



Electronic Theses and Dissertations

2020

A Machine learning model for task allocation.

Kibiwot, Simon Kipkemboi
Faculty of Information Technology
Strathmore University

Recommended Citation

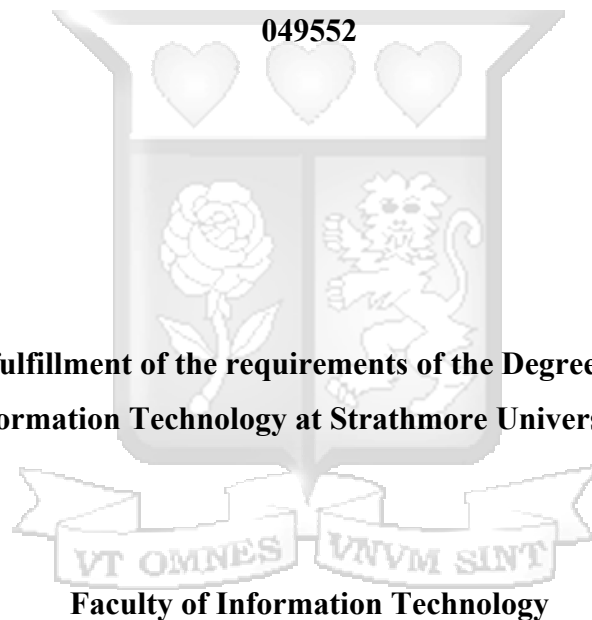
Kibiwot, S. K. (2020). *A Machine learning model for task allocation*. [Thesis, Strathmore University].

<http://hdl.handle.net/11071/12051>

Follow this and additional works at: <http://hdl.handle.net/11071/12051>

A Machine Learning Model for Task Allocation

KIBIWOT SIMON KIPKEMBOI



**Submitted in partial fulfillment of the requirements of the Degree of Master of Science in
Information Technology at Strathmore University.**

Strathmore University

Nairobi, Kenya

June, 2020

This thesis is available for library use on the understanding that it is copyright material and that no quotation from the thesis may be published without proper acknowledgement.

Declaration

I declare that this work has not been previously submitted and approved for the award of a degree by this or any other University. To the best of my knowledge and belief, the research proposal contains no material previously published or written by another person except where due reference is made in the research itself.

© No part of this thesis may be reproduced without the permission of the author and Strathmore University.

Kibiwot Simon Kipkemboi - 049552

Sign: 

Date: 7th July 2020



The thesis of Kibiwot Simon Kipkemboi was reviewed and approved by the following:

Dr. Bernard Shibwabo, PhD
Senior Lecturer,
Faculty of Information Technology,
Strathmore University

Sign: 

Date: 7th July 2020

Abstract

Task allocation is one among the key planning exercises that plays a major role in an organization's quest to satisfy Service Level Agreements and to attain operational excellence. Task allocation is a difficult issue that addresses the inter-dynamics of tasks and employees, having in mind factors such as skills utilization, fairness and diversity. Conflicts arise when tasks take too long to be resolved due to in expertise and poor task allocation. The individuals assigned end up not performing the task well and incomplete tasks/project are presented. The end user is usually dissatisfied and will end up giving negative feedback. Previous studies have not addressed the issue of matching two or more employees to the task with qualities possessed by the employee, preference of the team or partner given and the qualifications for the task given. The nature of this problem has within this research been equated to the stable marriage problem solved by the Gale and Shapley's Algorithm. However, a serious concern of the Gale-Shapley algorithm is its non-truthfulness also known as the man optimal result. Gale-Shapley's algorithm was used to bring out the human aspect during task allocation. The aim of this research is to formulate an algorithm to support task allocation problem using machine learning. To ensure time efficiency, a predictive machine learning algorithm (artificial neural network) has been used to show improved time with each pair towards a task. A mixed method research methodology was used whereby it combined elements of quantitative and qualitative research approaches. The precision of the developed model was at 0.713421 bringing the estimate of the regression model to 71.34%. The findings showed that the neural network can be used as an effective algorithm for predicting the task allocation.

Keywords: Task allocation, Stable Marriage Problem, Gale and Shapley's Algorithm, Matching Problem, Man optimal result, regression model.

Dedication

First and foremost, I dedicate this research to the Almighty God who is the source of wisdom and skills to accomplish this study. I also want to dedicate this to my wife and my family for their prayers, heartfelt support and encouragement.



Acknowledgement

Foremost I would like to express my sincere gratitude to my supervisor Dr. Bernard Shibwabo whose knowledge, support and guidance helped me to complete the research in time.

I would also like to appreciate my colleagues; Thomas and Richard and friends who advised and helped in the course of my research project. Their efforts made the project a success. Much appreciation goes to my friend Mr. Ochieng for his mentorship.

Finally, I highly appreciate the support of my manager Mr. Charles Kiilur for easing my work to enable me complete my research.



Table of Contents

Declaration.....	ii
Abstract.....	iii
List of Figures.....	ix
List of Tables	x
Abbreviations/Acronyms.....	xi
Chapter 1: Introduction	1
1.1 Background	1
1.2 Problem Statement	3
1.3 Aim.....	3
1.4 Research Objectives	3
1.5 Research Questions	3
1.6 Justification	4
1.7 Scope and Limitation	4
Chapter 2: Literature Review.....	5
2.1 Introduction	5
2.2 Challenges in Task Allocation	5
2.2.1 Maximizing Group Cohesion.....	5
2.2.2 Allocating the Right Human Resources to Multi-Project Environments.....	6
2.2.3 Time saving by Efficient Allocation of Professionals	7
2.2.4 Human Resource Allocation to Tasks for Project Cost and Time Saving.....	8
2.2.5 Task identification and Assigning Suitable Employees to Best Fitted Tasks.....	9
2.3 Current Approaches in Task Allocation.....	10
2.3.1 Software Architecture and Social Network Human Resource Allocation Method	10
2.3.2 Multidimensional model approach for the deployment of human resources.....	10
2.3.3 An Intelligent Tool for Dynamic Task Allocation.....	11
2.3.4 Human Task Management Reference Model	13
2.3.5 Task allocation in production systems.....	13

2.4 Variations for the Gale and Shapley's Algorithm	15
2.4.1 An equal approach to the stable marriage problem	15
2.4.2 Rotation-Based Formulation for Stable Matching.....	16
2.5 Alternative Algorithms for solving Task Allocation.....	17
2.5.1 Dynamic Task Allocation Algorithm for Hiring Workers that Learn	18
2.5.2 Solving Task Allocation to the Worker Using Genetic Algorithm.....	19
2.5.3 Worker Friend Relationship model for task allocation.....	20
2.6 Review of Artificial Neural Network.....	22
2.6.1 Justification for Artificial Neural Network	22
2.7 Conceptual Framework	23
2.6.1 Partner Selection.....	24
2.6.2 Task Allocation.....	24
Chapter 3: Research Methodology.....	26
3.1 Introduction	26
3.2 Research Design.....	26
3.2.1 Location of the study	26
3.2.3 Data Collection and Procedure	27
3.3 Model Development.....	27
3.4 System Development Methodology.....	28
3.5 System Design.....	28
3.6 Research Quality	29
3.7 Ethical Considerations.....	29
Chapter 4: System Analysis and Design	31
4.1. Introduction	31
4.2. Requirements Analysis.....	31
4.2.1. Functional Requirements.....	31
4.2.2. Non-Functional Requirements.....	31
4.3. System Architecture	32
4.4. Use Case Diagram.....	32
4.5 System Sequence Diagram.....	35

4.6 Class Diagram	35
4.7 Data Flow Diagram	36
4.8 Database Schema.....	37
Chapter 5: Implementation and Testing.....	39
5.1. Introduction	39
5.2 Model Components	39
5.2.1 Data Extraction Cleaning and Merging Component	39
5.2.1 System Components	43
5.2.3 Neural Network Components	47
5.3 Model Implementation	47
5.4 Software Flow	49
5.5 Model Architecture	50
Chapter 6: Discussion	54
6.1 Introduction	54
6.2 Model Validation.....	54
6.2.1 Precision Results.....	54
6.3 Model Implementation Outputs	54
6.4 Contribution to Research.....	55
6.5 Shortfalls of the Model.....	55
Chapter 7: Conclusions and Recommendations	56
7.1 Conclusion.....	56
7.2 Recommendations	57
7.3 Suggestions for Research	57
References.....	58
Appendices.....	64

List of Figures

Figure 2.1: Collabcrew System Architecture Outline (Samath, et al., 2017)	12
Figure 2.2: Concept Model, Further Developed from DYNAMO++ (Fasthe & Stahre, 2010) ...	14
Figure 2.3: Layers of the Artificial Neural Network	22
Figure 2.4: Conceptual Framework for A Machine Learning Model for Task Allocation.....	24
Figure 3.1: System Prototyping Development Methodology (Alan, et al., 2012)	28
Figure 4.1: Use Case Diagram	33
Figure 4.2: System Sequence Diagram.....	35
Figure 4.3: Class Diagram	36
Figure 4.4: Data Flow Diagram	37
Figure 4.5: Database schema	38
Figure 5.1: HR data Extraction Screenshot.....	40
Figure 5.3: Extraction Removing Repeated Rows.....	41
Figure 5.4: Complete Extraction.....	42
Figure 5.5: RT and HR datasets Joining.....	42
Figure 5.6: Merged HR and RT datasets in CSV.....	43
Figure 5.7: List of task categories.....	44
Figure 5.8: List of employees who can perform the task.....	45
Figure 5.9: Selection of preferred employee.....	46
Figure 5.10: Selection workings	46
Figure 5.11: Output best pairs with time taken.....	46
Figure 5.12: The Proposed Model Architecture	51

List of Tables

Table 4.1: Partner Selection.....	30
Table 4.2: Ranking the matches and Calculating Time Taken.....	31
Table 4.3: Initiating Task Allocation.....	31
Table 5.1: Dataset Description.....	45
Table 5.2: Model and System Testing.....	49
Table 5.3: Model and System Test Results.....	49
Table 5.4: User Acceptance Testing.....	50
Table 6.1: Precision Comparison Table.....	54



Abbreviations/Acronyms

ANN	-	Artificial Neural Network
CP	-	Constraint Programming
DNA	-	Deoxyribonucleic Acid
EEG	-	Electroencephalogram
ESMP	-	Equitable Stable Marriage Problem.
GA	-	Genetic Algorithm
HRIS	-	Human Resource Information System
FRS-MC	-	Friend Relationship Strength Mobile Crowdsourcing
ICT	-	Information and Communication Technology
IT	-	Information Technology
LoC	-	Levels of Competence
LoI	-	Level of Information
MD	-	Multi-Dimensional
R&D	-	Research and Development
ReLU	-	Rectified Linear Unit
RT	-	Request Tracker
SAT	-	Server Allocated Tasks
SAT	-	Propositional Satisfiability Problem
SMA	-	Stable Marriage Algorithm
WST	-	Worker Selected Tasks

Chapter 1: Introduction

1.1 Background

Task allocation is one among the key planning exercises that plays a major role in an organization's quest to satisfy Service Level Agreements and to attain operational excellence. Employees within an organization assume roles based on their skills, proficiency, and experience (Mulla et al., 2016). Since the execution of tasks requires specific skills, tasks need to be assigned to employees with appropriate skills/proficiency.

Incident management is a standard support activity in most companies aimed at ensuring a high quality of service and availability by restoring normal service operations as soon as possible and mitigating business effects. Enterprises also keep many applications in place to support their business. Diagnosing incidents is an important task because of complex triggers often aggregated by changes in the common environment, network, hardware and software (Liu & Lee, 2012).

According to Barash et al. (2007), one of the main indicators for an IT support organization's success is the amount of time it takes to resolve an accident. Reducing this value not only reduces total cost and cost of resource allocation but also increases customer satisfaction, which is one of the most important measures of a support center's performance. Effective assigning of duties to staff is critical in service delivery. It facilitates meeting the Service Level Agreements, makes good use of the employees and increases operational efficiency. Task allocation is a challenging issue that addresses the inter-dynamics of tasks and employees, taking into account factors such as diversity, usage, skills, and fairness.

This problem of task allocation whereby individuals are matched to tasks has been equated to the stable marriage problem which is the problem whereby two equal sized sets of participants who are elements of some underlying set or sets are to be matched or assigned to each in some way to meet some specified criterion-stability, which is dependent on the preferences of the participants (Gale & Shapley, 1962).

Gale and Shapley's algorithm has been used to solve problems that resemble the stable marriage problem in real life applications such as the college admission, the hospital resident's problem, employee shift scheduling for jobs and application in router technology. However, Gale and Shapley's algorithm major concern is its non-truthfulness also known as the woman pessimal or man optimal results. This is the result where every woman gets her worst choice partner or every man does not get the best partner he would have opted for. This means that the marriages are stable

but the women or men are highly unsatisfied with their partners (Gale & Shapley, 1962). Gale and Shapley's algorithm brings the aspect of human consideration whereby the employee's preference are taken into account for harmony and cohesion during execution of a task.

Gale and Shapley's algorithm has been used widely and different variations have been formed and improvements made to suite different problems in hand. (Iwama & Miyazaki, 2008). Most variations have been used to solve particular problems. Giannakopoulos et al. (2016) in their paper on a question of equitable stable marriage, try to find stable match which satisfies the notion of equity between the two sides of the market examined. Such equity means preventing varying degrees of satisfaction between the two sides. They focused on equity which is not addressed by the Gale and Shapley's algorithm. Another variation which was a rotational-base formulation of the stable marriage focused on many to many stable matching which is a subset of the general whole. Their experimental analysis of two hard forms of stable matching called sex-equal and balanced stable matching found that the SAT formulation outperformed the best Constraint Programming approach.

Alternative algorithms for the Gale and Shapley's algorithm mostly provided stable matching but did not address the human aspect of work thus there was no cohesion among employees. Some alternatives paired together employees to perform a task having proper skills but the employees were not of each other's preference. This did not provide efficiency where employees were goal oriented but were not friends. Other alternatives addressed the issue of job scheduling and task allocation but do not provide solutions in cases where more than one worker were assigned to a task. They also overlooked major factors such as knowledge and skills possessed by the human resource as determinants for efficient task allocation.

Research done showed that most current task allocation systems focus on task allocation efficiency without focusing on the human aspect of choice of partner and employee or worker attributes. Even though they focused on efficiency of task allocation, they did not investigate the relationship between the team members who will work together by providing cohesion between them.

1.2 Problem Statement

Two employees possessing same skills but have different proficiencies in those skills can take different times for a task to be processed. When assigning tasks, consideration must be given to a number of factors such as task complexity and the skill requirements specifications, processing times and employee skills, workload, usage and fairness (Paul et al., 2017). Presently, all these factors are mentally processed making the problem of task allocation extremely challenging and making it almost impossible to keep in check the above-mentioned factors and thereby having a higher risk of impacting the completion of the task.

According to Paul et al. (2017), previous works have not addressed the issue of matching two or more employees to the task with qualities possessed by the employee, preference of the team or partner given and the qualifications for the task given. The research problem addressed by this study is how to assign a task to the right combination of employees having preferences and attributes by developing a model that will correctly assign tasks to the right combination of employees completing the task in the shortest time thus ensuring efficiency and productivity.

1.3 Aim

The aim of this research is to formulate an algorithm to support task allocation using machine learning.

1.4 Research Objectives

- i. To investigate the factors to consider during task allocation.
- ii. To analyze challenges faced during task allocation.
- iii. To examine algorithms, approaches and models used in task allocation.
- iv. To develop a predictive algorithm for task allocation.
- v. To test the developed algorithm.

1.5 Research Questions

- i. What are the factors to consider during task allocation?
- ii. How is task allocation challenging?
- iii. What are the algorithms, approaches and models used in task allocation?
- iv. How can a task allocation predictive algorithm be developed?
- v. How can the functionality of the developed algorithm be tested?

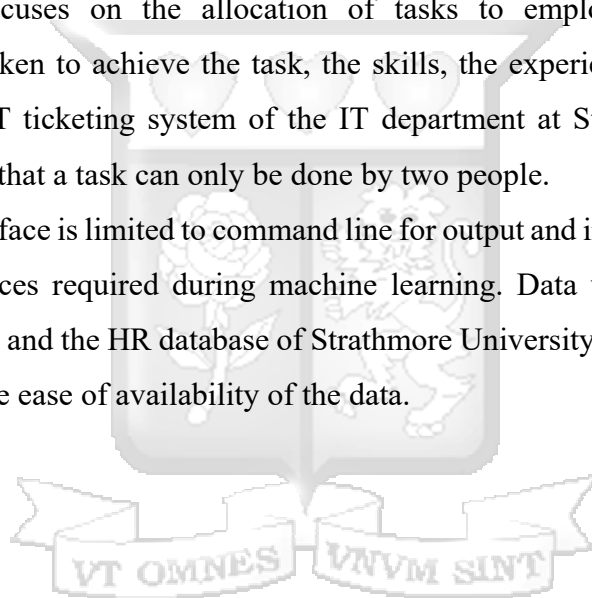
1.6 Justification

The research aims at formulating an algorithm that improves task allocation with machine learning. This can enhance the matching of two employees who need to work together thus being an improvement to the body of knowledge. By having a correct pair matching of individuals to the task, work is done in the least time and individuals work harmoniously and therefore improving efficiency and productivity. In the end the customer experience is improved. Towards the body of research Gale and Shapley's algorithm is reviewed to bring out the human aspect to task allocation. This brings about the employee preference in partner selection.

1.7 Scope and Limitation

This research focuses on the allocation of tasks to employees while taking into consideration the time taken to achieve the task, the skills, the experience and the gender. The study is limited to the IT ticketing system of the IT department at Strathmore University. An assumption here is made that a task can only be done by two people.

The research interface is limited to command line for output and interaction with the model, this is due to the resources required during machine learning. Data will be limited to the IT ticketing system database and the HR database of Strathmore University due to the limited time in doing the research and the ease of availability of the data.



Chapter 2: Literature Review

2.1 Introduction

This section provides information regarding task allocation challenges, a review of the current approaches in task allocation, a review of the variations and alternatives for the Gale and Shapley's algorithm, a review of the machine learning algorithm used and the conceptual framework that will give more information on the proposed model.

2.2 Challenges in Task Allocation

These are the problems faced during assigning of tasks that have proved to be a setback in the efficient and effective completion of tasks. Challenges include: group cohesion, allocating the right person for the job, saving cost and time and best fitting tasks to human resource. These are explained further below.

2.2.1 Maximizing Group Cohesion

Group unity is a challenge because the success or failure of a group is also based on the interdependence between the leadership skills of a group, their cooperation, confidence, and the technical skills of each of its leaders. Building a high team-work quality that stays together and collaborate is based on an appropriate level of cohesion (Baiden & Price, 2011).

According to Pérez et al. (2012), project managers from the standpoint of social interaction need to make better decisions when it comes to objectively deciding which human resources will perform better and if joining with other individuals from a given pool of resources improve their performance. Here, they equate this to Sociometry itself, as it is founded on the same theoretical premises that, if there is a probability of choosing participants to create one or more group teams of social experiences, an optimum result would be obtained.

From a mathematical calculation procedure, the developed, empirical research showed that members of a team with high social interaction displayed higher affective commitment and had higher performance working as a unit than groups with low social cohesion. This showed that human capital is an organization's absolute requirement, and various results can be achieved based on whether there is a proper balance of workers, role assignment, trust and motivation (Andrews et al., 2008).

The sociometric approach of job performance from a social relations perspective and group interaction partially addresses the problem but leads to other problems such as a group that has

socially fit individuals but do not have the technical requirements or a technically fit group but not socially fit. These two problems occur in large groups as explained. This shows that although the outcome of a team is highly dependent on how individuals communicate socially and build relations, it also is an outcome on the contribution of an individual towards the project. This is by the technical requirements required for the project which are composed of the individual skills possessed (Andrews et al., 2008).

2.2.2 Allocating the Right Human Resources to Multi-Project Environments

According to Hendriks et al. (1999), this is a challenge because the more projects that are involved and the more specific knowledge that is needed in every project, the more important, but also the more difficult, is the allocation process.

Hendriks et al. (1999) go further in explaining that it is also a challenge because poor allocation of experts to projects results to resource overload and understaffing, project delay, interference from management which is the need to follow up and micromanagement brought about by project delays and poor planning and finance and Budget allocation. Discipline where efforts are going to decline in the coming years needs to have a decreasing budget, disciplines where efforts are going to rise require more resources and an increasing budget. Due to the particular characteristics of each project based on a high degree of creativity, the project outcomes and the project timing are very unpredictable. Human capitals are the key and limited resource for R&D ventures. Knowledge is minimal. And nearly everybody is making their own small unique contribution to any project.

The success of each project relies heavily on the state-of-the-art building blocks that scientists and engineers use to create. Human creativity relies very much on the inspiration and commitment of each human engineer. In the R&D community, the assigning of experts to one project may be challenging due to the limited expertise that most people have. Human knowledge being a scarce resource it is very vital to allocate a project to the right human resource. The more project involved, the more detailed information required for each project and the more complicated the method of allocation becomes. For projects to beat the deadline and finish in good time, there needs to be a balance in the number of human resources assigned to one project (Hendriks et al., 1999).

Hendriks et al. (1999) proposed two solutions whereby resources had to have a resource dedication profile whereby each resource has his or her own specialization and allocation is done

using this profile. To them, this substantially influences the performance of an organization. The second solution was the project scatter factor which is the number of staff/resources needed to fulfill a one-year task. The lower the scatter factor means that less human resource will be needed on task/project thus an efficient allocation is observed.

Nedzelsky (2016) also raised this challenge whereby lack of available human resources due to over allocation, under allocation or unavailability lead to constraints in project/task scheduled. The project does not end in time and performance is jeopardized which is also known as Resource Constrained Project Scheduling. According to the Nedzelsky (2016), projects take longer due to the lack of human resource information in the HRIS system that would be essential during the allocation of a task to a human resource.

A balance is achieved on a project's portfolio which is a key concept of project methodologies when human resource information is added on the HRIS system. Resource availability has led to stalled projects especially when a resource leaves the company and no one else can handle the same work. In his approach in adding the information of the human resources to the system this is in terms of skills and education plan, reduces the duration of time to accomplish a task and increases efficiency. He further explains that a set of attributes identifying with particular roles by the resource better feature the specialization of every employee. Prompting a reality that every employee exercises what he is best at (Nedzelsky, 2016).

Having looked at specialization, which solves the over allocation and under allocation problems and in a way the unavailability of human resource to a project. He also highlights that in times of unavailability of resource, it is easy to assign another resource to the project who has the skill but no experience to hold the position until a fitting resource has been assigned to the project. He takes into consideration resource information pertaining to availability of resources in relation with the individual's schedule, the duty of the source taken from the individual's profile and lastly total capacity taken from the resource's information. This data is deemed useful especially in allocation of projects with regards to specialization (Nedzelsky, 2016).

2.2.3 Time saving by Efficient Allocation of Professionals

According to Daniel et al. (2009), one of the main challenges for software development organizations is signing off a quality software product within its planned deadline. They believe that the main cause for this delay is the time taken by professionals to learn different skills to undertake tasks and yet failure to meet a timetable that result in financial losses for the

organization. They further note that most software development projects have fallen because they have not gained adequate consideration from project management on the appropriate distribution of human resources to projects.

They go on to state that the management of software staff remains a very complicated task, given all the work and advances in the field. This is because people play a fundamental role in software development, and evaluate a project's quality and profitability (Silva & Costa, 2013). Therefore, it is very important to staff a project properly (Daniel et al., 2009).

The implementation of the model is important as there are many projects with various requirements that directly impact the time required to complete the project by allocating a greater or lesser number of human resources. However, the approach used by the author assumes that variations in the skills of professionals are highly important when solving the issue, because the allocation of human capital in software projects is especially complicated since developers' human characteristics affect the allocation. (Kang, et al., 2011).

However, this strategy does have certain drawbacks. First, developers are essentially not shared among teams, and sharing developers among teams lowers productivity as different teams are under the control of these developers. The second relates to project reach. When the scope changes, this change will only impact the time required to carry out the project, and not the reallocation of human capital, as that has been established previously. In the other hand, if a professional leaves the project, the team remains the same, because no project can be disrupted and/or delayed and this model does not involve sharing human capital (Kang, et al., 2011).

2.2.4 Human Resource Allocation to Tasks for Project Cost and Time Saving

As the size of software project increases, software planning process becomes more complicated and important. An inappropriate software plan often results in the failure of a project. The basic goal of the issue of human resource management is to create a project schedule that can be completed within the required time period, in order to minimize the overall cost of the project in terms of time and resources. The longer the project, the higher the cost and this may cause project failure (Park et al., 2014).

A developer can operate on more than one task at a time, although other researchers thought that only one task at a time could be performed by a developer. If at any stage a developer is involved in so many tasks, productivity may decline because the developer is unable to focus on one task due to constantly juggling tasks (Park et al., 2014).

Since the preceding task relationship suggests closely related tasks, assigning both pre-task and post-task to a developer is efficient in reducing context-switching costs. If a developer is required to work on a set of tasks that are not connected to each other, the developer may need to learn the new context for each assigned task. In allocation balance, the job size should be considered in the question of the allocation of human capital. On the one hand, if a few developers are assigned to a huge job, then the heavy workload will overwhelm developers. By comparison, if too many developers are assigned to a small job, the high overhead communication causes inefficiency (Park et al., 2014).

Developers with high staff level have more experiences than lower level developers, so they will manage a task rather than concentrating on the implementation, which is the main work of a low staff level developer. To manage each task efficiently, developers having different staff levels should be assigned together (Park et al., 2014).

In Park et al. (2014) search to assign a resource automatically to the task and to schedule the project optimally in terms of time and resources, they based their research on key practical consideration such as short project plan, reduction of multitasking time, allocation on relevant tasks and allocation balance.

Park et al. (2014) proposed solution for solving the human resource allocation problem uses a genetic algorithm that focuses on the practical considerations. By keeping in mind minimizing of the cost, they have also considered practical issues affecting a project schedule during planning. They also looked at the problem in a software project planning point of view, that even though the main purpose is to reduce cost, human resource allocation is also of important.

2.2.5 Task identification and Assigning Suitable Employees to Best Fitted Tasks

Lack of flexibility in human resources and poor project planning and scheduling increase inefficiency and projects delay leading to increased costs (Avinash & Ramani, 2014). Existing exploration shows that task pre-emption can decrease the time and cost of the product venture. Be that as it may, it decreases the adaptability of human asset allotment in venture arranging. In their quest of identifying task needs and assigning of appropriate resources to tasks based on their skills and experience with least cost, they propose an Event Based Scheduling technique and Ant Colony Optimization techniques to provide software project Scheduling and human resource allocation.

The event-based scheduling technique reduces cost and enables optimized resource utilization while the task precedence graph obtained through Ant Colony Optimization provides efficiency as the best employee is selected (Avinash & Ramani, 2014).

2.3 Current Approaches in Task Allocation

2.3.1 Software Architecture and Social Network Human Resource Allocation Method

Zhou (2008) presents an approach where he investigates resource allocation with a social network approach whereby an algorithm for matching employees and tasks is proposed. The algorithm is based on task attributes, software architecture, employee skills and employee preference and social network relations.

Zhou (2008) takes into account the employee's happiness and control of the issue and the qualities of the tasks for which he or she is most eligible, and distributes the workload as equally as possible. He also considers this approach the relationship between the job and the interpersonal relationships between the tasks. The currently applied solutions to this problem include the assigning of tasks by a manager without properly understanding the relationship between tasks and employee relations. Having in mind the human dimension by which the success of a project is directly linked to the standard of skilled people working and the manner in which management deploys resources to the project.

Although this research talks about employee preference on the task, it focuses on tasks that require two or more people. These tasks require cohesion between the employees. Even though the employees have been selected to do the work by their skills and probably they are the best at it, if they are not able to work together then efficiency will not be seen as either the task will not be completed or not done well or not completed in due time. An improvement to his model is to add cohesion as a feature whereby employee preference on who he/she wants to work with is considered. But this will most definitely depend on the available human resource (employees) and the skills set.

2.3.2 Multidimensional model approach for the deployment of human resources

Chen et al. (2009) approach aims to address the human capital distribution dilemma efficiently and rationally with a view to achieving the greatest gains with minimal resources. Their approach puts forth a multidimensional model of the process of allocating human capital in multi-projects. The model is based on the idea that the length of the edges of the object is taken as the

ratio of the capacity value required to do a job to itself. This builds a multidimensional model (Baseline MD Model). The theory goes further to take the ratios of a person's skill to the ability value needed to do the job as the length of the model's edges to construct another one. Competence is obtained by comparing the volume of the two MD models, resulting in a participant carrying out the task according to the task's needs. Further estimation of the summation of the effects (the member's effect performs the task and effect of each participant performing each task in the projects) gives full effect to the multi-projects.

According to Chen et al. (2009), the best allocation of human resource is given when the positive effect of multi-projects comes to maximum and the personnel capabilities meet the task requirements. The approach focuses on an MD model which focuses on improving human resource quality to meet the needs of the projects. The other strength they have is that, in terms of consistency, they look at the human resource, which means allocating the participants to the tasks based on the various abilities of individuals.

Chen et al. (2009) explains the attributes of everyone as different from others and brings the greatest difference to other resources. They go further to show that the MD model is an improvement of the cube model by its multidimensionality. A cube model cannot be used to determine a matter of more than three attributes.

The major weakness of the model is that it does not address the emergent attribute from the human resource. Due to the changing nature of human beings, a person may develop negative or positive attributes these can still be used in the model. This is where artificial intelligence comes in whereby every attribute deemed to be possessed by the human resource will be used. Every attribute that emerges will be fed to ANN as inputs. With ANN there are no limits.

2.3.3 An Intelligent Tool for Dynamic Task Allocation

Samath et al. (2017) in their quest to optimize the use of the available resources in order to deliver a high quality software product for project management software, they introduce a task management method that dynamically assigns tasks based on (historical data) abilities and previous team members' work. Their proposed method would collect data from various data sources, then predict each team member's aptitude for a given task and then assign tasks to the project in the most optimal and feasible way possible.

From the beginning, they defined the necessary databases by getting the history of tasks assigned to the workers, getting the employee information and also allowing the employees to rate

themselves. This data was collected and pre-processed in a data warehouse. Prediction analysis is performed on the data to know the employee suitable tasks as dependent on their skills. The predictive analysis results are sent to a task assignment and scheduling model which based on project limitations allocates teams to projects. There is also a peer reviewing feature that allows employees to review and rate their colleagues and this data will be summarized to be used for allocation on a later date (Samath et al., 2017). Figure 2.1 below depicts the model architecture of the components and how it works.

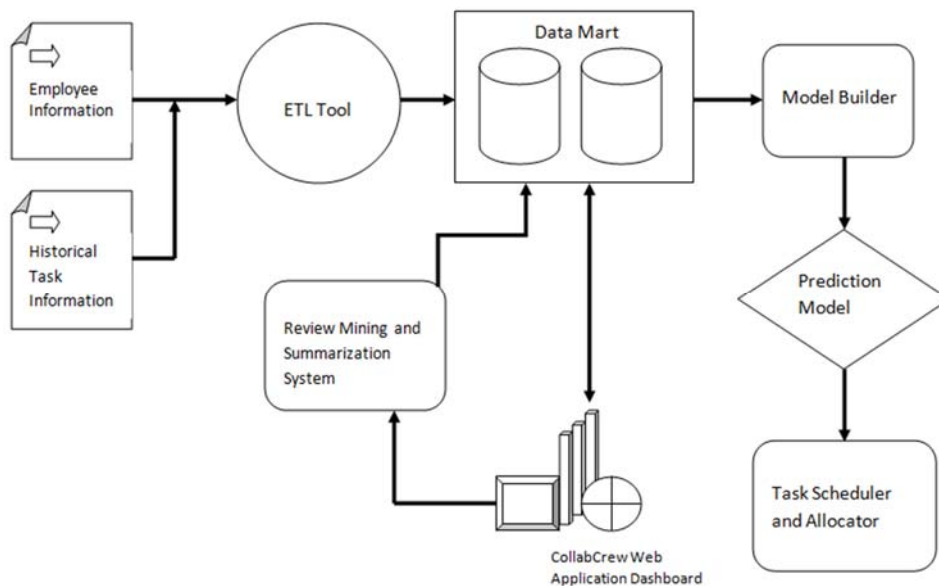


Figure 2.1: Collabcrew System Architecture Outline (Samath, et al., 2017)

Here the people factor is appreciated in that proper allocation of human resource to a task is key for a successful outcome and that this requires critical thinking and frequent team meetings and discussions.

A task management tool known as CollabCrew is introduced that dynamically allocates tasks based on the previous history of tasks done by team members and also the skills they possess. According to them, this program uses historical data for comparison in research performed by team members from its own database or from an external source to automatically allocate new tasks to it. This tool saves time and reduces complexity for project managers in trying to allocate the right resource to the right task (Samath et al., 2017).

This solution, however, does not investigate the relationship between the team members who will be working together. It focuses on the efficiency in the point of view of historical data of

how a task was done by each individual and the skills possessed by everyone. Cohesion is not emphasized if the task has been completed. Samath et al., (2017) show this by explaining that the purpose of their research is focused on an intelligent system that allocates team members dynamically to tasks based on historical data of previous works done by team members and skills possessed by the members within software based projects.

2.3.4 Human Task Management Reference Model

Schulte (2012) introduces a reference model for human activities that is used to handle human activities across structures and organizational boundaries. The several structures that help human task management have given rise to this and yet no comprehensive analyzes of the components involved have ever been conducted. This approach stimulates discussions on ways to solve tasks being done by resources crossing systems and organizational boundaries. By offering a framework that analyzes and compares current human task management solutions, this model promotes the creation of distributed and decentralized human task management solutions free of solid process automation systems and web application.

The reference model focuses on human task management. Schulte (2012) recommends the use of a personal task manager which allows a personal to list, view his current tasks, tracks their states and maintain relevant information. It additionally is platform and process independent as it is able to pass information about tasks to task managers of other individuals and can provide staff resolution facilities (Schulte, 2012).

Schulte (2012) covers the identification of a human task management solution that allows a person to maintain relevant information, outline all his tasks and track their states on the task resolution and task distribution. He does not view the human aspect in accomplishing the tasks, this is seen when he mentions that interaction patterns need to be analyzed as future works which is of most important. Interaction patterns is the cohesion that can be brought by two or more people working together to accomplish a task whereby each having the skills required for the task and preference on who to work with has also been factored.

2.3.5 Task allocation in production systems

Fast-Berglund and Stahre (2013) address the need for a quantitative and easy to use method for task allocation with the paradigm of mass customization. They present a design model of task allocation consisting of a five-step main loop assisted by other areas where information is gathered

to allow a decision to be made in the main loop. The concept model is designed to allow companies to increase versatile assembly methods and means for making assembly systems more proactive.

In addition, their model also considers the operator group's cognitive LoA, the LoI to and from the operators and different LoC levels. Figure 2.2 illustrates the concept.

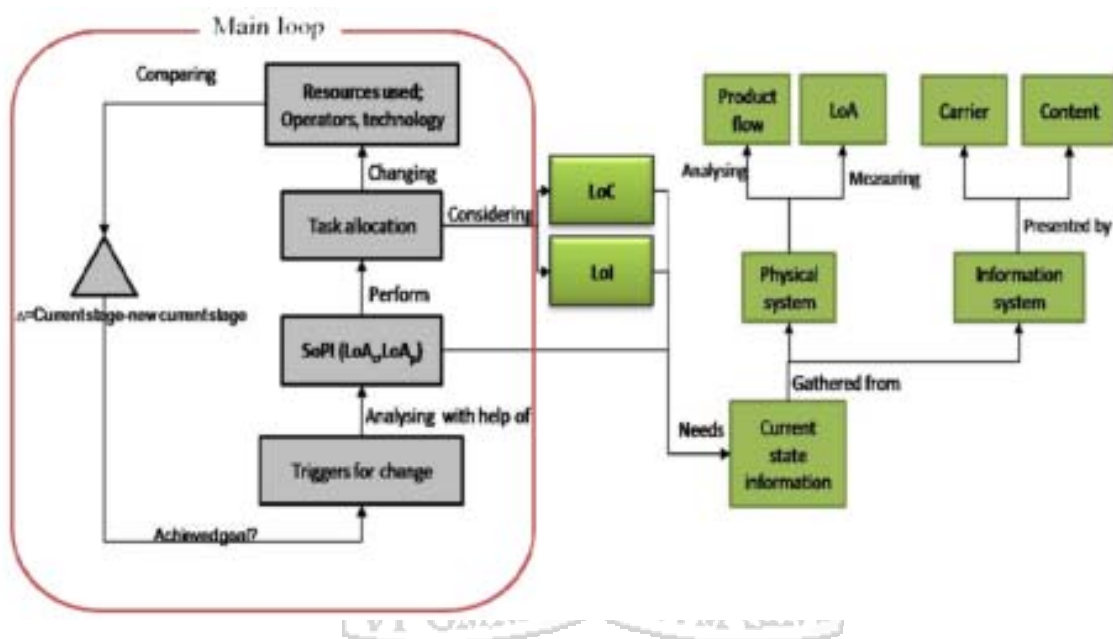


Figure 2.2: Concept Model, Further Developed from DYNAMO++ (Fasthe & Stahre, 2010)

This concept yields industry-relevant results and improves the efficiency of advanced, semi-automated device analysis for manufacturing. They also note that the validation and evaluation shows that the design model is easy for end-users to use and offers quantitative evidence for the comparison and analysis of various solutions. The various systems may be used at a comprehensive level depending on the client's reasons for modifying the system. It is true of Older's, et al. (2017) criteria to be cost-effective and convenient to use.

The concept model aimed at visualizing relationships between various fields and activities within an organization while redesigning a program in terms of level of automation through task allocation by adopting Older's, et al. (2017) 15 task allocation criteria process. It does not go

further to look on the skills of the personnel allocated the task or any of their attributes. Though this approach may validate that efficiency is improved, it is implied that levels of automation will improve efficiency. The concept as tried should be researched further to have additions of intelligence in the automation process.

2.4 Variations for the Gale and Shapley's Algorithm

Gale and Shapley's algorithm has been used widely and different variations have been formed and improvements made to suite different problems in hand. (Iwama & Miyazaki, 2008). This section analyses some of the different variations from the original Gale and Shapley's Algorithm.

2.4.1 An equal approach to the stable marriage problem

Giannakopoulos et al. (2016) in their paper on a question of equitable stable marriage, it is important to find a stable match which satisfies the notion of equity between the two sides of the market examined. Such equity means preventing varying degrees of satisfaction between the two sides. They're proposing an efficient and practical ESMP solution. They avoid a thorough search for stable matching and instead build on the Gale-Shapley algorithm in a way that treats both sides fairly and still achieve stable solutions. Attaining that equity is a nontrivial issue. They go further than previous research to perform a detailed systematic analysis of techniques on much broader data compared to a recent alternative that does not always end, as well as the Gale-Shapley algorithm.

According to Giannakopoulos et al. (2016), whereas the problem calls for the matching of men and women in such a way that there are no two people who would rather marry each other than their assigned spouses. The Gale-Shapley Stable Marriage Algorithm (SMA) only yields highly satisfied men and dissatisfied women, or vice versa, an ideal male or female solution, so it is inadequate for most real-world applications, requiring matching not only to be stable but equal.

Giannakopoulos et al. (2016) concluded that the algorithm would handle all sides in a rational and non-discriminatory manner in order to find a more equal solution to the SMP. Ideally the algorithm should be gender-blind, and the functions on both sides should be interchangeable. Thus, both genders (groups) were permitted to serve as proposers as well as accepters. In particular, they assigned the function of proposer to one side and that of acceptor to the other at each iteration of the algorithm. This was contradictory to the original Gale-Shapley algorithm in that the Gale-

Shapley algorithm assigns the two sides of the problem with two narrowly defined and mutually exclusive functions. One side is appointed as proposers, assuming an active role, while the other side is relegated to acceptor's role, anticipating and responding to the initiatives of the proposers. The result of the algorithm is therefore proposer-optimal, and acceptor-pessimal. Their approach produces a problem of non-termination often known as the issue of circular dependencies, because the repeatability of certain preferences will lead to endless repetitions of the states of the algorithm.

According to Giannakopoulos et al. (2016), unless the agents' preferred lists follow a circular pattern, there may be an endless loop in which a group of agents continuously reissue and break with each other's proposals. So long as the decision on who serves as proposers is taken in a state-dependent way, it is likely to undergo an endless loop until the system arrives at a state it was in before. The same conditions will result in an endless repetition of the same sequence of acts and return to the same state. For the correctness and termination of the algorithm, the monotonic property retained by SMA is necessary. Any approach that attempts to provide "equal" features to the problem while infringing this monotonous property can lead to unstable solutions and/or fail to terminate. A solution for this proposal was found, the use of a periodic function as a picking method for the proposer. This leads the algorithm to infinite loops, as the task time was correlated with the problem states being repeated. The ESMP cannot be used in our situation as an algorithm to solve matching, because it can lead to endless loops if consistency is not achieved. This will consume more time in solving the match problem.

2.4.2 Rotation-Based Formulation for Stable Matching

Siala and O'Sullivan (2017) introduce new Constraint Programming model for the many-to-many stable matching problem. They use the notion of rotation to have a novel encoding that is linear in the input size of the problem. They provide extra filtering rules to maintain arc consistency in quadratic time. Their experimental analysis of hard instances of gender-equal and balanced stable matching reveals the effectiveness of one of their solutions relative to the state-of-the-art constraint programming approach.

According to Siala and O'Sullivan (2017), every man has a list of preferences over women, and vice versa. The goal is to find a match where each man is associated with at most one woman who values a criterion called stability. A matching is stable if any pair (where m is a man and w is a woman) that does not belong to M satisfies the property that m prefers his partner in M to w or w prefers her partner in M to m . However, when dealing with circumstances in the real world, the

issue also requires additional optimist requirements. For certain cases, the problem is intractable, and advanced algorithms used to solve the standard version are typically difficult to adapt. The use of a modular approach, such as constraint programming, is very useful when dealing with these situations. In this paper, the authors continue this line of research by proposing an accurate and efficient model for all three stable matching variants: one-to-one, many-to-one, and many-to-many. Their proposals are based on a powerful structure called rotations. The latter was used to model the problem of stable roommates. They exploit some established rotation-related properties to suggest a novel SAT formulation of the general case of many-to-many stable matching. They have been able to demonstrate that the unit distribution on this formula ensures the survival of a particular solution. Next, they used this property to give an algorithm that maintains arc consistency if one considers many-to-many stable matching to be a (global) restriction. Their experimental analysis of hard instances of gender-equal and balanced stable matching shows that their approach is superior to the state-of-the-art constraint programming approach.

Siala and O’Sullivan (2017) in their results in the absence of known hard problems for many to many stable matchings, proposed to evaluate their approach on two hard problem variants; the sex-equal and the balance stable matchings variants of stable marriage. Using fundamental properties related to the notion of rotation in stable matching, we introduced a novel SAT formulation of the problem and then showed that arc continuity can be sustained in quadratic time. Our experimental analysis of two hard forms of stable matching called sex-equal and balanced stable matching found that the SAT formulation outperformed the best Constraint Programming approach in this literature.

Siala and O’Sullivan (2017) addressed the general case of many-to-many secure matching within the context of constraint programming. Since this was not experimentally evaluated on the propositions on hard variants in the many-to-many setting it cannot offer a solution to our problem even though constraint programming was being improved. It focuses on only the many-to-many matching which is a just but a subset of our target issue as it includes one-to-one and one-to-many stable matching.

2.5 Alternative Algorithms for solving Task Allocation

Task allocation being a challenging situation, different algorithms have been used or modified to solve the task allocation problem in different scenarios. This section will analyze some of the alternative algorithm for solving task allocation.

2.5.1 Dynamic Task Allocation Algorithm for Hiring Workers that Learn

Pan et al. (2016) work explores the issue of hiring workers who improve with experience. Experiments are performed to demonstrate that the output of employees increases with experience and that it is possible to model and forecast their learning rate. They suggest a complex recruiting process that represents the learning ability of workers. Both simulation and real-world crowd sourcing data indicate that their recruitment process can lead to high-precision results at lower cost compared to other mechanisms. Pan et al. (2016) propose to model the learning cycle of crowd workers using hyperbolic learning curves, to predict both their present output and their predicted future development. By allowing the system to balance the cost of hiring workers, the quality of the task results and the future benefit got from allocating tasks to workers for training purposes show how possible it is to incorporate information about workers learning into the hiring process.

In this work, Pan et al. (2016) demonstrated that, with some types of tasks, crowd workers will benefit from experience and improve their quality of work over time. After 15 separate case studies, a decision-theoretic selection paradigm was developed that reflects the thinking process of the workers. They found out that everyone's learning curve could be modeled and be presented on a decision-theoretic hiring model. The findings of the simulation and the tests have shown that, in terms of accuracy, unlike many widely employed hiring methods, their approach and model reduced the hiring costs and provided competitive performance.

Pan et al. (2016) model makes the unreasonable supposition that workers are constantly available; A further move forward in their research is to apply the concept and algorithm to the contexts where workers will join and exit the application., prompting intriguing elements as the framework attempts to find the correct harmony between greedily conglomerating answers from at presently available staff and educating potential employees whose future employment is uncertain. This can be easily achieved with machine learning algorithms whereby the algorithm adapts to different settings which can go beyond consensus task to handle even more complicated hierarchical tasks. This algorithm does not provide a stable allocation as random decision based on the learning potential while Gale and Shapley's algorithm provides a stable environment for the allocation of task.

2.5.2 Solving Task Allocation to the Worker Using Genetic Algorithm

This paper deals with the issue of job scheduling and staff distribution, in which each qualified staff is capable of performing several tasks and has different normal and worker ability to execute tasks, and the assigning of tasks is carried out on a daily basis. Kotwal and Tanuja (2015) propose a worker assignment model and create a heuristic algorithm that is a genetic algorithm whose output is to be tested against the best methods of discovering small-scale problems. According to them, the genetic algorithm is implemented in a manner that eliminates the amount of intervention required to consider the current solution. Genetic algorithm is basically used to minimize the total make-spam for scheduling jobs and assigning task to the worker.

Kotwal and Tanuja (2015) primary goal was to devote a number of employees to an equivalent amount of operations in order to minimize overall costs and thus to increase net income when assigning jobs / tasks to workers. According to them, the question of assignment occurs because the available tools, such as people, machines, etc., have differing degrees of productivity for carrying out specific tasks. Therefore, the expense, benefit or time of conducting the various activities is different. The question, therefore, is how the assignments can be made in order to maximize the goal. There are a variety of officers and a range of duties involved.

Any agent may be assigned to perform some role, incurring some expenses that will vary based on the nature of the agent job. Kotwal and Tanuja (2015) propose an algorithm that dynamically incorporates a heuristic approach during the decoding process to make it easier to solve GA job assignment problems. Use the GA operator as: Selection method, fitness function, crossover and mutation operator. They use a genetic algorithm, a type of machine learning, which draws its behavior from a representation of evolutionary processes in nature. It is achieved by generating inside a computer a population of individuals represented by chromosomes, basically a collection of character strings that are similar to the base-4 chromosomes that we see in our own DNA. Individuals in the population then undergo a phase of evolution. In the first case, their process starts with a solution called a demographic solution from one demographic to create a new population and that would provide a new population that is stronger than the existing one. Here in the population, every chromosome is represented as chromosomes. In each generation, the three key operators of genetic algorithm are performed; the selection, crossover and mutation.

The main weakness is that the algorithm cannot give solutions in cases where more than one worker is assigned to a task. This has been seen by the assumptions made by the authors. that

a worker is assigned to only one station and that the job assigned to that station is only processed and that the job cannot be split between two or more workers. This has also been shown when they conclude that future works should extend by assigning more than one worker to the job and the processing time often varies. Genetic algorithm cannot be used in our model as an algorithm to solve task allocation as it will not be enough to solve issues addressed.

2.5.3 Worker Friend Relationship model for task allocation

Zhao et al. (2019) propose mobile crowdsourcing system a task allocation model that considers attributes of social networks based on friend relationships. They protect the location privacy of workers by calculating the strength of the worker relationship using a coding mechanism known as GeoHash coding mechanism.

According to Zhao et al. (2019), tasks of mobile crowdsourcing systems are in two distinct modes: Worker Selected Tasks and Server Allocated Tasks. For WST mode, online workers can pick any spatial errand published by close by requesters without coordination with the mobile crowdsourcing server. For SAT mode, with the goal that the platform can apportion errands to close by workers to streamline the adequacy of mobile crowdsourcing, online workers occasionally report their location to the platform.

Zhao et al. (2019) further explain that workers do not have to disclose their position to the platform in WST mode, but in SAT mode the platform needs to monitor workers' positions and protect workers' spatial privacy. However, certain tasks can never be selected in WST mode because the platform has little influence over workers' characteristics. In SAT mode, however, the server will delegate these tasks to the staff to optimize the number of assigned tasks or to optimize the utility of the platform. So, in their paper, they suggest a model of task allocation in SAT mode.

Hence, their paper discusses in detail the relationship between people and their friends in order to delegate the same or similar tasks that the worker has performed to the friends of the employees, or to delegate specific tasks to the employees and their friends, thus minimizing costs. They continue by saying that this would not only increase the quality of sensed data and the completion rate of assignments but also save the costs of traveling to the target location, which will minimize unnecessary overheads and optimize the use of current resources.

Zhao et al. (2019) propose a Friend Relationship Strength Mobile Crowdsourcing that incorporates the adaptive threshold algorithm with GeoHash coding that aims to explore deeply the strength of the relationship between workers and their friends. Under the limitations of the task

and budget deadline, their paper applies friendly relationships to the mobile crowdsourcing task allocation problem, which can improve the accuracy of the task allocation, reduce the workers' travel costs and optimize the total allocation utilities (defined as total workers' utilities). Given that the platform needs to monitor workers' positions in SAT mode, it needs to protect workers' spatial privacy. This research also notes that GeoHash coding can provide a high degree of security of the spatial location information of workers.

Zhao et al. (2019) first explore the timely attribute of relationship strength between the workers and their friends. They then explore the strength of the relationship between workers and their friends in terms of geographical attributes. At the end of the day, their proposed FRS-MC algorithm is used to select the workers with the highest relationship strength. The FRS-MC algorithm measures the time relationship strength and the geographic relationship strength of the newly emerging worker and his colleagues, who already live in the mobile crowdsourcing network. The system selects the worker with the strongest relationship with the newly emerged worker and finds the task category this worker has accomplished. The system then considers the current activities in this group that it has. The system then selects the task from the task collection for allocation for them with the highest budget.

The algorithm helps workers to perform the part of the spatial task close enough to them so that workers can travel to the position of the spatial task before the time limit for the job. Thus mobile crowdsourcing systems can assign or suggest spatial tasks to active workers close to the job site, their system allocates the most appropriate crowd workers for each task based on the spatiotemporal attributes and task characteristics. Another strength aspect covered by the team is the protection of the location privacy of the workers (Zhao et al., 2019).

The algorithm is based on the social relationship built around friends which is one aspect of the problem statement being faced by my model, whereby two people paired together to perform a task can have the proper skills but are not friends which might impede the task assigned. This overlooks the situations where workers are goal oriented but are not friends. The algorithm limits our research in finding an optimal pair for a task with the efficient time as it focuses on the social relationship while overlooking other major Factors such as knowledge and expertise.

2.6 Review of Artificial Neural Network

An artificial neural network consists of processing units called neurons. An artificial neuron tries to replicate the structure and behavior of the natural neuron. A neuron consists of inputs (dendrites), and one output (synapse via axon). The neuron has a function that determines the activation of the neuron (Kukreja et al., 2016). In multi-layer artificial neural networks, there are also neurons placed in a similar manner to the human brain. Each neuron is connected to other neurons with certain coefficients. During training, information is distributed to these connection points so that the network is learned (Mijwil, 2018).

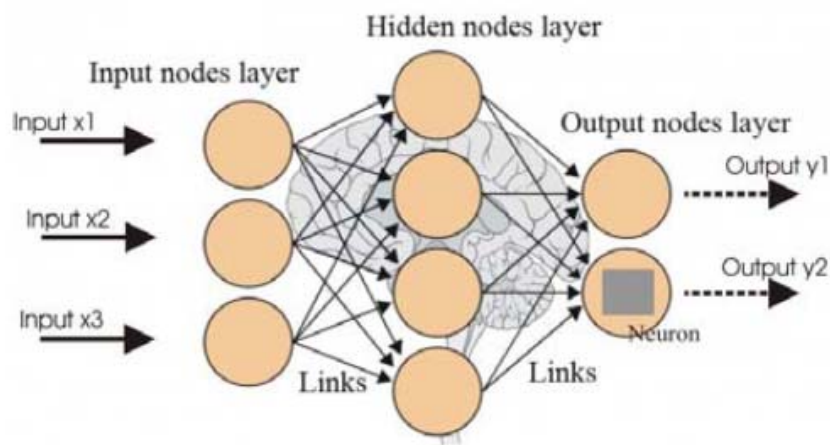


Figure 2.3: Layers of the Artificial Neural Network

As shown in Figure 2.3, a neural network consists of three layers: an input layer, an intermediate layer and an output layer. The blue boxes shown here represent the neurons and the arrows represent the connection points. The data set prepared for training at the input layer is shown to the network. The network assigns the weights of the events it learns to the connection points in the intermediate layer. Not every point has to be a value, and some points can be zero. A threshold value is added between the layers so that the zero values at the connection points do not become zero (Mijwil, 2018).

2.6.1 Justification for Artificial Neural Network

The choice of using Artificial Neural Network was purely out of the advantages offered by ANN. One of the most important advantage was the ability to work with incomplete knowledge. After ANN training, the data may produce output even with incomplete information. The loss of performance here depends on the importance of the missing information. Another important

feature is the ability of fault tolerance whereby corruption of one or more cells of ANN does not prevent it from generating output. This feature makes the networks fault tolerant. In addition to this (like other machine learning methods), they are capable of generalization, so they can interpret information which is different to that of the training data, thus representing a ‘real-world’ solution to a given problem by their ability to predict future cases or trends based on what they have previously seen. Thus, trained ANNs can be used as standalone executable systems in order to predict the class of an unknown case of interest, and therefore have the potential application in diagnosis (Lancashire et al., 2009). The ability of ANN to make machine learning whereby it learns events and make decisions by commenting on similar events is also a specified advantage of ANN. ANN also provides for parallel processing capability whereby they have numerical strength that can perform more than one job at the same time (Mijwil, 2018).

Finally, there are several techniques that can be used to extract knowledge from trained ANNs, and the importance of individual variables can be easily recovered using various methods such as the analysis of interconnecting network weights, sensitivity analysis and rule extraction. This, from a biological perspective, is perhaps one of the most useful aspects of ANN modelling (Lancashire et al., 2009).

A comparison was made between ANN, Naïve Bayes and Support Vector Machine results to determine the success rate. The results as displayed in Table 5.6. The output shows that ANN outperformed the rest. The precision of ANN was at 71.34% as compared to Naïve Bayes and Support Vector Machine with 61.33% and 52.7% respectively. This shows that ANN can be used as the machine learning approach for providing efficiency during task allocation.

2.7 Conceptual Framework

The model that the research proposes, addresses the issues by developing a model that solves the task allocation problem intelligently using a machine learning algorithm. The allocation problem being how to correctly assign a task to the correct/right combination of IT personnel having preferences and attributes and use the minimum time possible to accomplish this. Figure 2.4 represents the conceptual framework.

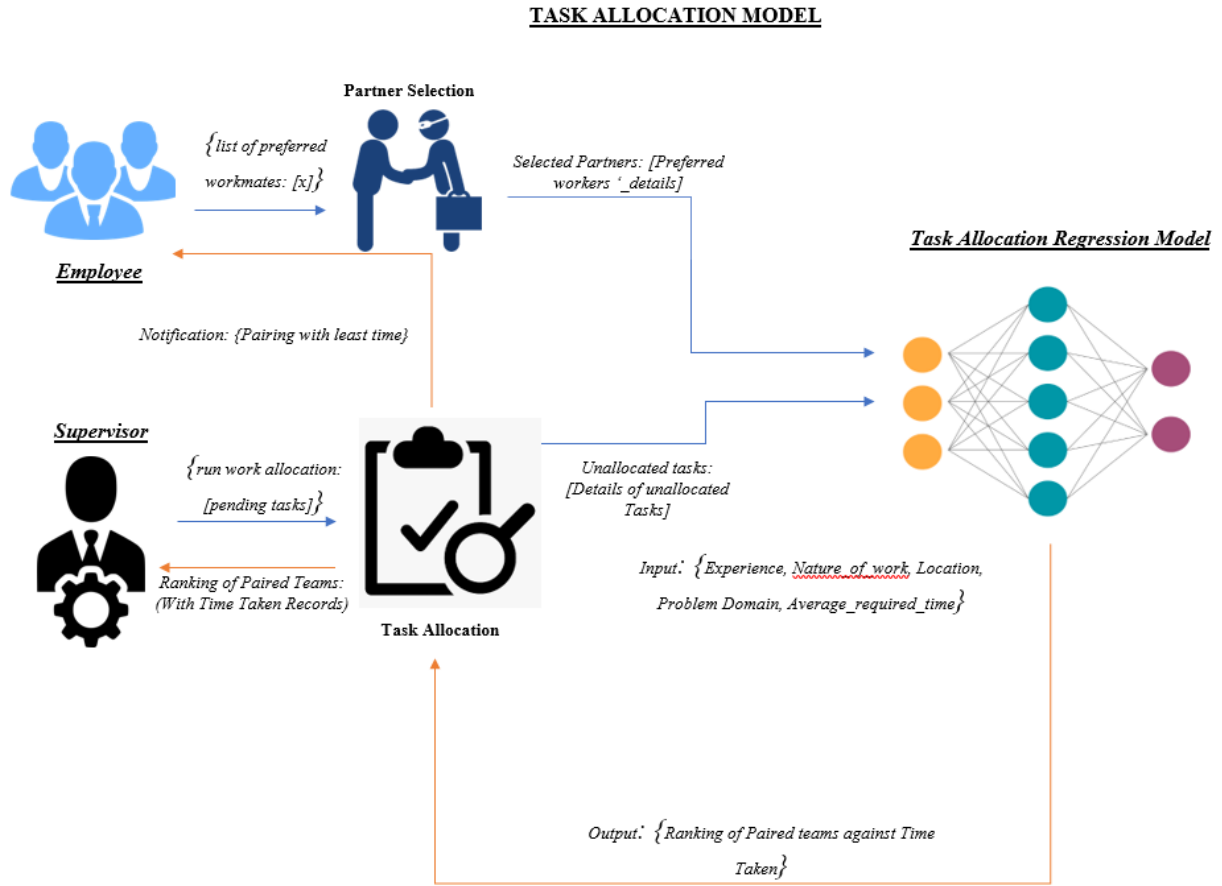


Figure 2.4: Conceptual Framework for A Machine Learning Model for Task Allocation

2.6.1 Partner Selection

The task allocation model allocates tasks to partnered pairs of employees who can complete the task in the least time. The process begins with the employee and ends with him/her getting information of who he/she has been assigned to work with on a given task.

An employee interacts with the system when selecting a partner, they prefer to work with. The employee is the one who initiates the process, the employee selects three people from a list of preferred partners he/she would like to work with. The selection from employee, is later fed to the neural network together with the task details that have been keyed in by the supervisor.

2.6.2 Task Allocation

This is where the task is matched to a pair and ranked. The inputs to the Artificial Neural Network (Task allocation regression model) will be the employee pairs, the details of the

unallocated task. This includes experience, nature of work, location of work, problem domain and the average time required. This is run against the historical data that was used in training the model. The data used from the HR and the RT datasets include the time taken for task accomplishment, the level of education, employment status, the age, the experience level, gender, marital status and also the religion. The output for the regression model will be the ranking of paired teams against time taken.

This output is displayed to the supervisor having details of the task with the best two paired teams. The supervisor can decide to choose who to work on the task of the two teams or can choose the best of the two. Supervisors decision can be based on the availability of the team members. On the same interface, the employee will view the task given and who the he/she will work with.



Chapter 3: Research Methodology

3.1 Introduction

Research methodology suggests the philosophy of process construction used to produce the hypothesis that is the methodological structure through which the analysis is performed (Remenyi et al. 1998). It lays forth the criteria for coordinating, preparing, developing and performing studies. Methodological decisions are dictated by the study method that the researcher practices. The research paradigm not only directs the use of data collection and interpretation approaches, but also the preference of opposing theorizing approaches (Sayer, 1992).

This Chapter reviewed the research design, tools and methodology used for the development of the model and finally the system design method adopted. For this research, mixed research method was used. This entails the use of qualitative and quantitative research methods. The quantitative aspect came into play through measuring accuracy of the proposed model. The qualitative aspect was applied in the process of proving the concept that the application of the proposed solution does have a direct positive correlation with the research problem.

3.2 Research Design

The research design is a work plan detailing how the research is to be undertaken, type of data to be collected, tools and techniques that will be employed to obtain data and the method to be used (van Wyk, 2012). The research was a mixed method research whereby it combined elements of qualitative and quantitative research approaches. This took advantage of the strengths of either approach while compensating for their respective constraints. Qualitative research was used to find out whether there was correlation of the model as the solution to the current situation. Since this was an explanatory kind of research whereby the proposed model was to solve the identified problem, literature review was used to gain valuable insight to the phenomenon under investigation. Quantitative research was used to measure the accuracy of the model.

3.2.1 Location of the study

The research was conducted at Strathmore University ICT services department, a private local university in Kenya. The choice for the University and the department is that the ICT services team have a current task allocation system, that offered a good research for the study. Additionally, issues that had arisen with the increase in the university population and the small ICT team, made it a preferable study setting.

3.2.3 Data Collection and Procedure

Secondary sources of data were used in this study. This made use of the existing databases, that is the RT ticketing system historical data and the HR system data.

3.3 Model Development

To develop the proposed model, the following steps were followed:

- i. Obtaining data
- ii. Pre-processing of data
- iii. Development of the model
- iv. Validation of the model

3.3.1 Obtaining data

The data that was used for the input of the machine learning model was obtained from the RT Ticketing System and from the Strathmore University HR database. This data was in the form of CSV files, which was downloaded from the two databases as zip files.

3.3.2 Pre-processing of data

The data that was obtained was pre-processed to remove duplicates and gaps. Talend Studio an open source data integration tool was used to integrate data from the two sources: RT Database and HR database. Talend Studio aided in extracting, transforming and loading the data from the two sources. which is an open-source Python Library was used. Duplicates and gaps were removed, and a complete data set was generated. This was used for training, testing and validation of the model.

3.3.3 Development of the model

The model was developed using SciKit Learn and pandas which are open-source Python libraries.

3.3.4 Validation of the model

Precision was used to validate the model since the proposed model is a regression model that is used to regress the time taken to complete an allocated task.

3.4 System Development Methodology

The system development methodology that was used was system prototyping. System prototyping was used to develop the system which allows for changes and refinements in quick succession, throughout the process. System prototyping is acknowledged as excellent choices when deadlines are limited, as they best allow the project team to change the system functionality based on a particular delivery date (Dennis et al., 2012). Deliverable of the thesis is a working prototype proving a concept that task allocation can be improved using Artificial Intelligence. The general flow of system prototyping is as illustrated in Figure 3.1.

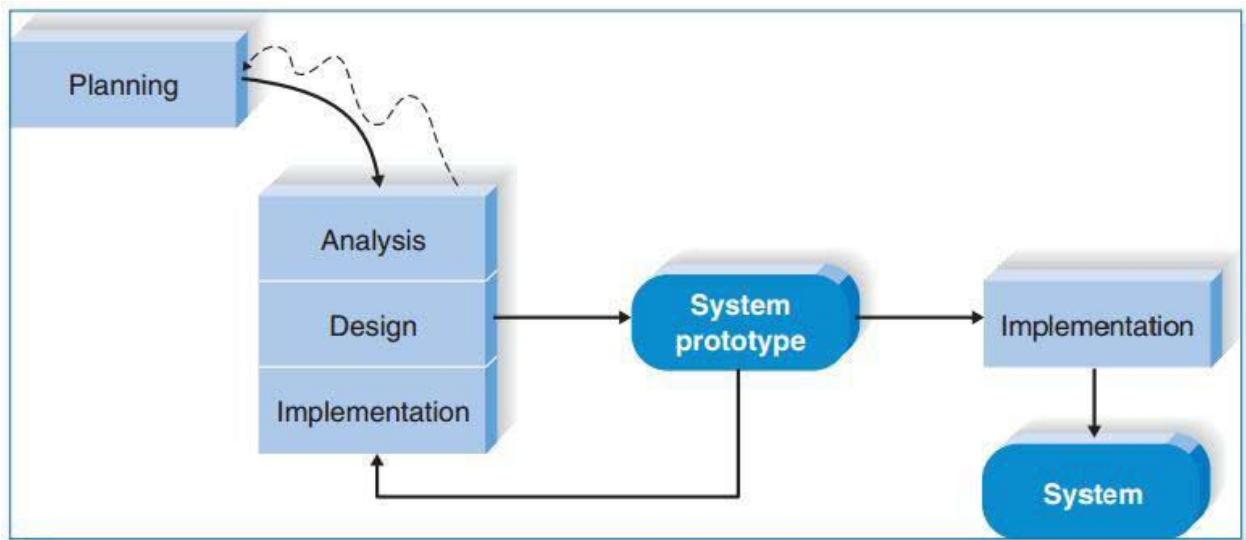


Figure 3.1: System Prototyping Development Methodology (Alan, et al., 2012)

System prototyping operates by carrying out stages of analysis, design and implementation at the same time in order to easily create a streamlined version of the proposed system and to offer review and input to users. After receiving feedback from the system's intended users, the developers are tasked with the task of re-analyzing, redesigning and re-implementing a second prototype that corrects bugs and adds more functionality. This cycle as outlined by Dennis et al. (2012) continues until the analysts, users, and sponsors agree that the prototype provides enough functionality to be installed and used in the organization.

3.5 System Design

In this study, system design involves the design of how data will flow through the model, how the users interact with the model, the sequence of activities through the model and the

different classes implemented. System design is very important as it will aid in outlining the different parts of the proposed model in solving the problem as indicated in the chapter 1.

Object Oriented analysis and design was used for the system design which is based on UML specification which offers an understandable model that reduces the system's complexity. According to Booch et al. (2007), object-oriented analysis and design is the method that leads us to an Object Oriented decomposition whereby object-oriented design uses a notation and process for constructing complex software systems and offers a rich set of models with which we s reason about different aspects of the system under consideration.

They go further in stating that Object-oriented design is a design approach that incorporates the object-oriented decomposition process and a language designed to represent both conceptual and physical as well as static and dynamic representations of the system under design. OOAD was selected as the system design as compared to other system designs because it could simply our design into small modules that can be understood by users and others who would like an understanding of the same. OOAD provides clarity and understanding and provides concurrent model use by multiple users (Booch et al., 2007). This enabled the visualization of the model in a way that users will understand the step by step process.

Some of the UML diagrams that were used include, the data flow diagrams, use case diagrams, system sequence diagram and collaboration diagrams, class diagram and the Star Schema.

3.6 Research Quality

A comparison between the findings and the reality was performed to verify the conclusions obtained from this study. Unusual responses have been removed which extend to be outliers. This form of validation is called triangulation of the data. Data triangulation enables the collection of data from different sources to notify or check one accurate body of data. In addition, the prediction enrollment model's accuracy was measured in order to be sure of the reliability level of the proposed solution.

3.7 Ethical Considerations

The study required information of the participants that is known only to them and the HR Department. The participants were assured that their information was confidential and anonymous

and was only used for academic purposes and only for the purpose of the research. A letter of ethical approval was sought from the Strathmore University Institution Review Board.

Confidentiality and anonymity was upheld through the removal of identifying information from the individual data, securely storing the electronic data and properly dispose or delete the study data.



Chapter 4: System Analysis and Design

4.1. Introduction

In this section, we outline the design architecture of the task allocation prediction model. This architecture will follow from the conceptual model shown in Figure 2.3. This section covers the interaction between the users and the model, the components of the model and the interaction between the various components of the model. The system design and architecture are represented using Unified Modeling Language (UML) diagrams.

4.2. Requirements Analysis

The system requirements analysis aims at determining the expectations of the users who will interact with the system being built. It provides an in-depth analysis of the requirements achieved by the model with respect to the study's primary objectives. The requirements that were gathered for this study can be categorized into three sections; the functional requirements, non-functional requirements and usability requirements.

4.2.1. Functional Requirements

- i. The model should allow a user to input csv data. Any other form of data should be rejected.
- ii. The model should accept data in the form of task details and employee match details.
- iii. The model should output best match to a task.
- iv. The model should output the best shortest time.
- v. The predicted match should be valid based on the input given from the user.

4.2.2. Non-Functional Requirements

4.2.2.1. Usability Requirements

The model is intended to be a representation of a system that will be used by the ICT department. The main users will be ICT employees and their supervisor. The model should therefore be simple and accurate to enhance user experience. Its prediction should have the least time in learning as this will directly affect services provided by the ICT team to their clients.

4.2.2.2. Reliability Requirements

An accurate matching should always be given by the model.

4.2.2.3. Supportability Requirements

The model should be accessible via the command line prompt. The system should be accessible across all operating systems.

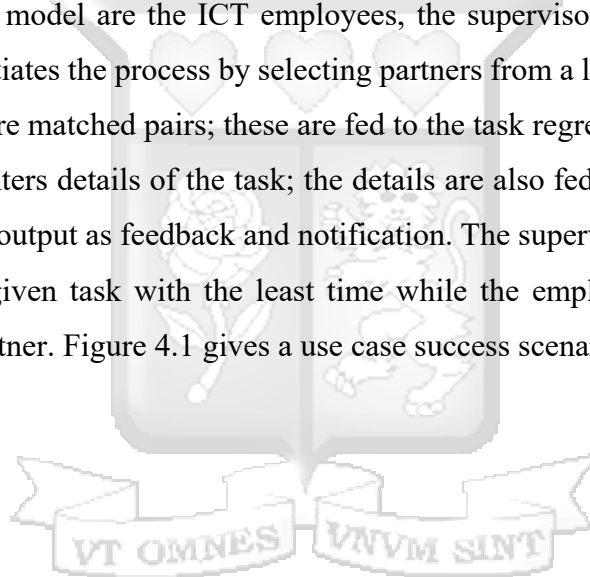
4.3. System Architecture

The system architecture captures the proposed model's design including the minimum user requirements that are achieved through the model or the resultant system it supports. The proposed architectural analysis addresses how different components of the model supports its functionality. The proposed system comprises of the User interface, and the task regression model for prediction.

4.4. Use Case Diagram

The actors in the model are the ICT employees, the supervisor, and the task regression model. The employee initiates the process by selecting partners from a list who they would like to work with. The outputs are matched pairs; these are fed to the task regression model.

The supervisor enters details of the task; the details are also fed to the model. The model learns the data and gives output as feedback and notification. The supervisor receives feedback as the matched pairs to a given task with the least time while the employee gets a personalized allocation of task and partner. Figure 4.1 gives a use case success scenario of the model.



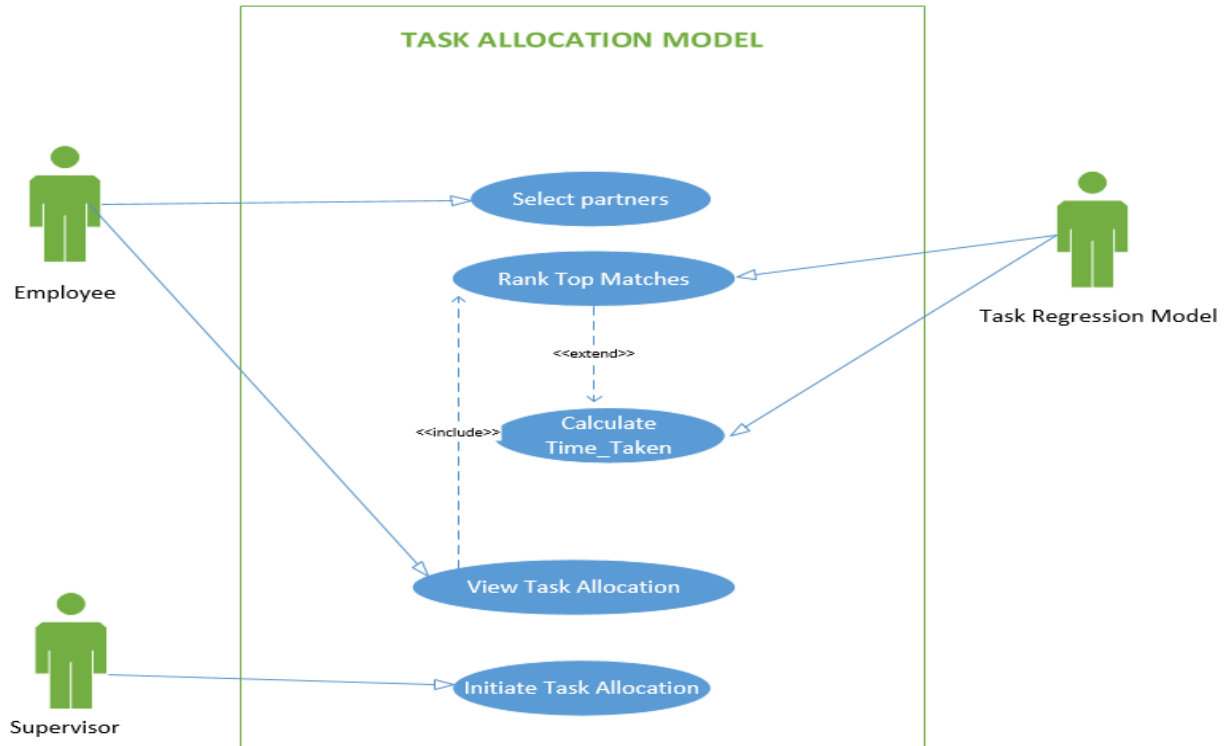


Figure 4.1: Use Case Diagram

Use Case Flows:

Table 4.1: Partner Selection

Use Case: Selection of Partner	
Primary Actors: Employee	
Precondition: Employee data	
Post-condition: Partners selected for each employee with employee details	
Main Success Scenarios	
Actor Intention	System Responsibility
1. Employee logins to the system	
	2. Allows login in.
3. Employee selects partners he prefers working with	
	4. Saves the selections
	5. Displays the selection

Table 4.2: Ranking the matches and Calculating Time Taken

Use Case: Ranking the matches and Calculating Time Taken	
Primary Actors: Task Allocation Regression Model	
Precondition: Task details and top matched pairs data available	
Post-condition: Ranking of paired teams against Time Taken	
Main Success Scenarios	
Actor Intention	System Responsibility
1. Input layer receives details of unallocated task	
2. Input layer receives details of selected pairs	
3. Hidden layer learns data	
4. Hidden layer ranks the paired teams against Time Taken	
	5. Display paired rankings against time taken

Table 4.3: Initiating Task Allocation

Use Case: Initiating Task Allocation	
Primary Actors: Supervisor	
Precondition: Issues raised as tasks with details	
Post-condition: Provide details of unallocated tasks	
Main Success Scenarios	
Actor Intention	System Responsibility
1. View unallocated task details	
2. Provide details of unallocated tasks	
	3. Displays unallocated task details

4.5 System Sequence Diagram

Figure 4.2 shows the sequence diagram for the model. An employee enters details of his preferred partners, these are partners that he would like to work with. The system returns and displays the employee's selection. The selection which has details of the employee and choice of partner is matched to produce a stable matching. This is then passed to the task regressor as one of the inputs. During this time, the supervisor initiates task allocation by entering details of the task. This forms the second part of the inputs for the task allocation regressor.

The system then predicts the output which is the ranking of the matches against time taken to accomplish a given task. This result is given to the supervisor and also the employee. Employee receives an email notification of the task and the partner they are going to work with to accomplish the task.

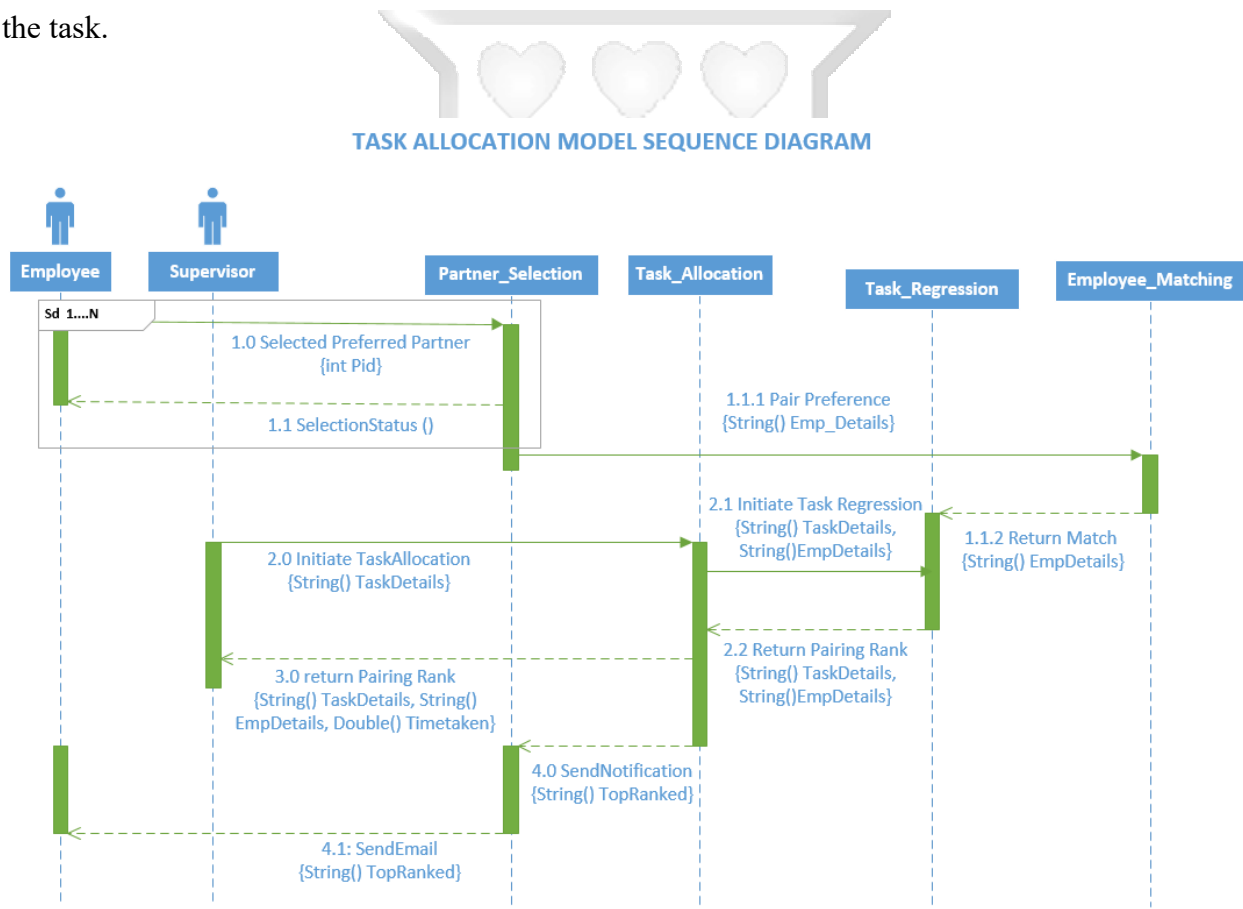


Figure 4.2: System Sequence Diagram

4.6 Class Diagram

Class diagrams represent the objects in the system, the attributes, the operations and the relationships among them. The main classes in this model include the employee, preference, task,

allocation, match, regressor and notification classes. Figure 4.3 shows the class diagram of the proposed model.

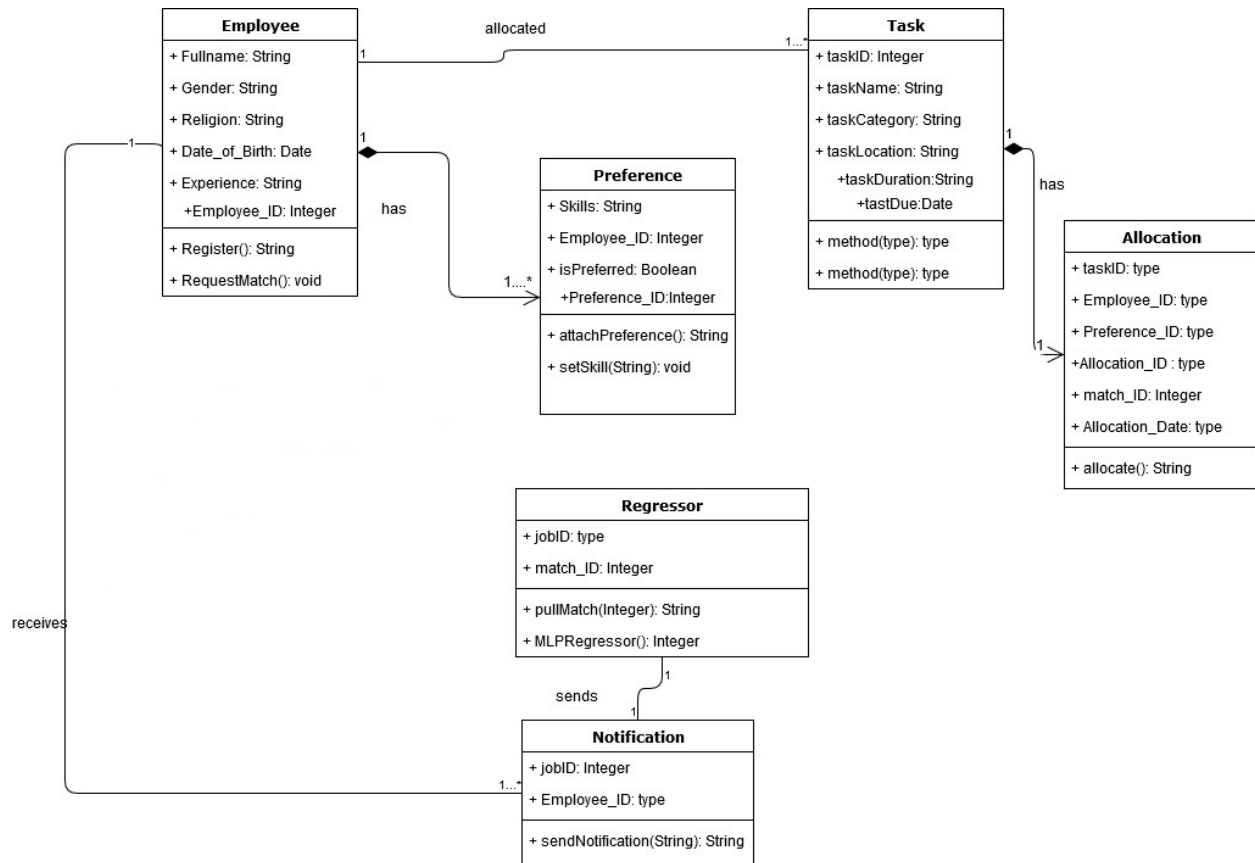


Figure 4.3: Class Diagram

4.7 Data Flow Diagram

Data flow diagrams are used to graphically represent the flow of data in a system. Data flow diagram has been used to show how data that has been input is transferred, manipulated, stored and distributed. Entities identified include the employee, supervisor and the task allocation regressor. Processes include initiate partner selection, match pairs, predict time taken, and send notification. Finally, data stores identified include the employee, task and allocation data stores. Figure 4.4 shows the data flow diagram of the proposed model.

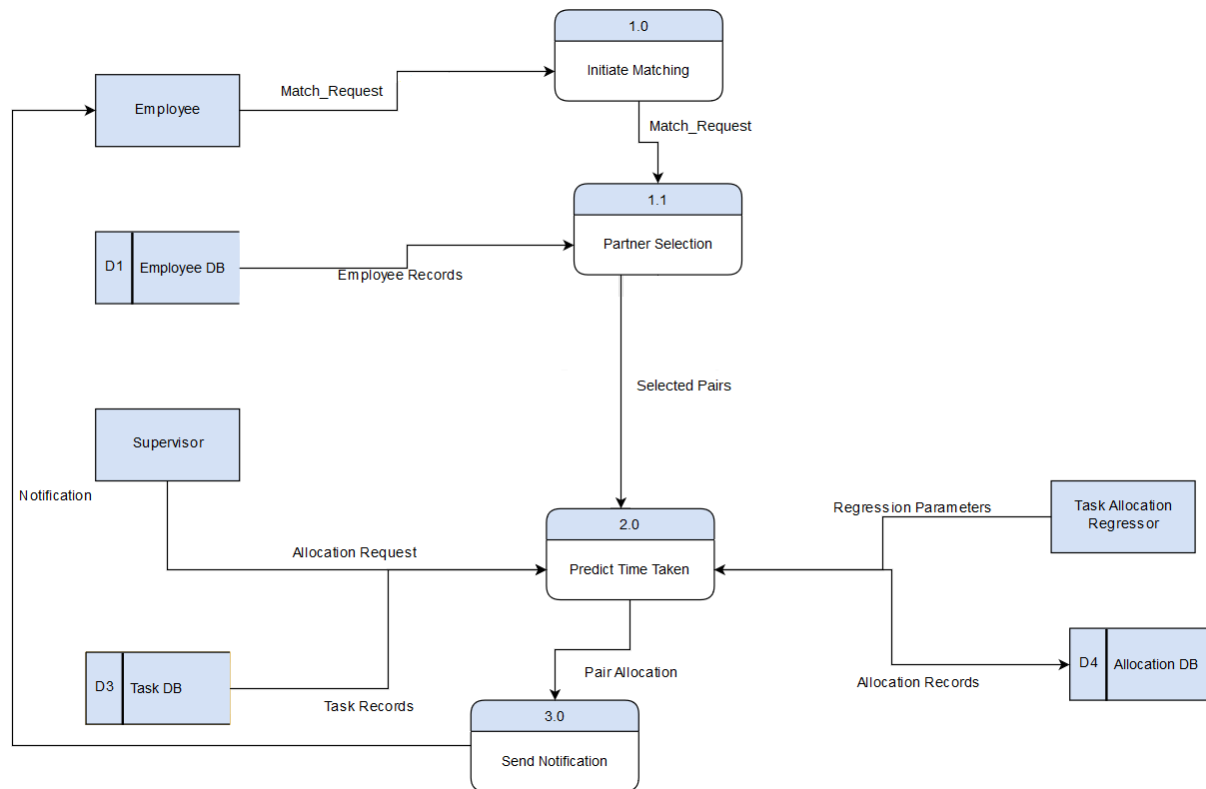


Figure 4.4: Data Flow Diagram

4.8 Database Schema

The database schema was used to define the entities and the relationship of the database in this model. The database contains a total of 5 tables that have been normalized. These include employee table, employee type table, allocation table, preference table and finally the task table. Figure 4.5 shows the database schema for the proposed model.

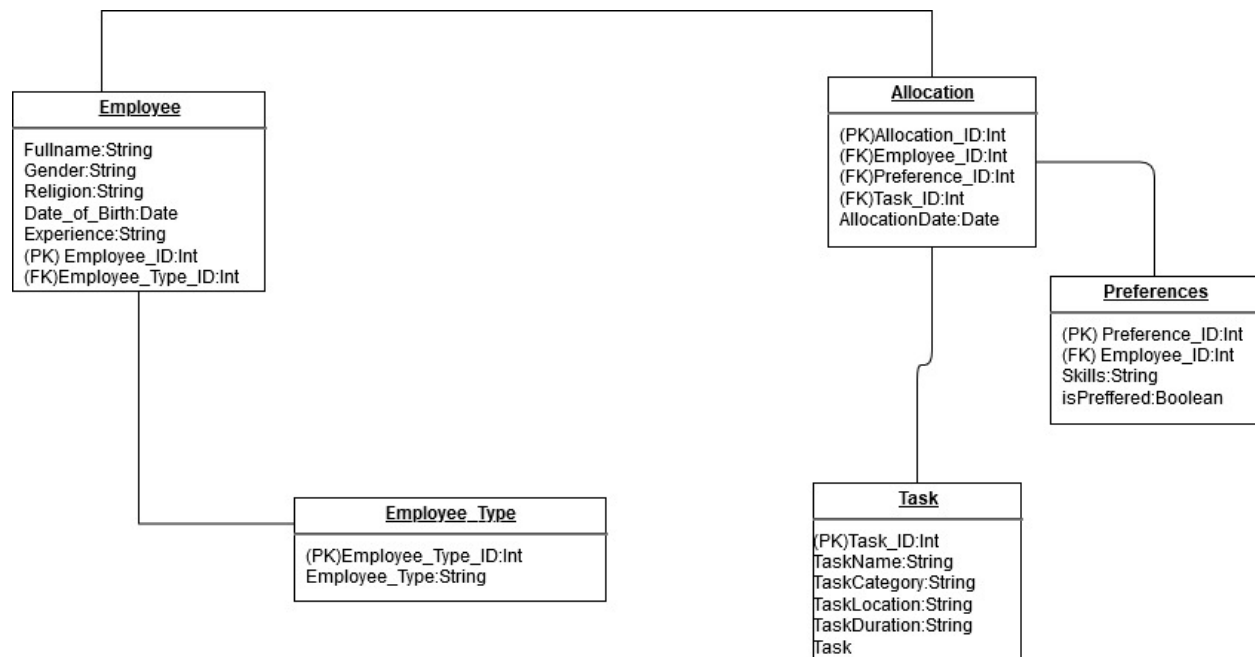
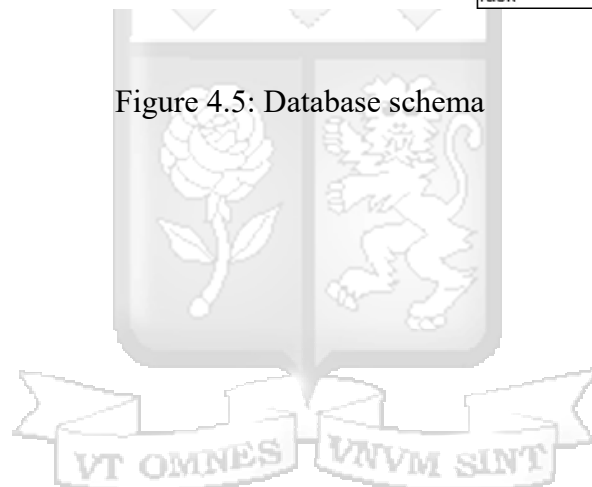


Figure 4.5: Database schema



Chapter 5: Implementation and Testing

5.1. Introduction

The prototype was developed through the use of neural network multilayer perceptron which is a class of feedforward artificial neural network. The model was implemented in two phases:

- a) Pairing Phase - this relied heavily on user preference.
- b) Time regression Phase - finding the pair with the least time taken. This was also implemented using Python version 3 (Artificial Neural Network).

Data was entered through the command line prompt, this data consisted of the employee preference on who to work and also data consisting of the task that needs to be worked on. This formed the input to the neural network. Historical data was also used which was stored as a pickle file and also used as an input to the neural network. Data was split into training, testing and validation set. The training set was fed into the neural network model.

5.2 Model Components

The model consisted of the data extraction cleaning and merging component, system components and Neural Network Components.

5.2.1 Data Extraction Cleaning and Merging Component

Since the neural network only accepts numerical values for purposes of formulating an accurate prediction model, data from the two sources, that is the HR data and RT data were extracted, cleaned and merged. This was achieved through the use of Talend Studio an open source data integration tool. Data from the HR database and RT database were first extracted as shown in Figure 5.1 and 5.2.

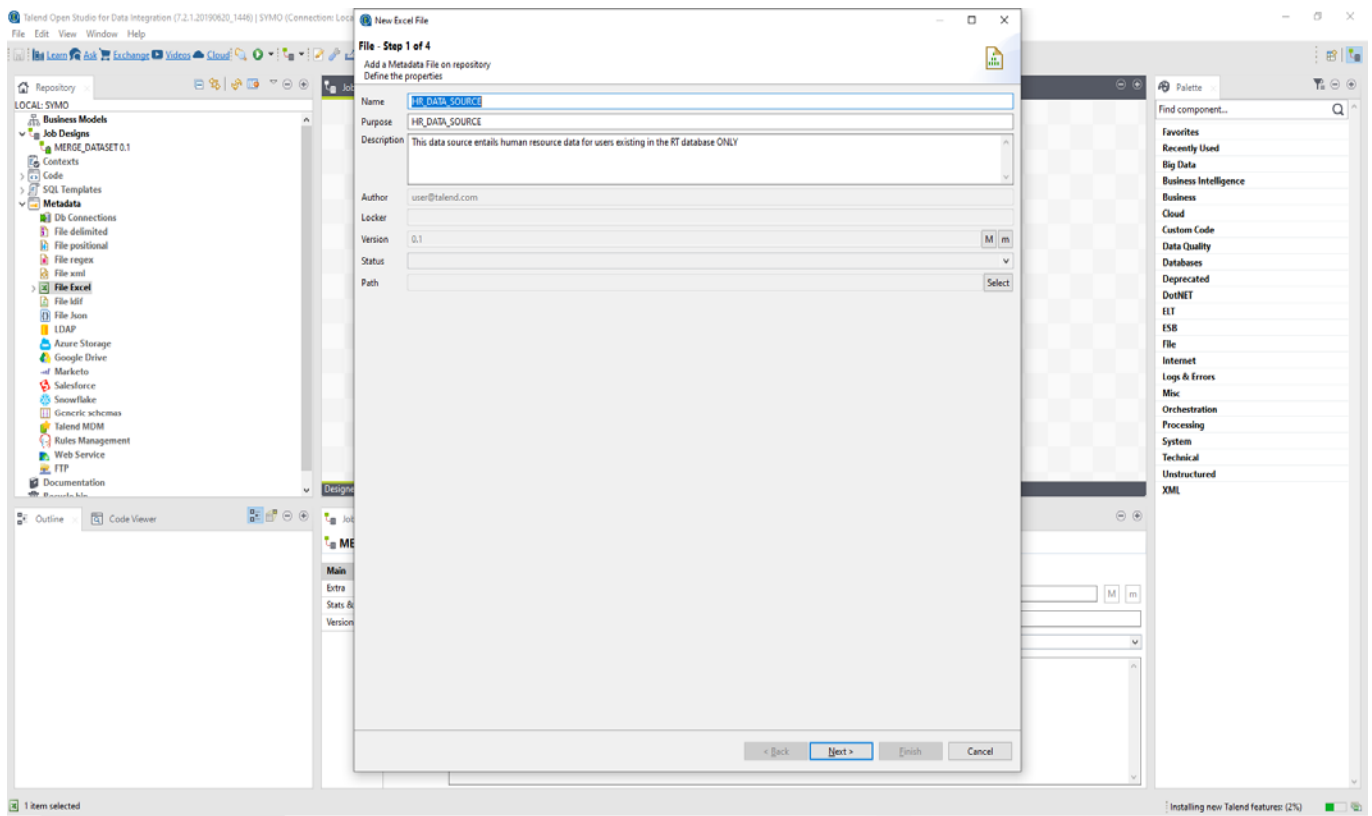


Figure 5.1: HR data Extraction Screenshot

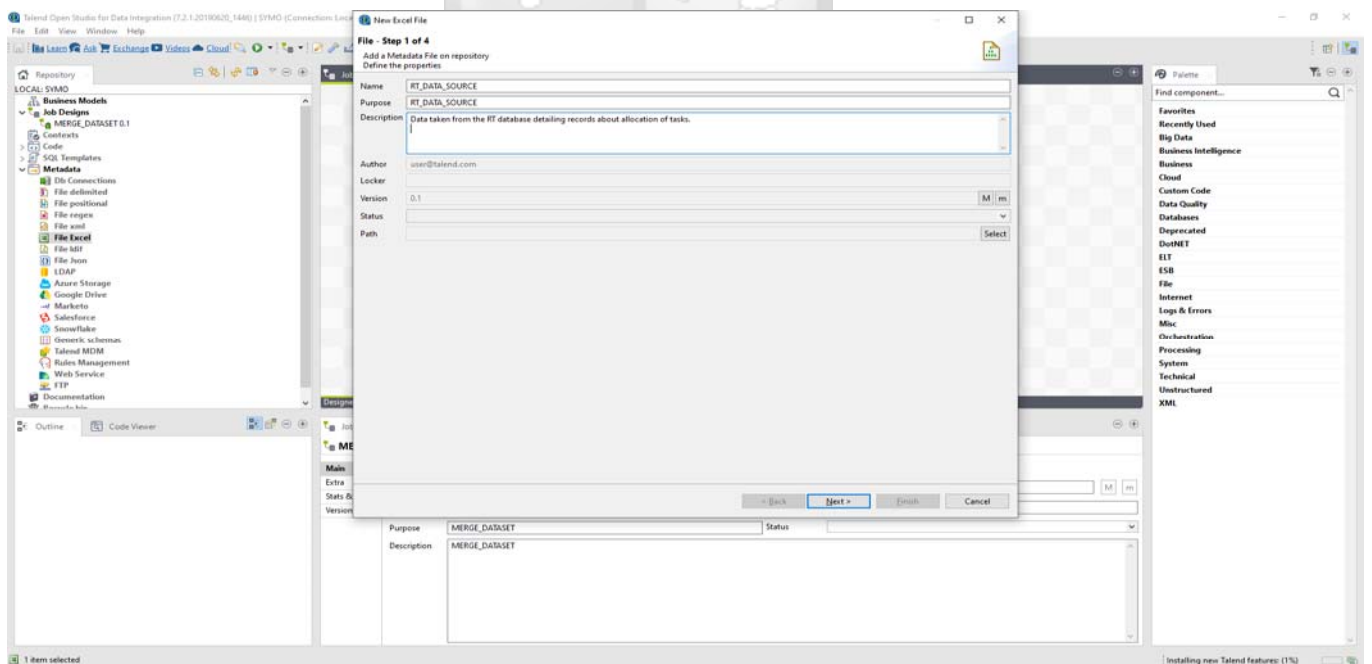


Figure 5.2: RT data Extraction Screenshot

Upon extraction, data was then merged together by removing the repeated rows in both the two datasets. Merging entailed the connection of the two datasets to one having similarities in the attributes. The attributes include: staff ID, time taken, gender, marital status, religion, meal allowance status, level, employment status, years in the organization and finally the age. This process is also known as transformation. Figures 5.3 and 5.4 depict the extraction process which is the removal of the repeated rows while Figure 5.5 shows how joining was done.

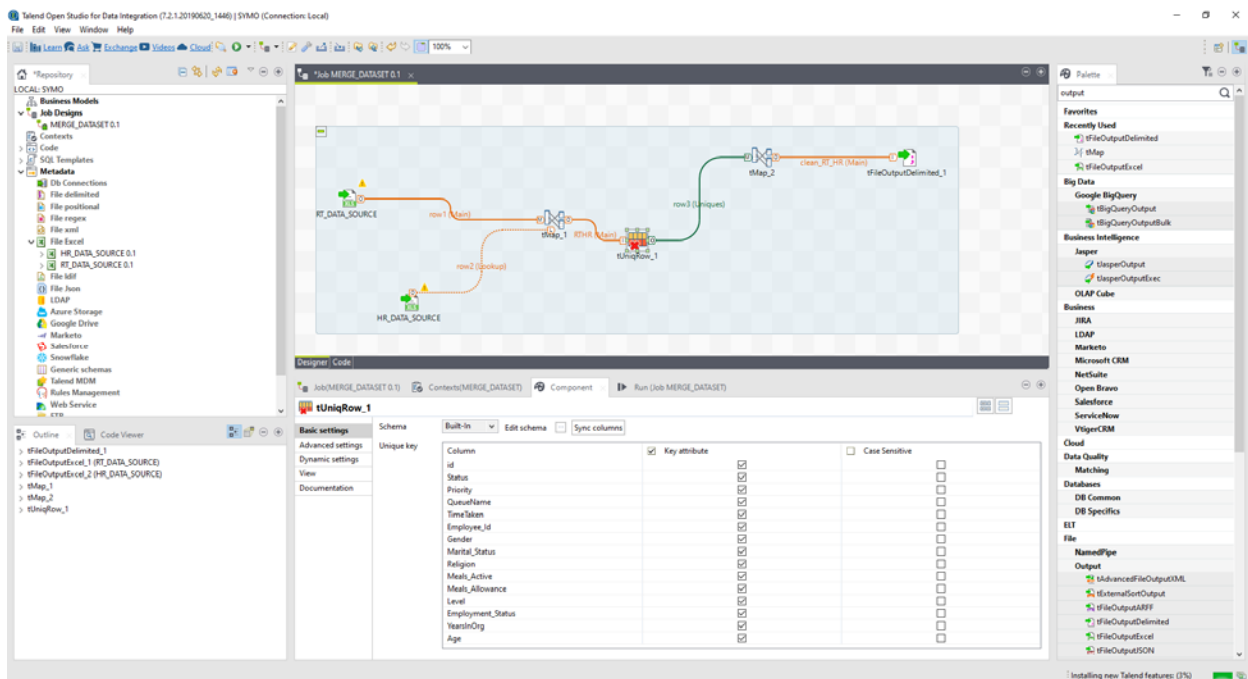


Figure 5.3: Extraction Removing Repeated Rows

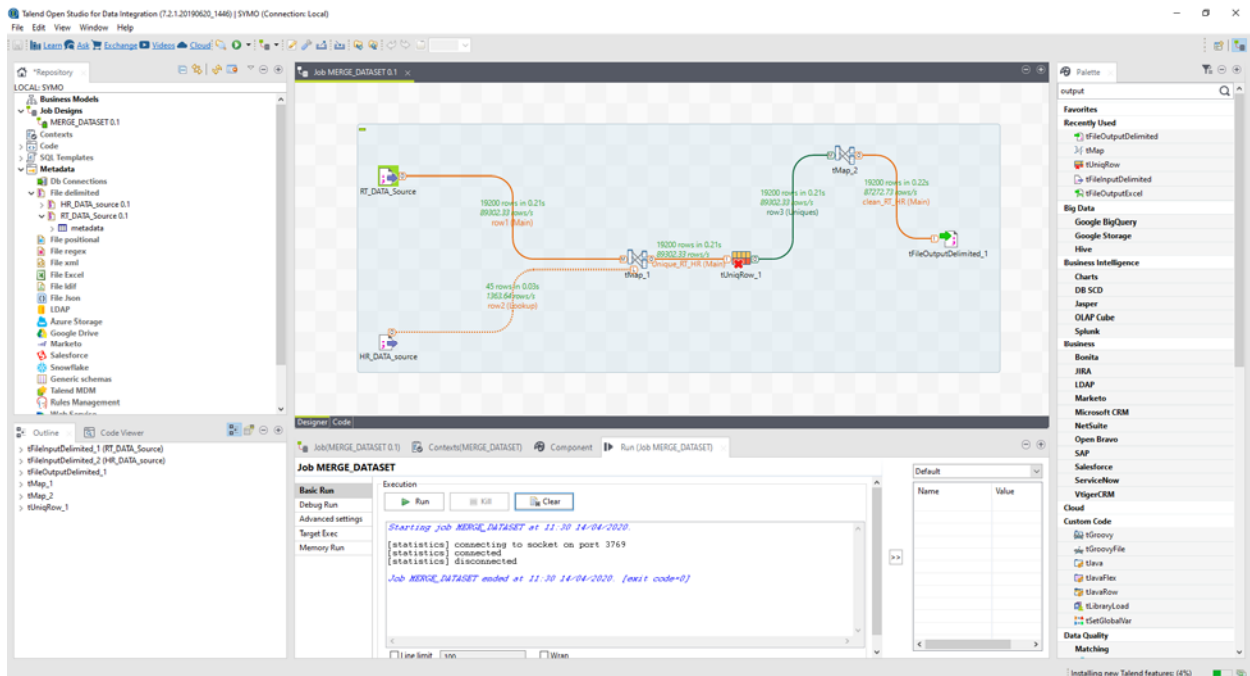


Figure 5.4: Complete Extraction

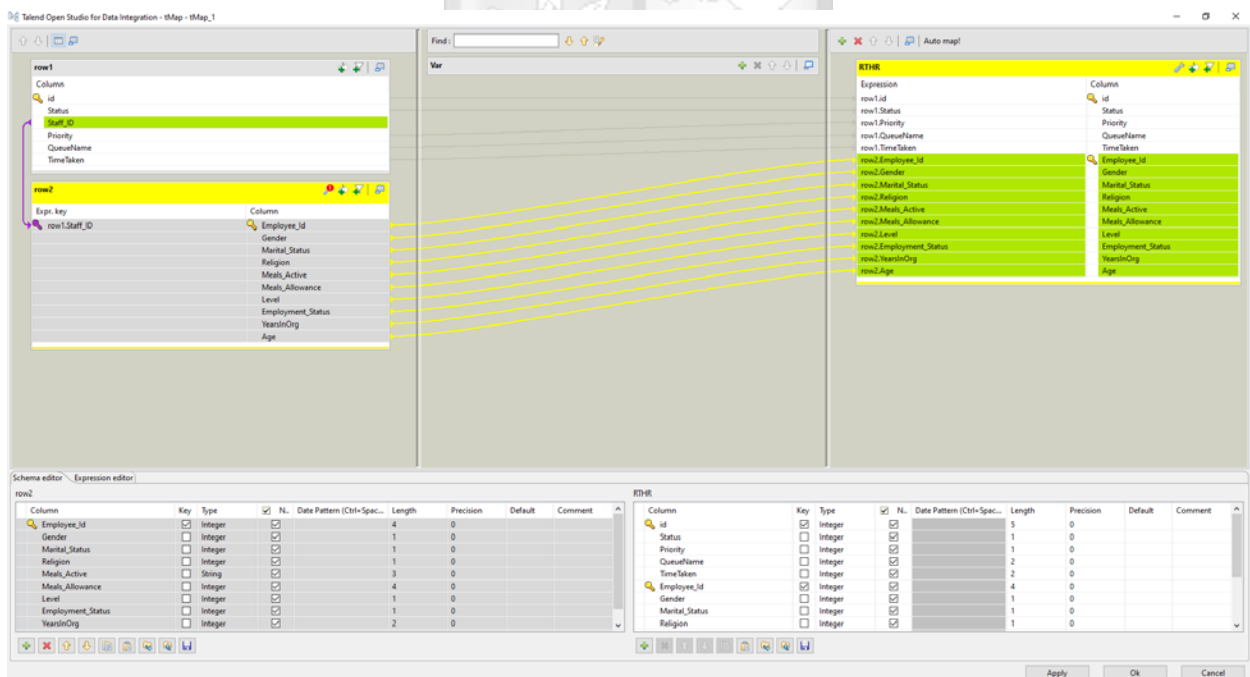


Figure 5.5: RT and HR datasets Joining

The output is loaded to file awaiting testing or training. Figure 5.6 is a screenshot of the merged HR and RT datasets in CSV format.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	id	Status	Staff_ID	Priority	QueueName	TimeTaken	Gender	Marital_Status	Religion	Meals_Active	Meals_Allowance	Level	Employment_Status	YearsInOrg	Age	
2	28513	2	759	3	16	18	2	1	3	1	4000	1	3	14	50	
3	34020	2	759	1	22	23	2	1	3	1	4000	1	3	14	50	
4	34059	2	759	1	15	34	2	1	3	1	4000	1	3	14	50	
5	34063	2	759	2	15	50	2	1	3	1	4000	1	3	14	50	
6	34074	2	759	2	22	58	2	1	3	1	4000	1	3	14	50	
7	34078	2	759	2	1	43	2	1	3	1	4000	1	3	14	50	
8	34079	2	759	1	1	37	2	1	3	1	4000	1	3	14	50	
9	34080	2	759	2	1	56	2	1	3	1	4000	1	3	14	50	
10	34081	2	759	1	1	60	2	1	3	1	4000	1	3	14	50	
11	34082	2	759	1	1	12	2	1	3	1	4000	1	3	14	50	
12	34129	2	759	3	15	53	2	1	3	1	4000	1	3	14	50	
13	34155	2	759	2	0	49	2	1	3	1	4000	1	3	14	50	
14	34209	2	759	3	1	16	2	1	3	1	4000	1	3	14	50	
15	34210	2	759	1	1	24	2	1	3	1	4000	1	3	14	50	
16	34211	2	759	2	1	59	2	1	3	1	4000	1	3	14	50	
17	34225	2	759	2	22	36	2	1	3	1	4000	1	3	14	50	
18	34229	2	759	1	15	44	2	1	3	1	4000	1	3	14	50	
19	34234	2	759	1	1	35	2	1	3	1	4000	1	3	14	50	
20	34237	2	759	3	15	19	2	1	3	1	4000	1	3	14	50	
21	34255	2	759	3	15	21	2	1	3	1	4000	1	3	14	50	
22	34263	2	759	2	15	38	2	1	3	1	4000	1	3	14	50	
23	34275	2	759	3	22	14	2	1	3	1	4000	1	3	14	50	
24	34299	2	759	3	22	44	2	1	3	1	4000	1	3	14	50	
25	34307	2	759	2	22	17	2	1	3	1	4000	1	3	14	50	
26	34325	2	759	3	0	35	2	1	3	1	4000	1	3	14	50	
27	34327	2	759	2	0	56	2	1	3	1	4000	1	3	14	50	
28	34334	2	759	3	22	30	2	1	3	1	4000	1	3	14	50	
29	34342	2	759	1	22	35	2	1	3	1	4000	1	3	14	50	
30	34343	2	759	1	22	41	2	1	3	1	4000	1	3	14	50	
31	34355	2	759	3	22	43	2	1	3	1	4000	1	3	14	50	
32	34377	2	759	3	22	30	2	1	3	1	4000	1	3	14	50	
33	34381	2	759	2	18	19	2	1	3	1	4000	1	3	14	50	
34	34386	2	759	2	15	53	2	1	3	1	4000	1	3	14	50	
35	34394	2	759	3	22	47	2	1	3	1	4000	1	3	14	50	
36	34395	2	759	3	22	43	2	1	3	1	4000	1	3	14	50	
37	34406	2	759	2	22	23	2	1	3	1	4000	1	3	14	50	
38	34420	2	759	1	18	38	2	1	3	1	4000	1	3	14	50	

Figure 5.6: Merged HR and RT datasets in CSV

5.2.1 System Components

Command line based approach was used to prove the model works. On the interface, the first information that pops up is a list of all the different task categories with their Ids as shown on Figure 5.7. This is where the supervisor selects the category of task that needs to be done. After selection, a list of employees who can perform the task is generated shown in Figure 5.8. The employee comes in here; he selects other employees he would like to work with as shown on Figure 5.9. The employee has three options of to choose from. After this selection is done. The next phase of the interface is to show the selection/workings and on the final phase is the output. This shows the pairs that can perform the task and the time taken. The pairs are the selections that were made earlier. The first two pairs are the output needed. Figure 5.10 and 5.11 depict this.

```

-----
INTELLIGENT TASK ALLOCATION MODEL (Simon Kipekmboi)
-----

Succesfully Connected to Manifest!
-----

(24, ' Printing Pin')
(1, 'Account Closure')
(2, 'AMS Management')
(3, 'Audiovisuals')
(4, 'Communications')
(5, 'Conferences & Events')
(6, 'Conferences & Events')
(6, 'Data Requests')
(7, 'Default')
(8, 'Elearning')
(9, 'Hardware')
(10, 'Movement')
(11, 'Network')
(12, 'Phones')
(13, 'Printer Management')
(14, 'Printing A/Cs and Credits')
(15, 'Printing Top UP')
(16, 'Purchases')
(17, 'SBS Support')
(18, 'Security')
(19, 'Software')
(20, 'Staff Training')
(21, 'Student Emails/Domain passwords.')
(23, 'Student Laptop')
(24, 'Systems Support')
(25, 'UPS')

(25, 'UPS')
(26, 'User Account Management')

-----
CHOOSE THE TASK CATEGORY ID FROM THE LIST ABOVE:
-----
15

```

Figure 5.7: List of task categories.


```

-----
26
(759, 'aamutabi (Amos Amutabi)')
(3060, 'AOtonglo (Allan Otonglo)')
(1024, 'bogutu (Benson Ogutu)')
(1116, 'CKiilur (Charles Kiilur)')
(1698, 'gmunyi (George Munyi)')
(3128, 'INato (Ian Nato)')
(2888, 'JKimotho (Joyce Kimotho)')
(2903, 'LKasera (Lawrence Kasera)')
(1221, 'mkihara (Maria Awuor)')
(1824, 'MMbabu (Mike Mbabu)')
(2248, 'mndeto (Martin Ndithi Ndeto)')
(305, 'Msiro (Michael Siro)')
(299, 'MSuka (Matthews Suka)')
(2713, 'NMugo (Newton Mugo)')
(2069, 'nwaweru (Ndirangu Waweru)')
(1717, 'skipkemboi (Simon Kipkemboi)')
(315, 'smomanyi (Stephen Momanyi)')
(2628, 'tomogo (Thomas Omondi Omogo)')
CHOOSE ANY THREE PEOPLE YOU WISH TO WORK WITH FROM THE TOP PEOPLE (KEY IN ID)
-----
-----

```

Figure 5.8: List of employees who can perform the task.

```

(1024, 'bogutu (Benson Ogutu)')
(2888, 'JKimotho (Joyce Kimotho)')
(1221, 'mkihara (Maria Awuor)')
(1824, 'MMbabu (Mike Mbabu)')
(305, 'Msiro (Michael Siro)')
CHOOSE ANY THREE PEOPLE YOU WISH TO WORK WITH FROM THE TOP PEOPLE (KEY IN ID)
-----
-----
ID FOR 1st Preference?:
-----
1824
-----ID FOR 2nd Preference?:
-----
305
-----ID FOR 3rd Preference?:
-----
1024
(2, 1, 4, 4000, 3, 4, 11, 36, 1, 16)

```

Figure 5.9: Selection of preferred employee.

```

(2, 1, 4, 4000, 3, 4, 11, 36, 1, 16)
(2, 1, 4, 4000, 3, 4, 11, 36, 2, 8)
(2, 1, 4, 4000, 3, 4, 11, 36, 2, 3)
Succesfully Connected to Manifest!
-----

```

Figure 5.10: Selection workings

```

-----
Name: ('MMbabu (Mike Mbabu)',)
Time taken(Minutes):
[60.]
Verdict :--[PROBABLE!!--
-----
Name: ('Msiro (Michael Siro)',)
Time taken(Minutes):
[24.60291161]
Verdict :--[PROBABLE!!--
-----
Name: ('bogutu (Benson Ogutu)',)
Verdict :--[DISQUALIFIED!!--
-----

```

Figure 5.11: Output best pairs with time taken

5.2.3 Neural Network Components

The components for the multilevel perceptron are as follows:

5.2.3.1 Input Layer

This is the first layer in the neural network. The input layer consisted of 8 nodes equal to the number of attributes chosen for the model. These attributes included the experience, which is derived from the HR data, which is the number of years one has been in the organization; the nature of work derived from the employee, which is the job description of the employee; the location which is derived from the task in hand; the problem domain which is also derived from the task where it is the category of the issue; the average required time also derived from the task which is the average time required to accomplish the task and finally the top 3 selection preferences from the employee.

5.2.3.2 Hidden Layer

The model developed used 8 nodes and three layers each having the 8 nodes. This was picked from the input layer.

5.2.3.3 Output Layer

This layer serves as an output from the neural network, it is the last layer of the neural network. The model has one output node indicating time taken to resolve the issue of two pairs ranked with the least time being the first.

5.3 Model Implementation

The proposed model was implemented through first fetching the RT and HR data, then normalizing it through the MinimumMaximumScalar which scaled the data before feeding it to the neural network for training. The end result was stored awaiting execution.

5.3.1 Data fetching

Talend Studio an open source data integration tool was used to integrate data from the two sources: RT Database and HR database. Talend Studio aided in extracting, transforming and loading the data from the two sources.

5.3.2 Dataset Description

This is the description of the dataset attributes from the two dataset, i.e. RT and HR databases. featured selection was done and a list of attributes were chosen to be used in the task allocation model. Only data relating to task allocation and identification were used to train, validate and test the model, this ensured clean data. Table 5.1 illustrates the attributes with their numerical values.

Table 5.1: Dataset Description

Variable	Description	Possible Values
ID	Task Details	User Support (1), Network (2) or Systems Development (3)
Status	This indicates whether the task has been resolved or not	Resolved (2) and Not resolved (1)
Queue Name	Nature of work	Network (1), Printing (2), student email and passwords (3), security (4) among others
Priority	This is the urgency of the task or ticket	Urgent (1) or Normal (2)
Staff ID	Identification of Employee	Number of Staff Identification
Level	This attribute shows the education level of the	Internship(1), Temporary Contract (2), Permanent Contract (3)
Time Taken	This attribute shows the time taken for a task to be completed	Time 0-60 Minutes
Years in Organization	This attribute shows the number of years an employee has been in the organization	Experience 0- 30 years
Meals Active	This attribute show whether meal allowance is active	Meal active (1) or Meal Inactive (2)
Meals Allowance	This is the amount that has been allocated for food	Full amount (1) Half amount (2) Or no amount (0)

Gender	This attribute shows whether the person is male or female	Male (1) or female (2)
Marital Status	This shows whether the employee is married or not.	Married (1) or Not Married (0)
Religion	This attribute shows the religion of the employee	Catholic (1), Protestant (2), Muslim (3)
Age	This is the length of time a person has lived	Age is between 19-60 years

5.3.3 Data Normalization

It is better to normalize data in order to have the same range of values for each of the inputs to the ANN model. This guarantees stable convergence of network weights and biases. MinimumMaximumScalar normalization function was used. The range used was between 0 and 1. MinimumMaximumScalar performed well in tuning the model to produce a scalable dataset.

5.3.4 Model Training

The model was trained using Scikit-learn's machine learning regressor. The hyper parameters were to the effect that there were hidden layer: the first hidden layer had 60 nodes, the second had 100 nodes and the third hidden layer 80 nodes. This matrix was deliberately chosen after a number of epochs and hyper parameter adjustments. The activation function used was reLU. This is because it is simple, fast and empirically it seems to work well. It also made it faster for the neural network to attain a convergence.

5.3.5 Storing of Model

Since we are using a command line based approach, the trained model as then stored as an executable python file with the extension pkl. A pkl file is a file created by pickle which is a python module that enables data to be serialized to files. During program runtime they are de-serialized back.

5.4 Software Flow

Command line based approach was used as interface for the employee and the supervisor who were users of the model. The first phase of the interface was the supervisor selecting from a

list of task categories which task was to be worked on. The system saves this selection. This selection is displayed on the interface. The second phase on the interface is for the employee to select his preferred employees. Employees had three options of selecting a preferred partner. The system saved the selection. This selection is displayed on the interface.

The third phase is the workings output whereby it displays the workings and the summary of the task and matches. This is fed to the neural network as input. It consists of the three preferred pairs as selected by the employee and the task details as selected by the supervisor.

Once prediction is done, feedback in the form of ranking of paired teams to a given task with the time taken is given on the prompt on the interface. This feedback enables the supervisor to assign or not assign to an employee the task.

The major challenge with the model creation was the issue of missing data and scanty data. Data like location and subject were missing and in some areas data was scanty. Another Challenge was the operational expenses during the transformation of data, this came due to issues with finding a laptop that can run and analyze the data without slowing down and hanging.

The environment used in the development of this model are as follows:

- i. Windows 10 Pro
- ii. Anaconda platform environment.
- iii. Python 3.8.2 programming language
- iv. Python Library- Numpy
- v. Python Library- Scikit learn
- vi. Python Library- Pandas.
- vii. Pickle
- viii. MySQL
- ix. Talend Open Studio for Data Integration

5.5 Model Architecture

Figure 5.12 illustrates the proposed model for task allocation. The initial steps for the model include data pre-processing where data was normalized using the MinimumMaximumScalar function that scaled data and allowed for numerical data. Training begins when the training set is then passed through the multilayer perceptron. The output for this

is written and stored into an executable file called task_match_Pred.pkl. This file is called up whenever a user needs to make a prediction. When a supervisor enters unassigned task details and needs to predict the pair that will be assigned the task, to avoid overfitting, a criterion is first chosen which is already encoded in the dataset stored that is separate from those used for training and validation.

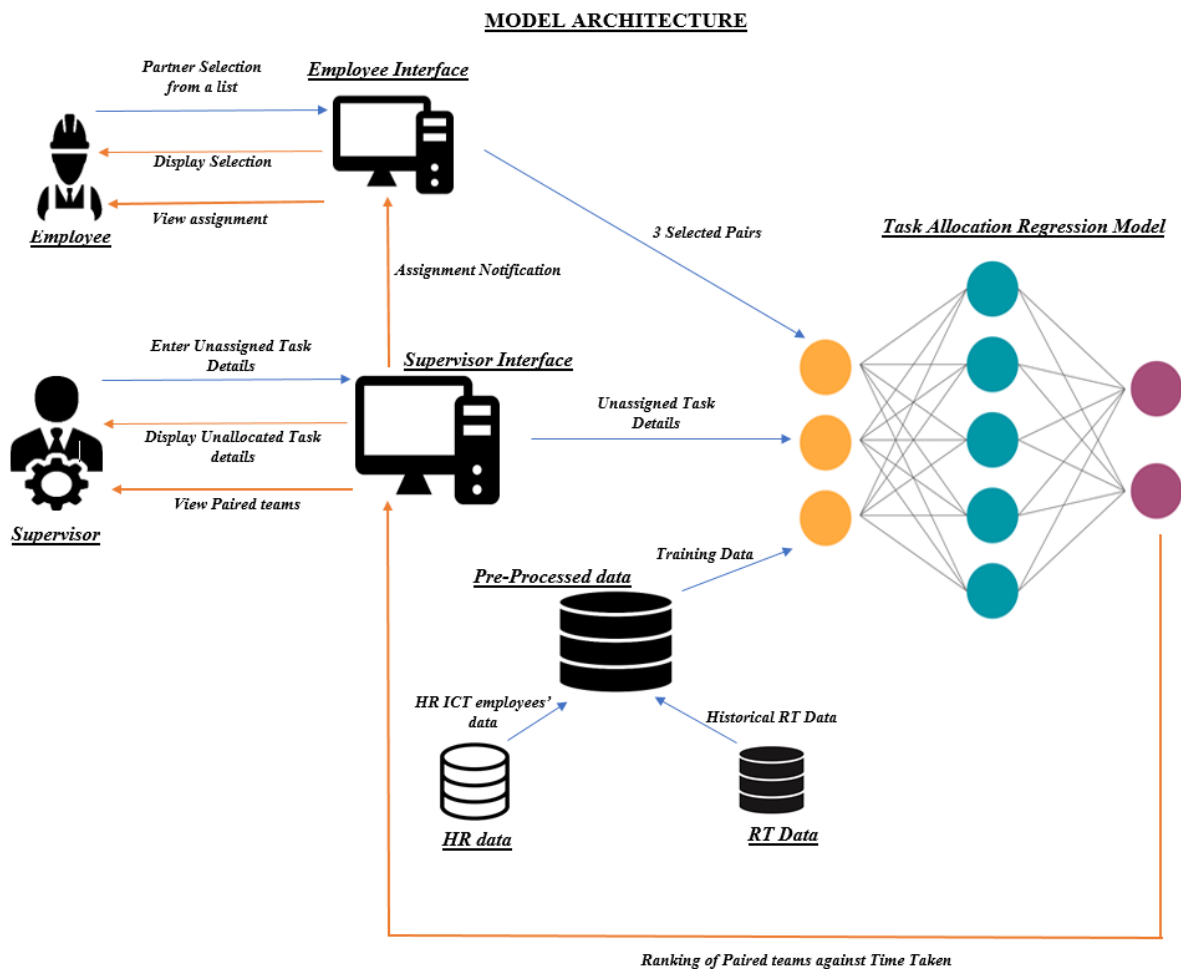


Figure 5.12: The Proposed Model Architecture

5.6 Model Testing

This involves the process of testing of the model in terms of functionality, reliability and performance. Below are the parameters used in Table 5.2.

Table 5.2: Model and System Testing

Test Case	Criteria Check	Priority
Functionality	Does the model allow users to select pairs?	High
Functionality	Does the model allow users to upload unassigned task details?	High
Performance	Does the model take a short time in prediction?	High
Reliability	Does the model consistently give correct matching?	High

5.7 Model and System Testing results

The proposed model performed successful in predicting the matched paired right for a task with the time taken. The model took a relatively shorter time in prediction which was consistence and accurate. The results for the system and model testing results are shown below in Table 5.3.

Table 5.3: Model and System Test Results

Test Case	Test Results	Remarks
Functionality	PASS	The model allows user input
Reliability	PASS	The model take a short time in predicting matches with time.
Performance	PASS	The model is accurate in consistently giving correct matching.

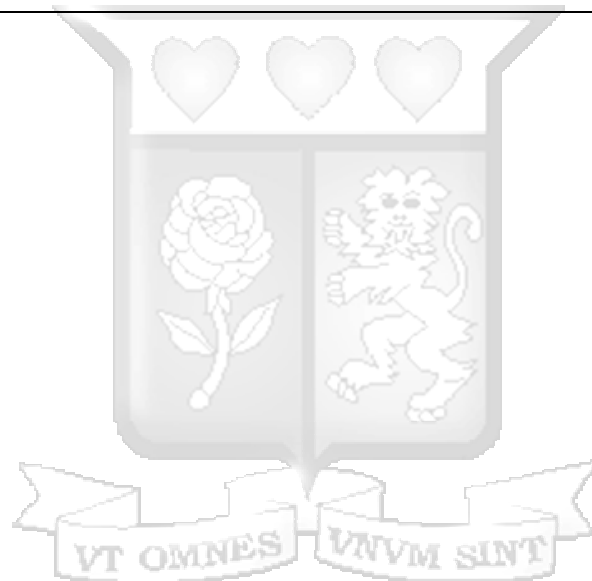
5.8 User Acceptance testing

This testing entailed the checking of whether the objectives of the model were achieved in terms of user experience. Moreover, it checks whether the system meets the specific criteria for which it was made for.

The user was handed the model to test and give feedback on the functionality, reliability and usability of the model with the current system. Table 5.4 shows the test cases that were handled.

Table 5.4: User Acceptance Testing

Test Case	Criteria Check	Priority
Functionality	Does the model allow users to select pairs?	High
Functionality	Does the model allow users to upload unassigned task details?	High
Reliability	How reliable is the proposed model according to the users in performing their tasks?	High
Functionality	Does the model notify users of matched assignments?	High
Acceptability	How likely is the proposed model likely to adopted by the ICT team for efficiency in performing their tasks?	High



Chapter 6: Discussion

6.1 Introduction

This section discusses the testing of the model which focused on the effectiveness, the efficiency, the ease of use, the interactivity, utility and convenience. The model's accuracy level was satisfactory enough as it was able to predict the best pairs for a given task, this made it reliable. This proved the functionality to be effective and efficient.

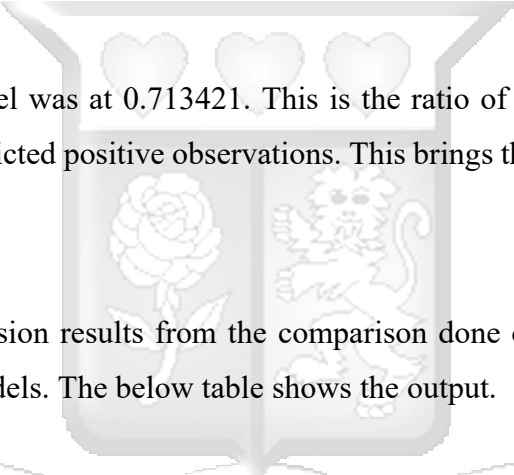
Since the output interface was command line based, it was easy to interact with the model as everything was straight forward for the functionalities. The speed of the model in executing the predictions in shortest time period was made possible by pickle and the use of stub files.

6.2 Model Validation

Precision of the model was at 0.713421. This is the ratio of correctly predicted positive observations to the total predicted positive observations. This brings the estimate of the regression model to 71.34%.

6.2.1 Precision Results

These were the precision results from the comparison done of ANN, Naïves Bayes and Support Vector Machine models. The below table shows the output.



	ANN	Naïve Bayes	Support Vector Machine
Precision	71.34%	61.33%	52.7%

Table 6.1: Precision Comparison Table

6.3 Model Implementation Outputs

The training output of the model was the time taken. Running the model however requires that the user keys in their preferred colleagues upon which the model narrows down the inputs of the selected workers. The training and validation output mirrored that of the training output. 20% of the dataset was used for testing and 10% of the dataset was used for validation.

6.4 Contribution to Research

The model developed provided an efficient way to allocate tasks to employees who can perform the task within the shortest time possible. This was effective in utilizing the employees well thus improving performance and also increased the end user satisfaction because the quick resolution means return to normalcy by the users.

6.5 Shortfalls of the Model

Since the proposed model takes an assumption of work performance by two people, it does not take into account work that can be done by an individual or more than two people. Even though this is possible, it goes beyond the scope of pairing for efficiency. If the model allowed for flexibility in allowing who can perform the work whether a single individual or more than two people, this can produce more accurate results as there can be times that some employees prefer working alone or if the task cannot be performed by two people and require more personnel.

Additionally, users who were supposed to be assisted, were not able to interact with the model in stating clearly their issues. This would have brought the aspect of specifics in the tasks or issues raised. User data can enhance the prediction in adding information about the user history. This can improve the levels of certainty and efficiency. Furthermore, since data obtained contained duplicates and gaps, data processing proved a challenge during the study. This reduced the available data that could be used for an effective model accuracy.

Chapter 7: Conclusions and Recommendations

7.1 Conclusion

The main objective of this research was to develop a predictive algorithm for task allocation that utilizes the artificial neural network algorithm. This objective was divided into five major objectives in order to adequately achieve the main objective. These major objectives are as follows:

- i. To investigate the factors to consider during task allocation.
- ii. To analyze challenges faced during task allocation.
- iii. To examine algorithms, approaches and models used in task allocation.

These first three objectives were met by reviewing the existing literature. Factors to consider during task allocation, challenges and approaches aided in understanding the whole concept of task allocation in different settings.

- iv. To develop a predictive algorithm for task allocation.

A model was developed to prove the concept using object system prototyping as the system methodology which allowed for changes and refinements in quick succession. The design for the model included the use of use case, class, dataflow and a database schema. They were used to provide analysis in gathering the functionalities. The model made use of secondary data. This objective was well covered in chapter 4 and chapter 5.

- v. To test the developed algorithm.

Validation of the outputs generated from the expected outputs of the model enabled the evaluation of the model. The model used a command line based interface which was very easy to use. The model was packed in a pickle executable file which operates in different environments thus interaction is faster through parsing of objects through stub files. It also offers easy and quick serialization. Upon validation, the model was accurate as the selected pairs had the least time. However, the, major shortfalls of this model was the fact that an assumption was made that a task had to be performed by two people. The model did not seek to assign a task to one person even if the task was small or even to assign to more than two people. The model also did not consider the availability of the persons assigned the tasks.

This research aimed at matching employees to tasks with consideration of the qualities possessed by the employee and choice of partner. The research was able to prove that when employee preference is taken into account together with right skills for the task, time is saved thus providing efficiency as tasks are performed faster and are done well.

7.2 Recommendations

In consideration of the results obtained from the implementation and testing of the proposed prototype, the following recommendations were made:

- i. To increase efficiency tasks that require one person and those that require more than two people to complete it effectively should be considered on the system.
- ii. To improve the levels of certainty and efficiency, users should be able to request directly on the system. User data will enhance the prediction in adding information about the user history. This can assist especially in cases where a task or issue has been reoccurring and this also it improves the level of preparedness of the IT personnel required to resolve the issue.
- iii. To ease the use of the proposed model, an end to end system should be developed that offers a web application and is incorporated to the RT ticketing system.

7.3 Suggestions for Research

Following the scope and limitations of this research, more research needs to be done on the following:

- i. Use an improved Gale and Shapley's' that removes biasness for matching. Matches with more than one output scenario should be provided to cater for situations whereby the first best match of individuals are not available for the task.
- ii. Employee experience after the task should also be a key consideration in future works. Since we are bringing the human aspect such that employees preferences are considered, a review should later on be carried that allows the employees to give feedback on their experience with the match given in performing the task.
- iii. Future research should look into all the aspects of employee task relationship even going further on to aspects such as designing skills and tasks for every employee's capability.

References

- Andrews, M. C., Kacmar, K. M., Blakely, G. L., & Bucklew. (2008). Group Cohesion as an Enhancement to the Justice—Affective Commitment Relationship. *Group & Organization Management*, 736–755.
- Avinash, A., & Ramani, K. (2014). A Hybrid Technique for Software Project Scheduling and Human Resource Allocation. *International Journal of Engineering Development and Research*, 2(3), 3243–3251.
- Baiden, B. K., & Price, A. D. F. (2011). The effect of integration on project delivery team effectiveness. *International Journal of Project Management*, 29(2), 129–136. <https://doi.org/10.1016/j.ijproman.2010.01.016>
- BallesterosPérez, P., GonzálezCruz, M. C. and F., & Diego, M. (2012). Human resource allocation management in multiple projects using sociometric techniques. *International Journal of Project Management*, 901–913.
- Barash, G., Bartolini, C., & Wu, L. (2007). Measuring and improving the performance of an IT support organization in managing service incidents. Second IEEE/IFIP International Workshop on Business-Driven IT Management, BDIM 2007, c, 11–18. <https://doi.org/10.1109/BDIM.2007.375007>
- Bartolini, C., Stefanelli, C., & Tortonesi, M. (2008). SYMIAN: A simulation tool for the optimization of the IT incident management process. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5273 LNCS, 83–94. <https://doi.org/10.1007/978-3-540-87353-2-7>
- Booch, G., Maksimchuk, R. a., Engle, M. W., Young, B. J., Conallen, J., & Houston, K. a. (2007). Object-Oriented Analysis and Design with Applications. In *Chemistry & biodiversity* (Vol. 1, Issue 11). <https://doi.org/10.1145/1402521.1413138>

- Campion, M. a., & Gina J. Medsker. (1993).
Campion_Medsker1993_Relations_Between_Work.pdf. In *Personnel Psychology*
(Vol. 46, pp. 823-). <https://doi.org/10.1111/j.1744-6570.1993.tb01571.x>
- Chen, J., Wang, Z., & Wang, S. (2010). Method research for the allocation of project human resources based on multi-dimensional model. *Proceedings - 3rd International Conference on Information Management, Innovation Management and Industrial Engineering, ICIIE 2010*, 1, 160–163. <https://doi.org/10.1109/ICIIE.2010.44>
- Chen, J., Yun, C., & Wang, Z. (2009). Multi-dimensional model method for the human resource allocation in multi-project. *2009 International Conference on Information Management, Innovation Management and Industrial Engineering, ICIIE 2009*, 4, 364–366. <https://doi.org/10.1109/ICIIE.2009.547>
- Daniel, L., Centeno, G., Ruiz-torres, A. J., & Otero, C. E. (2009). Computers & Industrial Engineering A systematic approach for resource allocation in software projects. *Computers & Industrial Engineering*, 56(4), 1333–1339. <https://doi.org/10.1016/j.cie.2008.08.002>
- Dennis, Wixom, & Roth. (2012). *Systems Analysis & Design* (5th Edition). John Wiley & Sons, Inc.
- e Silva, L. C., & Costa, A. P. C. S. (2013). Decision model for allocating human resources in information system projects. *International Journal of Project Management*, 31(1), 100–108. <https://doi.org/10.1016/j.ijproman.2012.06.008>
- Fast-Berglund, A., & Stahre, J. (2013). Task allocation in production systems - Measuring and analysing levels of automation. *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 46(15 PART 1), 435–441. <https://doi.org/10.3182/20130811-5-US-2037.00032>
- Gale, D., & Shapley, L. S. (1962). College Admissions and the Stability of Marriage Author (s): D . Gale and L . S . Shapley Source : The American Mathematical Monthly , Vol

. 69 , No . 1 (Jan . , 1962), pp . 9-15 Published by : Mathematical Association of America Stable URL : [http://www.Mathematical Association of America](http://www.MathematicalAssociationofAmerica), 69(1), 9–15.
<https://doi.org/10.2307/2312726>

Giannakopoulos, I., Karras, P., Tsumakos, D., Doka, K., & Koziris, N. (2016). An equitable solution to the stable marriage problem. Proceedings - International Conference on Tools with Artificial Intelligence, ICTAI, 2016-Janua(i), 989–996.
<https://doi.org/10.1109/ICTAI.2015.142>

Hendriks, M. H. A., Voeten, B., & Kroep, L. (1999). Human resource allocation in a multi-project R&D environment: Resource capacity allocation and project portfolio planning in practice. International Journal of Project Management, 17(3), 181–188.
[https://doi.org/10.1016/S0263-7863\(98\)00026-X](https://doi.org/10.1016/S0263-7863(98)00026-X)

Iwama, K., & Miyazaki, S. (2008). A survey of the stable marriage problem and its variants. Proceedings - International Conference on Informatics Education and Research for Knowledge-Circulating Society, ICKS 2008, i, 131–136.
<https://doi.org/10.1109/ICKS.2008.7>

Korathi, C. R. (2004). Research Methodology Methods and Techniques (Second Edi, Issue c). New Age International Publishers.

Kotwal, J. G., & Tanuja, S. D. (2015). Solving Task Allocation to the Worker Using Genetic Algorithm. 6(4), 3736–3741.
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.695.7180&rep=rep1&type=pdf>

Kukreja, H., N, B., S, S. C., & S, K. (2016). An introduction to Artificial Neural Network. Appita Journal, 1(5).

Lancashire, L. J., Lemetre, C., & Ball, G. R. (2009). An introduction to artificial neural networks in bioinformatics - Application to complex microarray and mass

- spectrometry datasets in cancer studies. *Briefings in Bioinformatics*, 10(3), 315–329.
<https://doi.org/10.1093/bib/bbp012>
- Liu, R., & Lee, J. (2012). IT incident management by analyzing incident relations. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 7636 LNCS, 631–638.
<https://doi.org/10.1007/978-3-642-34321-6-49>
- MATTSON, M. (1996). Object-Oriented Frameworks—A survey of methodological issues. 1996, 130p. Tese (Licenciatura)—Department of ..., 96–167.
<http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:ObjectOriented+Frameworks+A+survey+of+methodological+issues#7>
- Mijwil, M. M. (2018). Artificial Neural Networks Advantages and Disadvantages Solving Traveling Salesman Problem (TSP) View project Turkish Character Recognition (TCR) View project Artificial Neural Networks Advantages and Disadvantages. LinkedIn, March. <https://www.researchgate.net/publication/323665827>
- Mulla, A., Raravi, G., Rajasubramaniam, T., Bose, R. P. J. C., & Dasgupta, K. (2016). Efficient task allocation in services delivery organizations. *Proceedings - 2016 IEEE International Conference on Services Computing, SCC 2016*, 555–562.
<https://doi.org/10.1109/SCC.2016.78>
- Nedzelsky, R. (2016). Human resources allocation in project management. *Proceedings of the 27th International Business Information Management Association Conference - Innovation Management and Education Excellence Vision 2020: From Regional Development Sustainability to Global Economic Growth, IBIMA 2016*, May.
- Pan, S., Larson, K., Bradshaw, J., & Law, E. (2016). Dynamic task allocation algorithm for hiring workers that learn. *IJCAI International Joint Conference on Artificial Intelligence, 2016-Janua*, 3825–3831.

- Park, J., Seo, D., Hong, G., Shin, D., Hwa, J., & Bae, D. H. (2014). Practical human resource allocation in software projects using genetic algorithm. *Proceedings of the International Conference on Software Engineering and Knowledge Engineering, SEKE, 2014-Janua(January)*, 688–694.
- Paul, R., Bose, R. P. J. C., Chalup, S. K., & Raravi, G. (2017). Improving Operational Performance in Service Delivery Organizations by Using a Metaheuristic Task Allocation Algorithm. *CEUR Workshop Proceedings*, 1985, 25–37.
- Remenyi, D.S.J.; Swartz, E.; Money, A. and Williams, B. (1998). *Doing Research in Business and Management: An Introduction to Process and Method*. SAGE Publications, London.
- Samath, S., Udalgama, D., Kurukulasooriya, H., Thelijjagoda, S., & Premarathne, D. (2017). Collabcrew - An Intelligent Tool for Dynamic Task Allocation within a Software Development Team.
- Sayer, A. (1992). *Method in Social Science: A Realist Approach*. London: Routledge. nd.
- Schulte, D. (2012). Towards a human task management reference model. *CEUR Workshop Proceedings*, 847, 104–111.
- Siala, M., & O’Sullivan, B. (2017). Rotation-based formulation for stable matching. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10416 LNCS, 262–277. https://doi.org/10.1007/978-3-319-66158-2_17
- Zhao, B., Wang, Y., Li, Y., Gao, Y., & Tong, X. (2019). Task allocation model based on worker friend relationship for mobile crowdsourcing. *Sensors (Switzerland)*, 19(4), 1–23. <https://doi.org/10.3390/s19040921>
- Zhou, L. (2008). A project human resource allocation method based on software architecture and social network. *2008 International Conference on Wireless Communications*,

Networking and Mobile Computing, WiCOM 2008, 1–6.
<https://doi.org/10.1109/WiCom.2008.1749>



Appendix A: Originality report

Click to go back, hold to see history

Feedback Studio

Simon Kibiwot Kipkemboi | A Machine Learning Model for Task Allocation

A Machine Learning Model for Task Allocation

KIBIWOT SIMON KIPKEMBOI

049552

Match Overview

23%

<

>

1

www.mdpi.com

Internet Source

2%

>

2

Ioannis Giannakopoulo...

Publication

2%

>

3

link.springer.com

Internet Source

2%

>

4

Submitted to Strathmor...

Student Paper

2%

>

5

docplayer.net

Internet Source

2%

>

6

Submitted to Savitribai ...

Student Paper

1%

>

7

pdfs.semanticscholar....

Internet Source

1%

>

8

Lúcio Camara e Silva, A...

Publication

1%

>

9

ceur-ws.org

Internet Source

1%

>

Page: 1 of 72

Word Count: 16063

Text-only Report

High Resolution On

VT OMNES VNVM SINT

Appendix B: Ethical Approval Letter



3rd March 2020

Mr Kipkemboi, Simon
skipkemboi@strathmore.edu

Dear Mr Kipkemboi,

RE: A machine Learning Model for Task Allocation

This is to inform you that the SU-IERC has reviewed and **approved** your above research proposal. Your application approval number is SU-IERC0576/19. The approval period is **3rd March, 2020 to 2nd March, 2021**.

This approval is subject to compliance with the following requirements:

- i. Only approved documents including (informed consents, study instruments, MTA) will be used
- ii. All changes including (amendments, deviations, and violations) are submitted for review and approval by SU-IERC.
- iii. Death and life threatening problems and serious adverse events or unexpected adverse events whether related or unrelated to the study must be reported to SU-IERC within 72 hours of notification
- iv. Any changes, anticipated or otherwise that may increase the risks or affected safety or welfare of study participants and others or affect the integrity of the research must be reported to SU-IERC within 72 hours
- v. Clearance for export of biological specimens must be obtained from relevant institutions.
- vi. Submission of a request for renewal of approval at least 60 days prior to expiry of the approval period. Attach a comprehensive progress report to support the renewal.
- vii. Submission of an executive summary report within 90 days upon completion of the study to SU-IERC.

Prior to commencing your study, you will be expected to obtain a research license from National Commission for Science, Technology and Innovation (NACOSTI) <https://oris.nacosti.go.ke> and also obtain other clearances needed.

Yours sincerely,


for: Dr Virginia Gichuru,
Secretary; SU-IERC

Cc: Prof Fred Were,
Chairperson; SU-IERC



Appendix C: Complete Model Interface

Command Prompt

```
-----  
INTELLIGENT TASK ALLOCATION MODEL (Simon Kipekmboi)  
-----
```

```
Succesfully Connected to Manifest!  
-----
```

```
(24, ' Printing Pin')
```

```
(1, 'Account Closure')
```

```
(2, 'AMS Management')
```

```
(3, 'Audiovisuals')
```

```
(4, 'Communications')
```

```
(5, 'Conferences & Events')
```

```
(6, 'Conferences & Events')
```

```
(6, 'Data Requests')
```

```
(7, 'Default')
```

```
(8, 'Elearning')
```

```
(9, 'Hardware')
```

```
(10, 'Movement')
```

```
(11, 'Network')
```

```
(12, 'Phones')
```

```
(13, 'Printer Management')
```

```
(14, 'Printing A/Cs and Credits')
```

```
(15, 'Printing Top UP')
```

```
(16, 'Purchases')
```

```
(17, 'SBS Support')
```

```
(18, 'Security')
```

```
(19, 'Software')
```

```
(20, 'Staff Training')
```

```
(21, 'Student Emails/Domain passwords.')
```

```
(23, 'Student Laptop')
```

```
(24, 'Systems Support')
```

```
(25, 'UPS')
```

```
(26, 'User Account Management')
```

CHOOSE THE TASK CATEGORY ID FROM THE LIST ABOVE:

25
(3060, 'AOtonglo (Allan Otonglo)')

(1024, 'bogutu (Benson Ogutu)')

(2888, 'JKimotho (Joyce Kimotho)')

(1221, 'mkihara (Maria Awuor)')

(1824, 'MMbabu (Mike Mbabu)')

(305, 'Msiro (Michael Siro)')

(299, 'MSuka (Matthews Suka)')

CHOOSE ANY THREE PEOPLE YOU WISH TO WORK WITH FROM THE TOP PEOPLE (KEY IN ID)

ID FOR 1st Preference?:

3060

-----ID FOR 2nd Preference?:

1024

-----ID FOR 3rd Preference?:

299
(2, 0, 3, 4000, 1, 3, 1, 29, 1, 10)

(2, 0, 3, 4000, 1, 3, 1, 29, 3, 10)

(2, 0, 3, 4000, 1, 3, 1, 29, 2, 10)

Succesfully Connected to Manifest!

Succesfully Connected to Manifest!

Name: ('skipkemboi (Simon Kipkemboi)',)
Time taken(Minutes):
[30.]
Verdict :--[PROBABLE!!--

Name: ('mndeto (Martin Ndithi Ndeto)',)
Time taken(Minutes):
[60.]
Verdict :--[PROBABLE!!--

Name: ('Msiro (Michael Siro)',)
Verdict :--[DISQUALIFIED!!--

Appendix C: Model Training Code Snippet

```
#
# print(X_train)
# print(y_train)
scaler = pre.Normalizer()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.fit_transform(X_test)

mlp = MLPRegressor(alpha=1e-3,hidden_layer_sizes=(60,100,80),random_state=1,activation="relu",max_iter=100000)
|

mlp.fit(X_train_scaled, y_train.ravel())

predict_train = mlp.predict(X_train_scaled)
predict_test = mlp.predict(X_test_scaled)

# print("Test Loss: "+(mlp.loss))
print(mlp.loss)
print(r2_score(y_test, mlp.predict(X_test_scaled),multioutput='variance_weighted'))

# Save to file in the current working directory
pkl_filename = "pickle_model.pkl"
with open(pkl_filename, 'wb') as file:
    pickle.dump(mlp, file)
```

