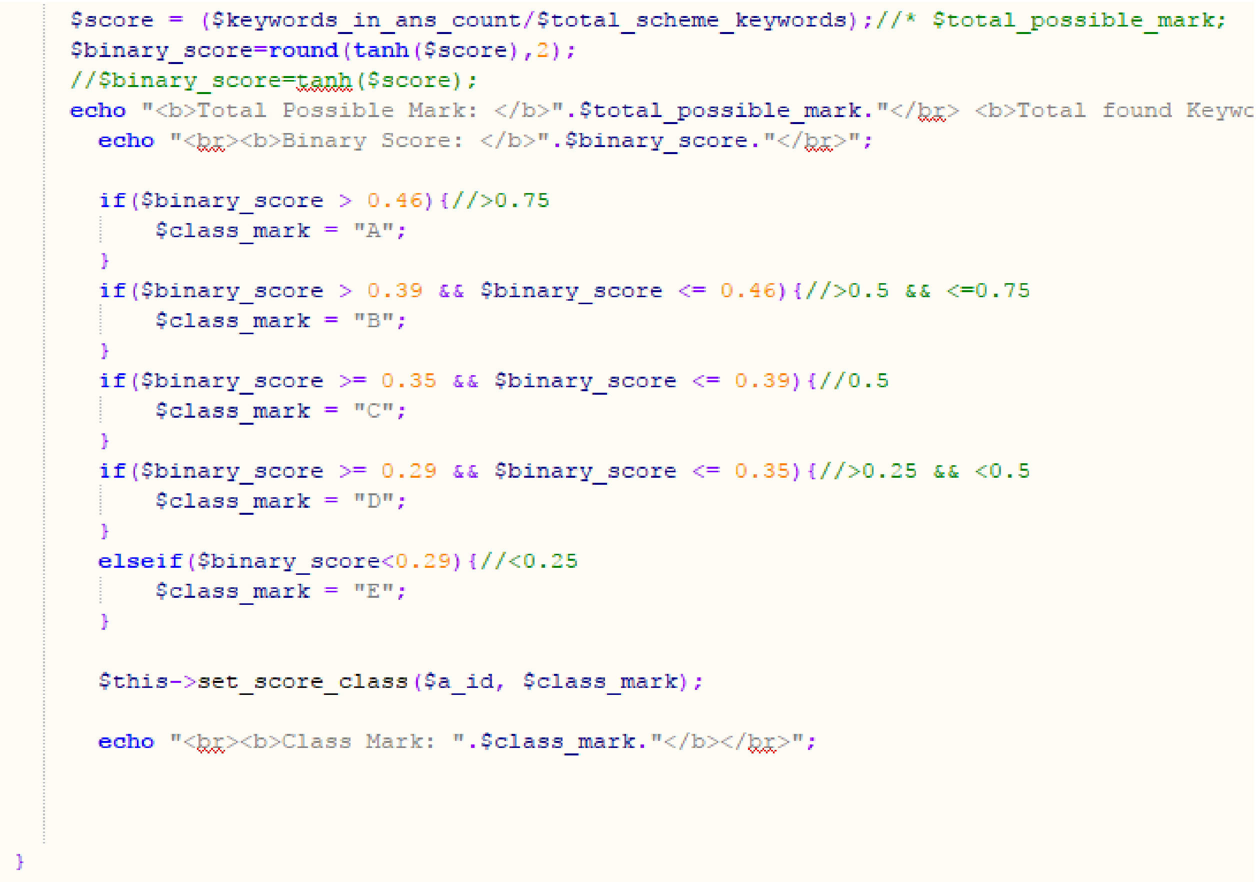


```
public function pos_tag($pos,$lemmatizer,$sentence){

    $result = $pos->tag(explode(' ', $sentence));
    $keywords_pos = array('NN', 'NNS', 'NNPS', 'JJ', 'JJR', 'JJS', 'RB', 'RBR', 'RBS', 'VB', 'VBD', 'VBG', 'VBN', 'VBZ');

    foreach($result as $key=>$value){
        foreach($value as $key2=>$value2){
            if(in_array($value2[1],$keywords_pos)){
                $lemmas = $lemmatizer->getOnlyLemmas($value2[0]);
                $keywords[]= $lemmas[0];
            }
        }
    }
    $keywords=array_values(array_unique($keywords));
    $keywords=array_filter($keywords, 'strlen');
    $this->_keywords = $keywords;
}

public function lemmatize_ans($ans,$lemmatizer){
    $prep_ans = strip_tags($ans);
    $sent_array=explode(' ',$prep_ans);
    foreach($sent_array as $word){
        $lemmas = $lemmatizer->getOnlyLemmas($word);
            $lemma_array[]= $lemmas[0];
    }
    $new_ans=implode(" ",$lemma_array);
    return $new_ans;
}
```

From keywords synonyms, hypernyms, hyponyms and *othernyms* are mined, compared, and counted for existence between the candidate and scheme answer. A mark is then awarded and returned by the score_answer() method as demonstrated in the next page.



```
public function score_answer($stud_ans,$prepared_scheme_answer,$a_id,$lemmatizer,$q_id){
    $score = 0;
    $keywords_in_ans_count = 0;

    $stud_ans_array = explode(" ",$stud_ans);
    $stud_ans_array = array_flip($stud_ans_array);
    $this->_cand_ans = $stud_ans_array;

    echo "</br><b>STUD ANS: ".$stud_ans."</b></br>";
    $total_scheme_keywords = count($prepared_scheme_answer);
    $total_possible_mark = $this->get_question_total_mark($q_id);

    foreach($prepared_scheme_answer as $key=>$value){
        $stud_ans_array = $this->candidate_answer();
        /******************************************keyword************************************************/
        if(isset($stud_ans_array[$value])){
            echo "Keyword Exists: ".$value."</br>";
            $keywords_in_ans_count+=1;
            unset($stud_ans_array[$value]);//unset already counted keyword
            $this->_cand_ans = $stud_ans_array;//unset already counted keyword
        }elseif(!isset($stud_ans_array[$value])){//if keyword doesn't exist
        /******************************************synonym************************************************/
            $keyword_synnate_exists = $this->half_mark($stud_ans_array,$value,$lemmatizer);
            if ($keyword_synnate_exists !== false){
                $keywords_in_ans_count+=1;
       /******************************************hypernym************************************************/
            }elseif($keyword_synnate_exists == false){
                $keyword_hypernym_exists = $this->athird_mark($stud_ans_array,$value);//check for hypernyms
                if ($keyword_hypernym_exists !== false){
                      $keywords_in_ans_count+=1;
       /******************************************hyponym************************************************/
                }elseif($keyword_hypernym_exists == false){
                      $keyword_hyponym_exists = $this->quater_mark($stud_ans_array,$value);//check for hyponyms
                      if ($keyword_hyponym_exists !== false){
                          $keywords_in_ans_count+=0.5;
       /******************************************others************************************************/
                      }elseif($keyword_hyponym_exists == false){
                          $keyword_othernym_exists = $this->afifth_mark($stud_ans_array,$value);//check for othernyms
                          if ($keyword_othernym_exists !== false){
                              $keywords_in_ans_count+=0.33;
                          }
                      }
                  }
              }
        }
    }
```



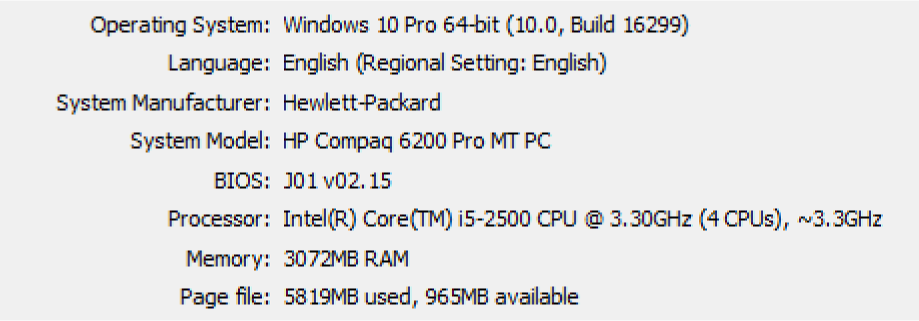
```
$score = ($keywords_in_ans_count/$total_scheme_keywords);//* $total_possible_mark;
$binary_score=round(tanh($score),2);
//$binary_score=tanh($score);
echo "<b>Total Possible Mark: </b>".$total_possible_mark."</br> <b>Total found Keywc
  echo "<br><b>Binary Score: </b>".$binary_score."</br>";

  if($binary_score > 0.46){//>0.75
      $class_mark = "A";
  }
  if($binary_score > 0.39 && $binary_score <= 0.46){//>0.5 && <=0.75
      $class_mark = "B";
  }
  if($binary_score >= 0.35 && $binary_score <= 0.39){//0.5
      $class_mark = "C";
  }
  if($binary_score >= 0.29 && $binary_score <= 0.35){//>0.25 && <0.5
      $class_mark = "D";
  }
  elseif($binary_score<0.29){//<0.25
      $class_mark = "E";
  }

  $this->set_score_class($a_id, $class_mark);

  echo "<br><b>Class Mark: ".$class_mark."</b></br>";


}
```

## 5.8. Experimentation

Experiments were done and inferences reached at every time a new answer and marking scheme were fed into the AES marker engine. The variables under scrutiny and testing were as proposed in chapter three i.e. independent variable: number of keywords and synonyms contained in student answer. Dependent variable: accuracy of the answer.

### 5.8.1. Experiment setup

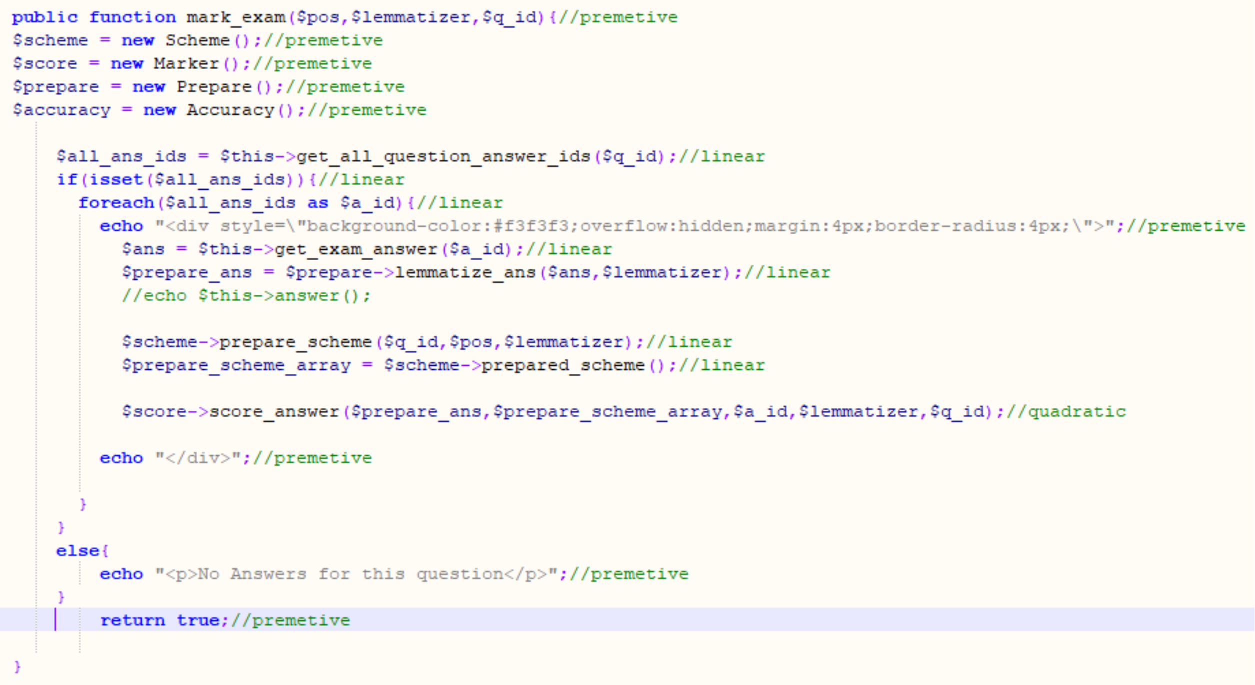
The experiment was carried out using the developed algorithm by the direct feeding of data collected. A total of 151 student answers provided by 80 students for 10 exam questions were fed into the AES system. The experiments were run in a computing environment consisting of windows O.S, Quad core CPU running at 3.3GHz and 4GB RAM as summarized in figure 5.8 below:



| | |
|---|---|
| Operating System: | Windows 10 Pro 64-bit (10.0, Build 16299) |
| Language: | English (Regional Setting: English) |
| System Manufacturer: | Hewlett-Packard |
| System Model: | HP Compaq 6200 Pro MT PC |
| BIOS: | J01 v02.15 |
| Processor: | Intel(R) Core(TM) i5-2500 CPU @ 3.30GHz (4 CPUs), ~3.3GHz |
| Memory: | 3072MB RAM |
| Page file: | 5819MB used, 965MB available |

**Figure 5. 4: Experimentation computing Environment Machine Specs**

The computing tools of implementation including MySQL relational database system and XAMPP web engine were used as discussed in section 5.5. The main AES engine written in PHP – 7 was able to mark and grade answers, the engine code is shown below:



```
public function mark_exam($pos,$lemmatizer,$q_id){//premetive
$scheme = new Scheme();//premetive
$score = new Marker();//premetive
$prepare = new Prepare();//premetive
$accuracy = new Accuracy();//premetive

    $all_ans_ids = $this->get_all_question_answer_ids($q_id);//linear
    if(isset($all_ans_ids)){//linear
      foreach($all_ans_ids as $a_id){//linear
        echo "<div style=\"background-color:#f3f3f3;overflow:hidden;margin:4px;border-radius:4px;\">";//premetive
          $ans = $this->get_exam_answer($a_id);//linear
          $prepare_ans = $prepare->lemmatize_ans($ans,$lemmatizer);//linear
          //echo $this->answer();

          $scheme->prepare_scheme($q_id,$pos,$lemmatizer);//linear
          $prepare_scheme_array = $scheme->prepared_scheme();//linear

          $score->score_answer($prepare_ans,$prepare_scheme_array,$a_id,$lemmatizer,$q_id);//quadratic

        echo "</div>";//premetive

      }
    }
    else{
        echo "<p>No Answers for this question</p>";//premetive
    }
        return true;//premetive

}
```

The model achieved an average marking accuracy of 89%.

# CHAPTER SIX: DISCUSSION OF RESULTS

## 6.1. Introduction

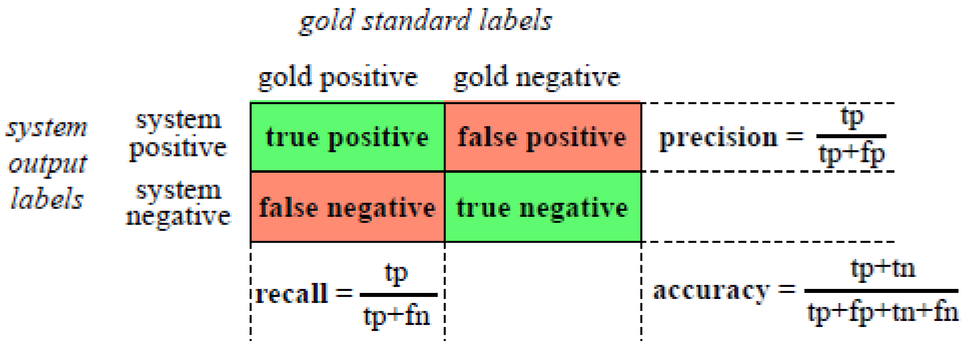
This research aimed at automating the exam marking and grading process to introduce efficiencies such as cutting down on time taken to mark and improving the accuracy of grading. This chapter represents the findings of the algorithm implementation. The model accuracy is discussed in conjunction with the algorithm efficiencies.

## 6.2. Model accuracy

A total of 151 answers were marked and graded by the artificial exam scorer implemented in the previous chapter. The data had first been cleaned to come up with golden labels that are considered the correctly expected grading as per the rating of a human marking expert. The model accuracy was calculated using the confusion matrix model and chi-square accuracy measure.

### 6.2.1. Automated Marking Algorithm Accuracy, Exactness and Sensitivity

Accuracy of an artificially intelligent system can be measured by constructing a confusion matrix an example adopted from Jurafsky & Martin (2017) is shown in figure 6.1. The system is scrutinized by checking how many data sets were classified correctly by the system against the inputs labeled correctly by a human expert also known as gold standards. Outputs classified as correct and were labeled correct are termed *true positives* while those labeled incorrect and marked incorrect are termed *true negative*.



*gold standard labels*

| | | gold positive | gold negative | |
|---|---|---|---|---|
| *system output labels* | system positive | 🟩 true positive | 🟥 false positive | precision = $\frac{tp}{tp+fp}$ |
| | system negative | 🟥 false negative | 🟩 true negative | |
| | | recall = $\frac{tp}{tp+fn}$ | | accuracy = $\frac{tp+tn}{tp+fp+tn+fn}$ |

**Figure 6. 1: General confusion matrix (Jurafsky & Martin, 2017)**

Accuracy of the classifier can be calculated using the formulae:

$$Accuracy = \frac{tp + tn}{tp + fp + tn + fn}$$

However, this measure is limited in the case of unbalanced frequency in different classes of data inputs thus it is not used precisely to measure system performance (Jurafsky & Martin, 2017). Two other measure are thus introduced: precision focusing on how many positives the system achieves and recall focusing on how many positives the human expert labeled. Precision measures exactness of model while recall measures completeness or sensitivity.

$$precision = \frac{tp}{tp + fp} \qquad recall = \frac{tp}{tp + fn}$$

Table 6.1 below shows the confusion matrix generated by the Artificial Exam marker for 151 answer samples. From the table accuracy of 0.76158940397351 was calculated, precision and recall were calculated as:

$$Accuracy \; = \frac{42 + 10 + 7 + 1 + 55}{151} = 76.15\%$$

**Table 6. 1: Precision & Recall Distribution**

| | | | **human expert gold labels** | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | **A** | **B** | **C** | **D** | **E** | ***Precision*** |
| system | | **A** | 42 | 0 | 0 | 0 | 1 | 0.976744 |
| outputs | | **B** | 5 | 10 | 0 | 0 | 0 | 0.666667 |
| Artificial | | **C** | 9 | 2 | 7 | 7 | 5 | 0.233333 |
| Marker | | **D** | 0 | 0 | 0 | 1 | 1 | 0.5 |
| | | **E** | 1 | 2 | 1 | 2 | 55 | 0.901639 |
| | | ***Recall*** | 0.736842 | 0.714286 | 0.875 | 0.1 | 0.887097 | |
| | | | | | | | | |
| | | **micro averaged precision** | | | **0.655677** | | | |
| | | **micro averaged recall** | | | **0.662645** | | | |

$$\therefore \mathit{microaveraged\ precision} = \frac{0.9 + 0.66 + 0.23 + 0.5 + 0.9}{5} = \ 0.65$$

$$\mathit{microaveraged\ recall} = \frac{0.7 + 0.7 + 0.8 + 0.1 + 0.8}{5} = \ 0.66$$

This means the artificial marker algorithm marks with 65% exactness with 66% sensitivity. The balance between precision and recall is calculated through the F-measure.

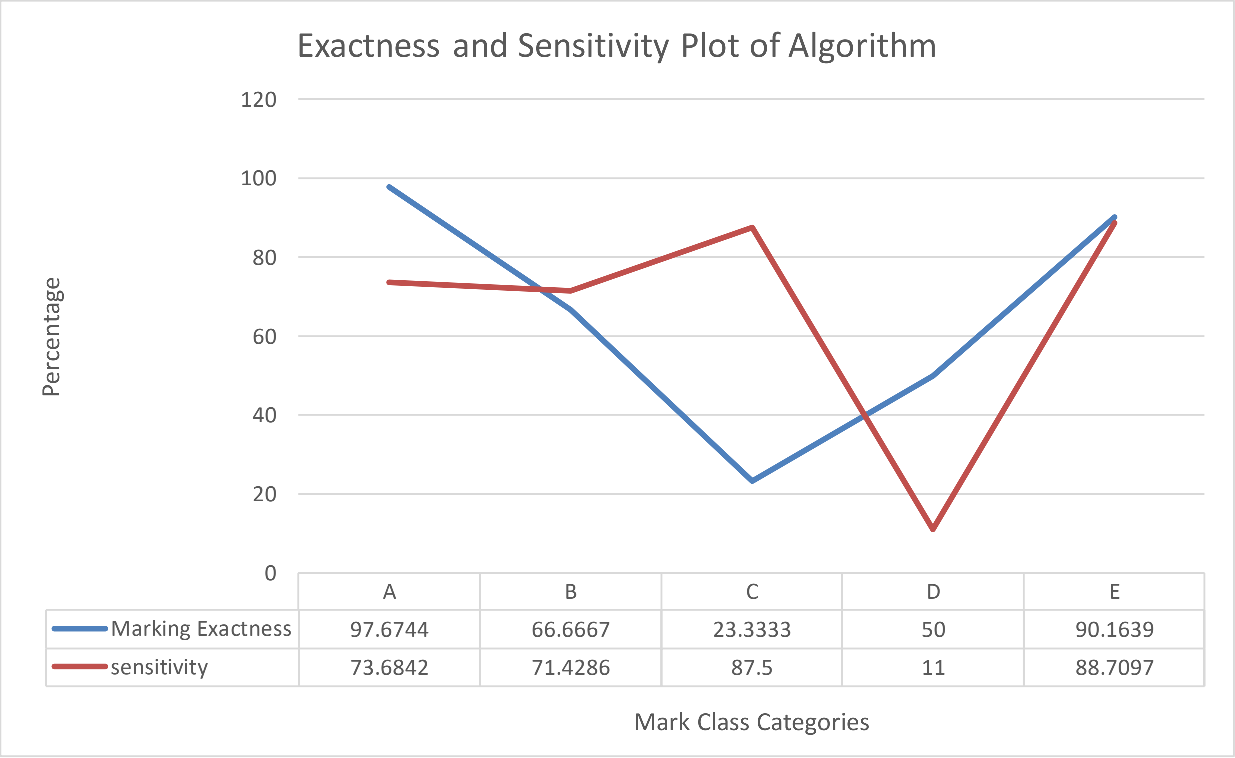
$$F = \frac{2PR}{P + R}$$

Thus F-measure for the marking algorithm is:

$$= \frac{2 * 0.65 * 0.66}{0.65 + 0.66}$$

$$= \ 0.6549$$

Meaning there is a 65% balance between algorithm exactness and sensitivity.



|  | A | B | C | D | E |
|---|---|---|---|---|---|
| Marking Exactness | 97.6744 | 66.6667 | 23.3333 | 50 | 90.1639 |
| sensitivity | 73.6842 | 71.4286 | 87.5 | 11 | 88.7097 |

**Figure 6. 2: Sensitivity & Marking exactness inverse relationship**

### 6.2.2. Chi-square measure

Chi -square measures accuracy based on observed verses expected in relation to a population and its sample (Jaynes,1995). In this study the null hypothesis stated the claim that marking is equal to or greater than 90% when keywords and related word relationship between student answer and marking scheme are considered i.e. $H_0\ Accuracy\ \geq 90\%$. The chi-square formula was used to calculate a probability value that tests the null hypothesis claim. The formula is:

$$x_c^2 = \sum \frac{(O_i\ -\ E_i)^2}{E_i}$$

Where "$c$" represents the degree of freedom i.e. $n - 1$ where $n$ is the number of categorical data, $x$ is the chi score from where a p-value is calculated while $O$ and $E$ represent the Observed and the Expected respectively. The calculated value obtains probability from a chi-square distribution table. This study observed 151 student answers categorized as short essay answers with marks between 1 & 4. The chi-square p-value was calculated as:

| **CATEGORY of answers per total mark** | **Frequency (No. Of sampled answers)** | **observed** | **Expected (90% of Frequency)** | $\boldsymbol{O\ -\ E}$ | $\boldsymbol{(O - E)^\wedge 2}$ | $\boldsymbol{((O - E)^\wedge 2)/E}$ |
|---:|---:|---:|---:|---:|---:|---:|
| **1** | 54 | 54 | 49 | 5 | 25 | **0.510204** |
| **2** | 39 | 29 | 36 | -7 | 49 | **1.361111** |
| **3** | 21 | 18 | 19 | -1 | 1 | **0.052632** |
| **4** | 37 | 22 | 34 | -12 | 144 | **4.235294** |
| | | | | | | **6.159241** |

From the chi-square probability distribution table, the chi-score $6.159241$ gives a p-value of $0.104117$ thus the algorithm is $1 - 0.104117\ =\ 0.895883$ meaning the model is $89.5883\%$ accurate. This value is slightly lower than the null hypothesis $90\%$ thus leading to a **rejection** of the **null hypothesis**.

## 6.3. Algorithm efficiency

Algorithm efficiency measure the computational expense of running an algorithm. Computational expense is understood as the algorithm demand for computational resources understood as processing time and space during execution. Algorithm efficiency can be calculated using the big O analysis (Clifford, 2010).

### 6.3.1. Automated Marking Algorithm Runtime Analysis

In $O(n)$ runtime analysis there are seven primitive functions defining algorithm complexity. They include the constant, logarithm, Linear, $n \log n$, Quadratic, Cubic and Exponential functions summarized in figure 6.3 below for every successive growth of $input\ n$:

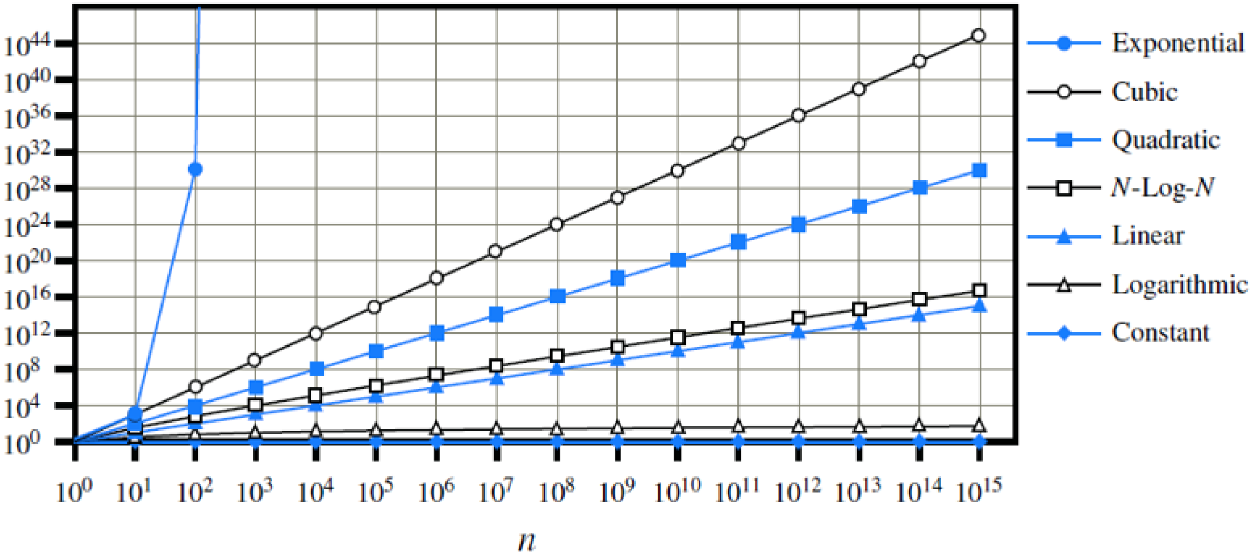
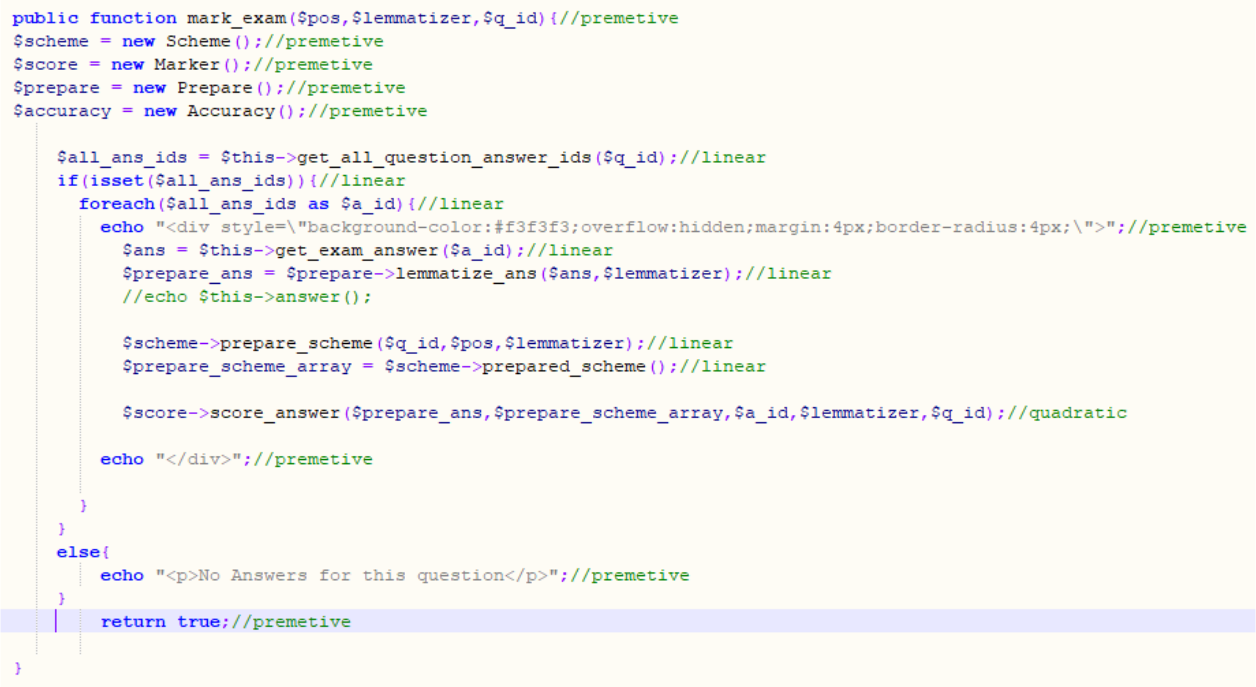


**Figure 6. 3: Big O runtime Analysis seven functions (Clifford, 2010)**

An algorithm is understood as a function $f(n)$ that takes in $n\ inputs$ processes them and returns an output. Primitive operations are not bothered in Big O analysis. These include the computational cost of assigning a value to a variable, object referencing, arithmetic processing, comparisons, element accessing in a simple data structure, method calling or returning method value (Clifford, 2010).

The automated marker algorithm runtime analysis yielded a quadratic runtime algorithm as per the line by line analysis shown in figure 6.4 of the main marking function.



```
public function mark_exam($pos,$lemmatizer,$q_id){//premetive
$scheme = new Scheme();//premetive
$score = new Marker();//premetive
$prepare = new Prepare();//premetive
$accuracy = new Accuracy();//premetive

    $all_ans_ids = $this->get_all_question_answer_ids($q_id);//linear
    if(isset($all_ans_ids)){//linear
      foreach($all_ans_ids as $a_id){//linear
        echo "<div style=\"background-color:#f3f3f3;overflow:hidden;margin:4px;border-radius:4px;\">";//premetive
          $ans = $this->get_exam_answer($a_id);//linear
          $prepare_ans = $prepare->lemmatize_ans($ans,$lemmatizer);//linear
          //echo $this->answer();

          $scheme->prepare_scheme($q_id,$pos,$lemmatizer);//linear
          $prepare_scheme_array = $scheme->prepared_scheme();//linear

          $score->score_answer($prepare_ans,$prepare_scheme_array,$a_id,$lemmatizer,$q_id);//quadratic

        echo "</div>";//premetive

      }
    }
    else{
        echo "<p>No Answers for this question</p>";//premetive
    }
        return true;//premetive

}
```

**Figure 6. 4: Big O runtime analysis for automated marking algorithm**

# CHAPTER SEVEN: CONCLUSION AND RECOMMENDATIONS

Automated Exam Scoring has been a matter of interest ever since the rise of Artificial Intelligence and Computational Linguistics. Various algorithms have been developed in an aim to achieve a process traditionally done by human expert instructors and examiners. These algorithms have always been improving in accuracy as more research has been published. However, the currently achieve marking accuracy levels are still lower to allow replacing human markers with a software given the sensitive nature of academic exams. This study aimed at automating the exam marking and grading process to introduce efficiencies such as cutting down on time and cost expenses and achieving high levels of accuracy at the same time.

Key contribution made by this study into Automated Exam Scoring is the consideration of synonyms as an extension to the keyword approach currently achieving the highest accuracy level. The developed algorithm was able to improve the keyword marking methodology researched by (Frost, 2008) by a margin of 16% from 73% to 89%. The model achieved more accuracy when grading lower mark answers achieving 99.9% when marking 1-mark answers. However, the null hypothesis claiming a 90% accuracy was rejected. In future research studies this accuracy can be raised by overcoming the limitation of keyword approach to marking. This approach fails to capture semantic relations by only focusing on words ignoring a whole answer statement. The hybrid approach combining keyword approach, synonyms approach, semantic parsing approach as well as machine learning approach can achieve percentages higher than the realized.

There was a limit in the dataset used. Only short essay answers were considered. Future research should focus on answers possessing more mark distribution. In addition, all marked answers were obtained from one subject i.e. computer science thus a more diverse answer base can shade more light on automated marking especially on the concept of context and pragmatics. The lexical dictionary used also had limits. WordNet lexical dictionary captures more general semantic-lexical relationships thus performs relatively poor in domain specific text-mining.

There is also need for a wide-range of researchers to be involved in the algorithm fine tuning. Since development requires a human expert to gold label the marking scheme, this research was limited to only one particular field where the researcher is an expert.

# References

Abhimanyu, C., Abhinav, P., & Chandresh, S. (2013). Natural Language Processing. *INTERNATIONAL JOURNAL OF TECHNOLOGY ENHANCEMENTS AND EMERGING ENGINEERING RESEARCH*, 1-4.

Alastair, P., Gill, E., & Ayesha, A. (2004). Let's Stop Marking Exams. *IAEA* (p. 21). Cambridge: University Of Cambridge.

*Automated Scoring: Automated Language Assessment*. (2017). Retrieved , July 15; 2017, from http://www.pearsonassessments.com/automatedlanguageassessment/ourtechnology/automatedscoring.html

*Budget Highlights 2016/2017.* (2016). Nairobi: Mwananchi publishers.

Burstein, J., Kukich, K., Wolff, S., Lu, C., & Chodorow, M. (1998). Computer Analysis of Essays. *Proceedings of the NCME Symposium on Automated Scoring.* Montreal.

Clifford A. S. (2010). *A Practical Introduction to Data Structures and Algorithm Analysis* (3rd ed.). Blacksburg: Virginia.

Charu, C. A., & Cheng, X. Z. (2012). *Mining Text Data.* New York: Springer Science+Business Media LLC.

Chemweno, B. (2016, March 4). Examination cheating in Kenya goes up by a record 70 per cent. *The Standard*.

Commission for University Education. (2017). *Press statement on quality audit of universities in Kenya conducted in January and February .* Nairobi: Government Press.

Deerwester, S. C., Dumais, S. T., Furnas, G., & Harshman, R. (1998). *Computer information retrieval using latent semantic structure.* Us patent 4,839,853.

Dikli, S. (2006). Automated Essay Scoring. *Turkish Online Journal of Distance Education-TOJDE*, 49-62.

Elsevier. (2006). WordNet(s). In C. Fellbaum, *Encyclopedia of Language and Linguistics* (pp. 665-670). Cambridge MA: MIT Press.

Frost, J. (2008). *Automated Marking of Exam Papers.* Oxford: Oxford University Press.

Gari, A. (2017, January 1). KCSE 2016: Candidate disputes 'B' grade, says more time needed to mark exams. *The Star*, pp. 2-3.

George, O. (2011). Higher education quality in Kenya: a critical reflection of key challenges. *Quality in Higher Education*, 299-315.

International Business Machines. (2017). *Ibm Test Score History: Automated Test Scoring*. Retrieved June 20; 2017, from http://www-03.ibm.com/ibm/history/ibm100/us/en/icons/testscore/

Jaynes, E. T. (1995). *Probability Theory: The Logic of Science.* Washington DC: Washington University.

Jurafsky, D., & James, H. M. (2017). *Speech and Language Processing: An Introduction to Natural Language Processing, 3 Ed.* Stanford University Press.

Kiptanui, D. R., Cheruto, L. K., & Kimutai, D. R. (2011). Student Factors Influencing Cheating in Undergraduate Examinations in Universities in Kenya. *Problems of Management in the 21st Century*, 173-181.

Kenya National Bureau of Statistics. (2015). *Kenya Facts and Figures.* Nairobi: Government Press.

Kenya National Examinations Council. (2016). *Criteria for elligiblility to be a marking center.* Nairobi.

Kumar, E. (2011). *Natural Language Processing.* New-Delhi: I.K International Publishishing House.

Landauer, T. K., Foltz, P. W., & Laham, D. (1998). Introduction to Latent Semantic Analysis. *Discourse Processes*, 259-284.

Kenya Law. (2008). *Kenyan Constitution.* Nairobi: Government Press.

Linden, W. J. (2010). Item Response Theory. *International Encyclopedia of Education*, 81-88.

Mahwah, N. J., & Lawrence E. A. (2003). *Automated essay scoring: A cross-disciplinary perspective.* New Jersey: Lawrence Erlbaum Associates Inc.

Marcus, M. P., Santorini, B., & Marcinkiewicz, M. A. (1993). Building a large annotated corpus of English: The Penn treebank. *Computational Linguistics*, 313-330.

Michael, W. K., & Jacob B. (2010). *Text Mining - Applications and Theory.* Cornwall: John Wiley & Sons, Incorporated.

Miller, G. A. (1995). WordNet: A Lexical Database for English. *Communications of the ACM Vol. 38, No. 11: 39-41*, 1-3.

Ministry of Education Science & Technology. (2014). *Education Task Force Report.* Nairobi: Government Press.

Ministry of Education Science & Technology. (2015). *Media Statement on the Kenya National Examinations Council by CS Dr. Fred Matiangi.* Nairobi: Government Press.

Ministry of Planning. (2008). *Vision 2030 Abridged.* Nairobi: Government Press.

Pearson. (2017). *Automated Language Assessment: Pearsons Assessments*. June 18; 2017, Retrieved from http://www.pearsonassessments.com/automatedlanguageassessment.html

Ratcliffe, R. (2014, September 23). Teachers' tales of exam-marking issues: 'this is only going to get worse. *theGaurdian*.

Sheffield, T. U. (2000). *The Institute for Language Speech and Hearing: Grady Ward's Moby.* Retrieved October 24; 2017, from http://icon.shef.ac.uk/Moby/

Stuart, J. R., Norvig, P. (2010). *Artificial Intelligence A Modern Approach Third Edition.* New Jersey: Prentice Hall.

Vasagar, J. (2012, May 17). Examiners axed after marking mistakes. *theGaurdian*.

Weiss, M. A. (2014). *Data Structures and Algorithm Analysis in C++.* Florida : Pearson.

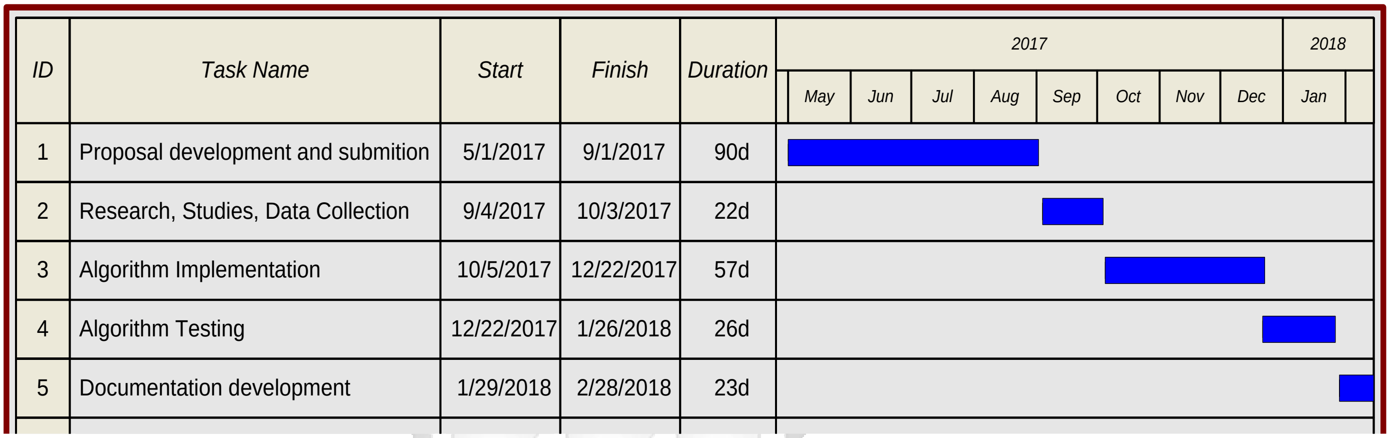
Write to Learn. (2017). *The Research Behind WriteToLearn: Building Student Summarization, Writing and Reading Comprehension Skills With Guided Practice and Automated Feedback.* Pearson.

Xinming, A., & Yiu, F. Y. (2014). *Item Response Theory: What It Is and How You Can Use the IRT.* Cary: SAS Inc.

VT OMNES
VNVM SINT

# Appendix A: Gantt chart

This entire project took place within the period demonstrated below by the use of a Gantt chart.



| ID | Task Name | Start | Finish | Duration |
|---|---|---|---|---|
| 1 | Proposal development and submition | 5/1/2017 | 9/1/2017 | 90d |
| 2 | Research, Studies, Data Collection | 9/4/2017 | 10/3/2017 | 22d |
| 3 | Algorithm Implementation | 10/5/2017 | 12/22/2017 | 57d |
| 4 | Algorithm Testing | 12/22/2017 | 1/26/2018 | 26d |
| 5 | Documentation development | 1/29/2018 | 2/28/2018 | 23d |

Figure A. 1 Gantt chart

## Appendix B: Chi – Square distribution

| d.f. | .995 | .99 | .975 | .95 | .9 | .1 | .05 | .025 | .01 |
|---:|---:|---:|---:|---:|---:|---:|---:|---:|---:|
| 1 | 0.00 | 0.00 | 0.00 | 0.00 | 0.02 | 2.71 | 3.84 | 5.02 | 6.63 |
| 2 | 0.01 | 0.02 | 0.05 | 0.10 | 0.21 | 4.61 | 5.99 | 7.38 | 9.21 |
| 3 | 0.07 | 0.11 | 0.22 | 0.35 | 0.58 | 6.25 | 7.81 | 9.35 | 11.34 |
| 4 | 0.21 | 0.30 | 0.48 | 0.71 | 1.06 | 7.78 | 9.49 | 11.14 | 13.28 |
| 5 | 0.41 | 0.55 | 0.83 | 1.15 | 1.61 | 9.24 | 11.07 | 12.83 | 15.09 |
| 6 | 0.68 | 0.87 | 1.24 | 1.64 | 2.20 | 10.64 | 12.59 | 14.45 | 16.81 |
| 7 | 0.99 | 1.24 | 1.69 | 2.17 | 2.83 | 12.02 | 14.07 | 16.01 | 18.48 |
| 8 | 1.34 | 1.65 | 2.18 | 2.73 | 3.49 | 13.36 | 15.51 | 17.53 | 20.09 |
| 9 | 1.73 | 2.09 | 2.70 | 3.33 | 4.17 | 14.68 | 16.92 | 19.02 | 21.67 |
| 10 | 2.16 | 2.56 | 3.25 | 3.94 | 4.87 | 15.99 | 18.31 | 20.48 | 23.21 |
| 11 | 2.60 | 3.05 | 3.82 | 4.57 | 5.58 | 17.28 | 19.68 | 21.92 | 24.72 |
| 12 | 3.07 | 3.57 | 4.40 | 5.23 | 6.30 | 18.55 | 21.03 | 23.34 | 26.22 |
| 13 | 3.57 | 4.11 | 5.01 | 5.89 | 7.04 | 19.81 | 22.36 | 24.74 | 27.69 |
| 14 | 4.07 | 4.66 | 5.63 | 6.57 | 7.79 | 21.06 | 23.68 | 26.12 | 29.14 |
| 15 | 4.60 | 5.23 | 6.26 | 7.26 | 8.55 | 22.31 | 25.00 | 27.49 | 30.58 |
| 16 | 5.14 | 5.81 | 6.91 | 7.96 | 9.31 | 23.54 | 26.30 | 28.85 | 32.00 |
| 17 | 5.70 | 6.41 | 7.56 | 8.67 | 10.09 | 24.77 | 27.59 | 30.19 | 33.41 |
| 18 | 6.26 | 7.01 | 8.23 | 9.39 | 10.86 | 25.99 | 28.87 | 31.53 | 34.81 |
| 19 | 6.84 | 7.63 | 8.91 | 10.12 | 11.65 | 27.20 | 30.14 | 32.85 | 36.19 |
| 20 | 7.43 | 8.26 | 9.59 | 10.85 | 12.44 | 28.41 | 31.41 | 34.17 | 37.57 |
| 22 | 8.64 | 9.54 | 10.98 | 12.34 | 14.04 | 30.81 | 33.92 | 36.78 | 40.29 |
| 24 | 9.89 | 10.86 | 12.40 | 13.85 | 15.66 | 33.20 | 36.42 | 39.36 | 42.98 |

# Appendix C: Sample Exam Marking Scheme

**1.** *John van Neumann* contributed majorly to the design and architecture of the modern computer.

a. With the use of a suitable diagram, show the **5 components** of the *van Neumann* architecture and how they interrelate. **(5 marks)**

**Award 5 marks for any diagram showing the below interrelationships**

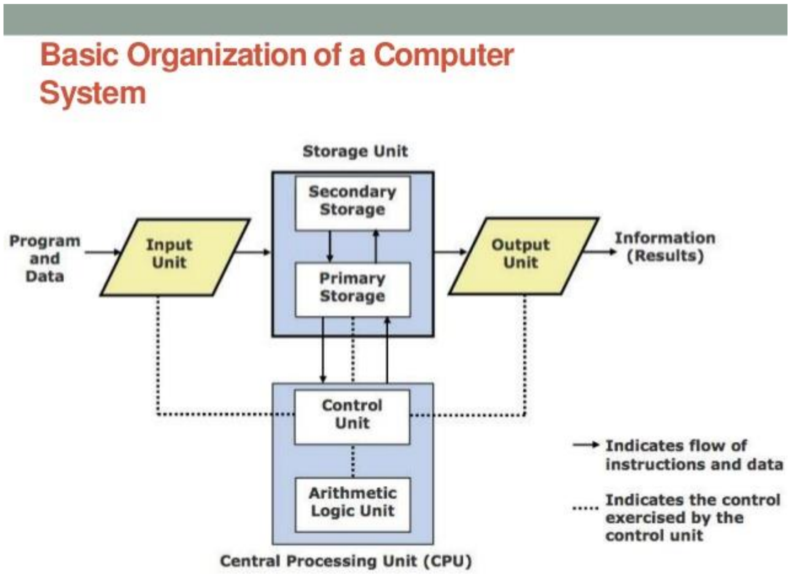

Basic Organization of a Computer System
Storage Unit
Secondary Storage
Primary Storage
Program and Data
Input Unit
Output Unit
Information (Results)
Control Unit
Arithmetic Logic Unit
Central Processing Unit (CPU)
Indicates flow of instructions and data
Indicates the control exercised by the control unit

**Figure C. 1: Van Neumann Model**

b. Clearly explain the functionality of each of the **5 components** from your diagram above. **(5 marks)**

**Award 1 mark for correct point out of a component and 1 mark for explanation**

**Inputting:** The processes of entering data and instructions into the computer

**Storing:** Saving data and instructions making them available for processing and output

**Processing:** Performing *arithmetic operations* **(addition, multiplication, subtraction and division)** or *logical operations* **(OR, AND, NAND & NOR)** on data to transform them to information

**Outputting:** Producing useful information in a desired form such as audio, video or printed copies

**Control:** Directing/coordinating the manner in which all the above operations are carried out

**2.** A computer scientist doing an experiment on digital logic passed two signals; **on** & **off** signal through an **XOR gate**. After obtaining the result, he included another **on** signal with the *first result* and passed the two signals through an **NAND (NOT AND)** gate.

i. **Construct** truth tables for **XOR, AND** & **NAND** logical operations using **1 to represent *true*** & **0 to represent *false*. (3 marks)**

**Award 1 mark for each correct column in the table**

**Table C. 1: Truth Tables**

| INPUTS (Data) | | OUTPUT (Information) | | | | | |
|---|---|---|---|---|---|---|---|
| A | B | OR | NOR | AND | NAND | XOR | NXOR |
| 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |

ii. Using your above truth tables, **evaluate** the final results of the scientists' experiment. Use **1 to represent *on*** & **0 to represent *off*. (2 marks)**

**Award 1 mark for each stage of calculation**

$$1 \oplus 0 = 1$$

$$\neg (1 \wedge 1) = 0$$

## Appendix D: Scaled Sigmoid Activation Function: hyperbolic tangent

Sigmoid function has many advantages over the step function. At first the activation is analogue in nature as opposed to the digital step function whose output is bound at $(0,1)$

$$A = \frac{1}{1+e^{-x}}$$

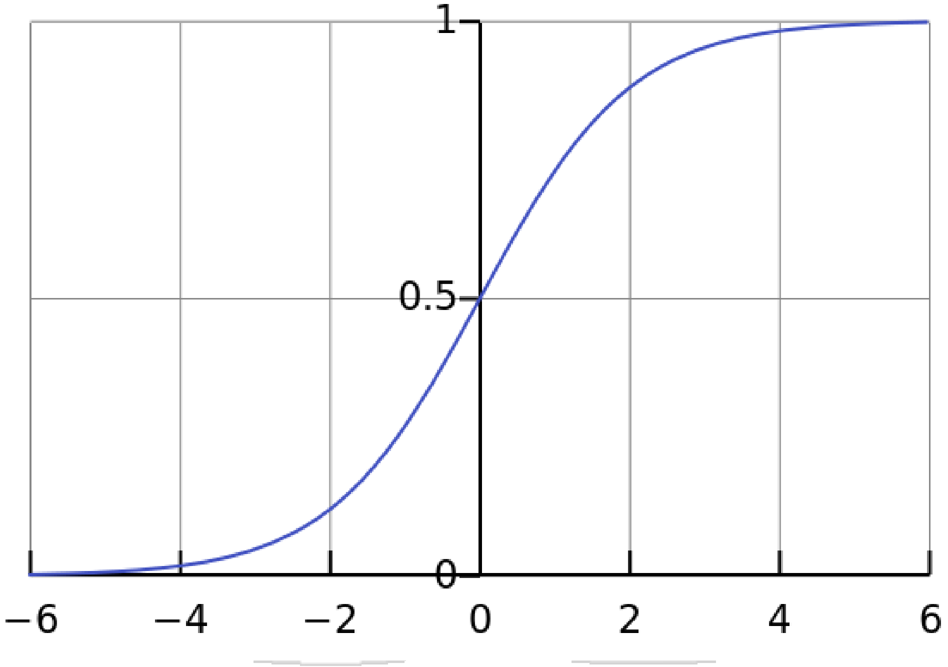
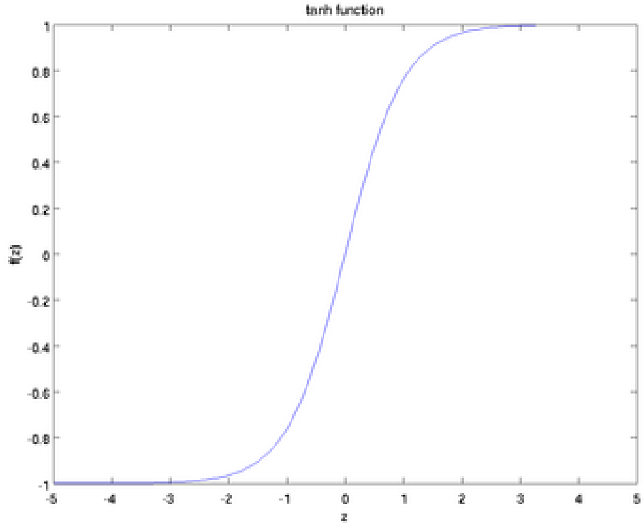


**Figure D. 1: Sigmoid Activation Function**

Activation $\boldsymbol{A}$ is bound between $(0,1)$ unlike the step function whose activation ranges between $(-\infty, +\infty)$

However, sigmoid function suffers the problem of vanishing gradients towards its tail on both the high-end and lower-end.

The hyperbolic $tanh$ overcomes this limit. It is a scaled sigmoid function with strong gradients than for sigmoid. It ranges $(-1,1)$.



$$f(x) = tanh(x) = \frac{2}{1+e^{-2x}} - 1$$

**Figure D. 2: Hyperbolic Tangent Activation Function**