March - April 2020
ISSN: 0193-4120 Page No. 307 - 321

A Machine Learning based Software Project Schedule Management Solution

[1]Muhammad Ehsan Rana, [2]Wang Wei
[1][2]School of Computing and Technology, Asia Pacific University of Technology and Innovation (APU)
Technology Park Malaysia (TPM), 57000 Bukit Jalil, Kuala Lumpur, Malaysia
[1]muhd_ehsanrana@apu.edu.my,[2]TP049066@mail.apu.edu.my

Article Info
Volume 83
Page Number: 307 - 321
Publication Issue:
March - April 2020

Article History
Article Received*: 24 July 2019*
Revised*: 12 September 2019*
Accepted*: 15 February 2020*
Publication*: 12 March 2020*

Abstract
Software project schedule management has become a concern for many small and medium-sized companies. They still use traditional project management solutions and are unable to meet the needs of modern software. Some large software companies do acquire services of professional tools for managing software project progress, but small and medium-sized software companies are still using legacy management methods, resulting in frequent project delays. Artificial intelligence is the trend of current technology development which is shaping the current IT landscape. Nowadays, many industries such as the service industry and manufacturing involve artificial intelligence technologies in their business processes. This paper focuses on the application of machine learning for software project schedule management, which can greatly improve the efficiency of software development and reduce the cost involved. This research uses a linear regression model in machine learning to create a predictive model that can be used to predict the development time required by each developer. Data analysis can then provide effective advice to software project managers.

Keywords: *Software Project Schedule Management, Machine Learning (ML), Artificial Intelligence, Predictive Model.*

1. INTRODUCTION

With the advancement of social science and technology and the development of intelligent technology, software engineering has become the pillar of the industry to promote the rapid development of the social economy. With the rapid development of the software industry, software project schedule management has become the top priority in the software industry [1]. At present, large software companies have professional team management software project progress, but small and medium-sized software companies are still using backward management methods, resulting in frequent project delays. Using backward software project schedule management methods will greatly increase the cost of software development. This paper proposes to apply machine learning technology to software project schedule management, which is a technical method to assist software developers in project schedule management.

2. LITERATURE REVIEW

2.1. WHAT IS SOFTWARE PROJECT SCHEDULE MANAGEMENT?

The software product is unique in that it is an invisible product and does not have physical properties. Software products combine ideas, concepts, processes, algorithms, organization, optimization, and efficiency [2]. The specific characteristics of the software project management content are as follows:

Scope management of software projects. In order to complete the established arrangement of the software project, the management measures of the intrinsic program of the software project are

controlled [3]. There are scope divisions, scope provisions, scope changes, etc.

Time management of software projects. In order to ensure that the software project is completed as scheduled, the relevant management procedures are carried out [4]. It includes several tasks such as specific task division, character order, schedule, schedule estimation and schedule.

Management of team human resources. Each position needs to arrange reasonable team members, reasonably assign tasks to team members, and ultimately ensure that software projects can be completed within the specified time. [5].

2.2. DIFFICULTIES IN SOFTWARE PROJECT SCHEDULE MANAGEMENT

1.The accuracy of software project schedule estimates is not accurate.

In software project schedule management, software project schedule estimation is a huge challenge. According to statistics, most teams tend to estimate the software project progress when the estimated value is lower than the actual value. The actual completion time for most projects exceeded 25% to 100% of the estimate. [6].

2.Did not find and reduce project risks as early as possible.

The later the error is discovered in software development, the greater the loss to development. The main function of the milestone development model is inspection and metrics. To avoid wasting time and money, the milestone development model can test the results in stages based on the output of each stage. [7]. In the early stages of the project, milestone reviews can be used to identify problems in requirements and design projects, which can reduce the likelihood of late modification and rework. For example, software project development can be divided into five stages. If errors are found in the first stage and errors are corrected in time, the loss of the project is not large, but if errors are found in the last stage, the cost of modifying the errors will be much greater.

3. Project schedule lacks effective supervision and control.

The average person has a habit of loosening and tight at work. Milestones dictate what tasks a developer needs to accomplish at a particular time, so that the work can be distributed reasonably, and the granularity of management can be refined. For large software development projects, each phase of the work needs to be completed step by step. With milestones, you can clearly understand the completion of each step of the project, you can more accurately understand the progress of the project.[8].

2.3. WHAT IS MACHINE LEARNING?

Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed.

Machine learning is to enable computers to simulate human learning behaviors, automatically acquire knowledge and skills through learning, reorganize existing knowledge structures, and constantly improve their own mental energy and self-improvement [9]. That is, machine learning studies how to use machines to identify and utilize existing knowledge to acquire new knowledge and skills. It is the core of artificial intelligence and the fundamental way to make computers intelligent. Machine learning mainly focuses on three aspects: learning mechanism, learning method and task oriented. Its application is almost in all fields of natural science.

2.4. USE MACHINE LEARNING TO IMPROVE SOFTWARE PROJECT SCHEDULE MANAGEMENT

The current challenge for project managers is to reasonably plan the project time and enable team members to achieve the project's goals at a given time[10]. Accurately estimating the project

completion time is the biggest problem in the project manager's work. Using machine learning technology can take software project schedule management to a whole new level[1]. There is many project schedule management software, such as ClickUp, which are testing how machine learning technology predicts what users are going to do. Applying machine learning techniques to project schedule management should have the following features:

- Assign tasks to the right members of the team accurately.
- Properly estimate when the task is completed[11].
- Can provide users with reasonable advice to develop a project plan.

2.5. Regression Models

Regression is a technique for modeling and analyzing the relationships between variables, and how the variables affect the outcome [12]. Linear regression refers to a regression model consisting entirely of linear variables [13]. Starting from a simple case, Single Variable Linear Regression is a technique for modeling the relationship between a single input independent variable (feature variable) and an output dependent variable using a linear model. The more general case is Multi Variable Linear Regression, which embodies the relationship between multiple independent input variables (feature variables) and output dependent variables. The model remains linear because the output is a linear combination of input variables [14].

3. Methodology

The software development cycle can be divided into the following five phases: the requirements phase, the design phase, the development phase, the testing phase, and the release phase. The progress management of software development needs to be managed for these five phases [15].

It takes 3-5 months to develop a mobile app with existing data. Kinvey launched a survey in 2013 to survey 100 software designers working on mobile apps [16]. The conclusion is to develop a native mobile. The APP takes about 18 weeks, and the front-end development of the application takes about 8 weeks, and the back-end development takes about 10 weeks.

GoodFirms released a report in 2017 to develop a feature-rich and complex mobile application, such as Instagram and Uber, which takes 4.5-5.5 months to develop and develop a medium-complexity mobile Applications such as WhatsApp take approximately 4.6 months to develop. Developing a user-friendly but less functional application, such as Tinder, takes about 3.8-4.1 months to develop [16].

The complexity of software programs and the proficiency of developer skills can affect the software development cycle.

3.1. Research Design

The whole research process can be divided into six stages, namely the questionnaire stage, the data collection stage, the predict model creation stage, the forecast data sorting stage, the tool production stage and the best data stage. The overall process can be seen in the following picture:

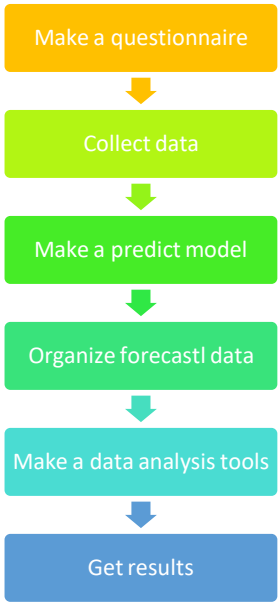

Figure 1：The process of research

March - April 2020
ISSN: 0193-4120 Page No. 307 - 321

The forecast data analysis phase is divided into five steps. The first step is to use Python to generate a linear graph of the original data. This can visually see the degree of dispersion of the collected data and can have a general understanding of the original data; then continue to Use the algorithm in NumPy to find the maximum, minimum, median and average of the original data. With this data, you can have a deeper understanding of the original data, as shown below:

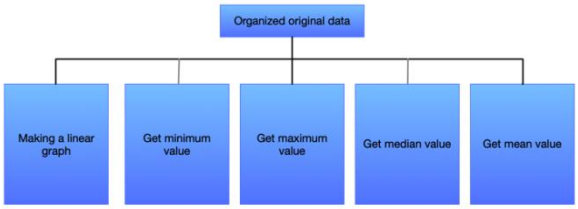

Figure 2: The process of organized original data

In the data analysis phase, you need to use the algorithm to get the best value of the time required for each stage of software development, then use python to generate a table, then use matplotlib and pandas toolkit to generate the bar chart and pie chart, the bar shape The graph can make the optimized data compare with the original data more intuitively. The pie chart can abstractly show the proportion of time required for each stage of software development to the total time, which is convenient for drawing the final conclusion.

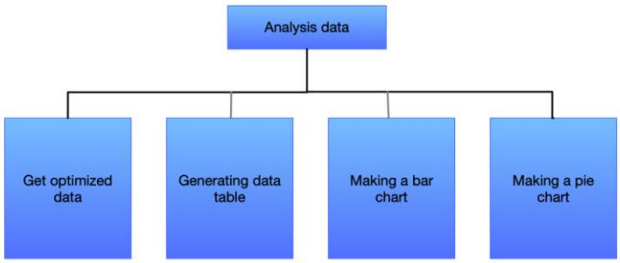

Figure 3: The process of Analysis data

3.2. RESEARCH ACTIVITIES

3.2.1. DATA COLLECTION

There are relatively few data on software development cycle time statistics on the Internet. Here, more than 100 software development practitioners are surveyed in the form of questionnaires. The questionnaire format is as follows. The questionnaire is used to collect sample data. Through the data collected by the questionnaire, the CSV file is unified.

Since the general software development is divided into the above five stages, the content of the questionnaire is also to understand the time required for the developers with different professional levels to complete these stages and then organize the data.

In this questionnaire, six questions were asked to 100 investigators at the questionnaire stage. The questions are as follows:

As per your experience, what is the estimated time taken for the following phases of software development:

- Requirements phase
- Design phase
- Development phase
- Testing phase
- Release phase
- Overall software development

Respondents need to accurately estimate the time they need and feed back to the questionnaire. After collecting the data of 100 investigators through the questionnaire, the csv file was unified, and the following figure is part of the data in the csv file.

Questionnaire data

ID	req	des	dev	test	rel	year
1	35	35	39	20	15	1
2	34	34	40	20	15	1
3	33	34	39	19	15	1
4	34	33	39	19	16	1
5	35	34	38	18	14	1
6	32	31	35	16	15	2
7	31	31	35	16	16	2
8	30	31	36	16	14	2
9	30	30	35	16	14	2
10	29	30	35	16	14	2

Figure 4: A part of questionnaire data

March - April 2020
ISSN: 0193-4120 Page No. 307 - 321

3.2.2. ORIGINAL DATA SORTING

After the data collection is completed, the original data needs to be sorted. First, the working age of the software development is x-axis, and the specific time of each development is the y-axis, and the scatter plot is drawn. And the linear regression analysis in machine learning is used to calculate the a and b values in each data. The model is finally created, and the time required by a developer at each stage of the software development cycle can be predicted after the developer's working age is known. For example, the following picture:

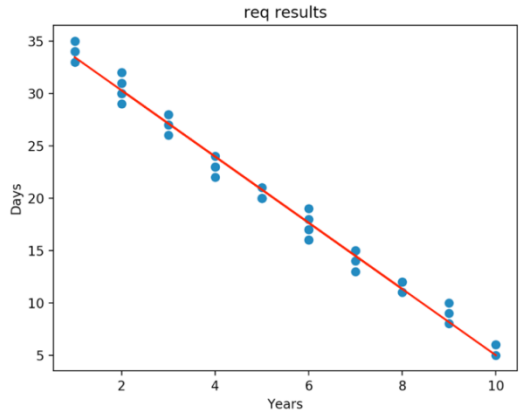

Figure 5: The sample of create predict model

The second stage is to sort out the forecast data. The forecast data is arranged to make the data more intuitive to the user. At this stage, you need to use python to make 100 pieces of data collected into a linear graph. NumPy, Pandas and Matplotlib are three toolkits, and import these three packages into the build environment, as shown below:

```
from __future__ import division
import numpy as np
import pandas as pd
from pandas import Series
import matplotlib.pyplot as plt
```

Figure 6: Import toolkits

the three toolkits, the data files needed for the research are imported. After the import, the first column is removed. This column belongs to the ID of 100 survey objects. This column of data is useless for research analysis, and then the original is drawn in python. It can be seen from the figure that the values are quite scattered, which also indicates that the degree of professionalism of the respondents is different, but it can be seen from the figure that the time of the test and release phases are relatively small, and the development phase takes a long time. Since the data is too scattered, the data needs to be further processed, and the optimal value of each stage is calculated by an algorithm.

3.2.3. PROCESSING FORECAST DATA

It can be seen from the raw data that the overall data is quite scattered. In this step, it is necessary to calculate the maximum, minimum, average, median and mode of time required for each development step. Here you can use np.min to extract the minimum value in each column of data; use np.max to extract the maximum value of each column; use np.mean to calculate the average value of each column; use np.median to extract each column. median. The specific operation is as follows:

```
min1 = np.min(df.iloc[:,1])
min2 = np.min(df.iloc[:,2])
min3 = np.min(df.iloc[:,3])
min4 = np.min(df.iloc[:,4])
min5 = np.min(df.iloc[:,5])

max1 = np.max(df.iloc[:, 1])
max2 = np.max(df.iloc[:, 2])
max3 = np.max(df.iloc[:, 3])
max4 = np.max(df.iloc[:, 4])
max5 = np.max(df.iloc[:, 5])

mea1 = np.mean(df.iloc[:, 1])
mea2 = np.mean(df.iloc[:, 2])
mea3 = np.mean(df.iloc[:, 3])
mea4 = np.mean(df.iloc[:, 4])
mea5 = np.mean(df.iloc[:, 5])

median1 = np.median(df.iloc[:, 1])
median2 = np.median(df.iloc[:, 2])
median3 = np.median(df.iloc[:, 3])
median4 = np.median(df.iloc[:, 4])
median5 = np.median(df.iloc[:, 5])
```

Figure 7：Produce forms using python

The extracted data can be made into a 6-row, 6-column table, and a new CSV file is generated. The figure below is the data extracted from the original data.

	max_data	mean_data	median_data	min_data
req	35	19.222222	19.0	5
des	36	21.373737	23.0	6
dev	40	20.848485	21.0	5
test	20	13.414141	14.0	6
rel	15	10.777778	11.0	6

Figure 8：Python generated results

Further analysis of the extracted data, due to the limited number of samples, and the large span of the original sample data, in order to further optimize the data, choose to remove the maximum

March - April 2020
ISSN: 0193-4120 Page No. 307 - 321

and minimum values in each column and then find the average of each column. This method can reduce the error of the sample to some extent. The value obtained can be referred to as an optimized average.

```
optimal1 = (mea1 * 100 - min1 - max1) / 98
optimal2 = (mea2 * 100 - min2 - max2) / 98
optimal3 = (mea3 * 100 - min3 - max3) / 98
optimal4 = (mea4 * 100 - min4 - max4) / 98
optimal5 = (mea5 * 100 - min5 - max5) / 98
```

Figure 9：Algorithm for refining data

In the end, the extracted maximum, minimum, average, median and optimized averages are statistically made into a 6-row, 7-column table, and a bar chart is used to make a further understanding of the data.

result

	max_data	mean_data	median_data	min_data	optimal_data
req	35	19.2222222222222	19	5	19.2063492063492
des	36	21.3737373737374	23	6	21.381364667079
dev	40	20.8484848484848	21	5	20.8147804576376
test	20	13.4141414141414	14	6	13.4225932797361
rel	15	10.7777777777778	11	6	10.7834467120181

Figure 10：Final result table

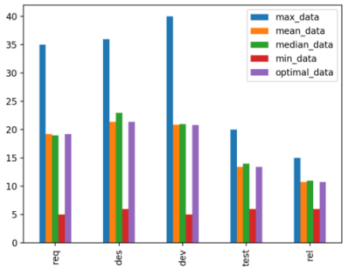

Figure 11：Bar chart generated using python

At the end of the five stages of software development, a pie chart is generated with a percentage, which may be more intuitive to understand the total time required for each stage of the software development process.

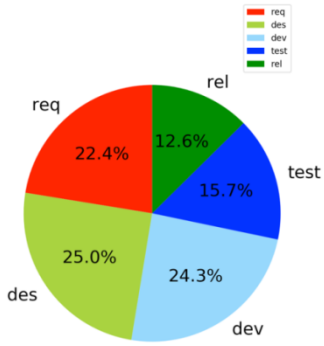

Figure 12：Pie chart generated using python

4. PROPOSED ARTEFACT

The artefact of this research is a set of methods for analyzing data using Python. The first is data collection, then the data collected by Python is used, and a set of predictive models is created using machine learning linear regression algorithm, and then generated with analysis the chart of value, the final user can refer to the data of the generated chart and then combine the actual planning of the progress management of the existing project. The specific simple process is as follows:

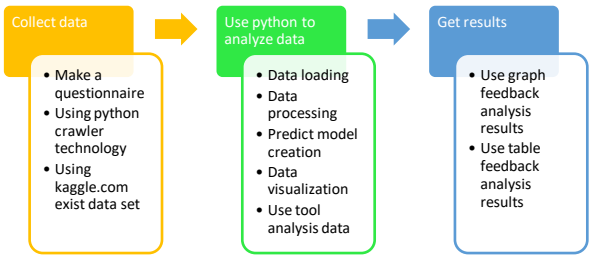

Figure 13: The flow chart of artefact

4.1. DATA COLLECTION

There are many ways for users to collect data. The most common way is to use the form of a questionnaire. In addition, you can use python crawler technology to extract the data needed for analysis on the corresponding website or use the existing ones on the kaggle.com website data set. The larger the number of data, the more accurate the conclusions will be. Using the questionnaire to investigate the company's employees can get more

realistic conclusions. The specific form of the questionnaire can include the following questions:

As per your experience, what is the estimated time taken for the following phases of software development:

- Requirements phase
- Design phase
- Development phase
- Testing phase
- Release phase
- Overall software development

After completing the questionnaire, the collected data can be made into a form in the following format, and then the data is imported into the data analysis tool.

Table 1: The format of data

ID	req	des	dev	test	rel	years
1	…	…	…	…	…	…

4.2. Use Python to Analysis Data

4.2.1 Advantage of Python

Python has many advantages in data analysis. It has many scientific computing tools, such as NumPy, matplotlib, pandas, and so on. Python also has powerful programming capabilities, which can greatly improve the efficiency of data analysis [17].

4.2.2. Data loading

After the data collection is completed and the data is organized into csv files, the data files are read using python's "pd.read_csv" method. In the csv file, the first column of data is the ID column, which has no useful in data analysis. Continue to use python's "df.drop" method to remove the ID of the data. The specific method is as follows:

```
df = pd.read_csv(file_path)
df_1 = df.drop(['ID'], axis=1)
```

Figure 14: Data loading

4.2.3. Data processing

After the data is loaded, the collected data needs to be processed. The data processing is divided into the following two steps:

Step 1:

The working age of the software development for the survey object is the x-axis, and the specific time for each development is the y-axis, and a scatter plot is drawn. And the linear regression analysis in machine learning is used to calculate the "a" and "b" values in each data. The model is finally created, and the time required by a developer at each stage of the software development cycle can be predicted after the developer's working age is known. The specific process is as follows:

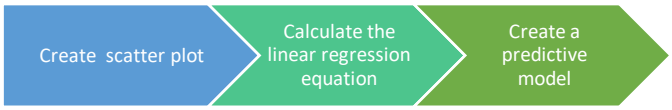

Figure 15: The flow of data processing step 1

Step 2:

After completing the first step of data processing, it is necessary to use the prediction model to predict the data of multiple employees and then perform the second step of data processing to find the maximum, minimum, intermediate and average values of multiple employees. The specific process is as follows:

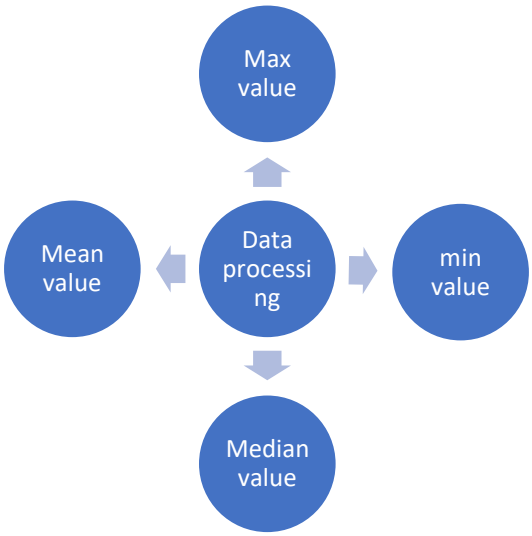

Figure 16: The flow of data processing step 2

March - April 2020
ISSN: 0193-4120 Page No. 307 - 321

In the NumPy toolkit of Python, the four methods "np.max", "np.min", "np.mediam" and "np.mean" can calculate the maximum, minimum, median and average values of the overall data. Use the above four methods to process the data, as shown in the following figure:

```
16      min1 = np.min(df.iloc[:,1])
17      min2 = np.min(df.iloc[:,2])
18      min3 = np.min(df.iloc[:,3])
19      min4 = np.min(df.iloc[:,4])
20      min5 = np.min(df.iloc[:,5])
21
22      max1 = np.max(df.iloc[:, 1])
23      max2 = np.max(df.iloc[:, 2])
24      max3 = np.max(df.iloc[:, 3])
25      max4 = np.max(df.iloc[:, 4])
26      max5 = np.max(df.iloc[:, 5])
27
28      mea1 = np.mean(df.iloc[:, 1])
29      mea2 = np.mean(df.iloc[:, 2])
30      mea3 = np.mean(df.iloc[:, 3])
31      mea4 = np.mean(df.iloc[:, 4])
32      mea5 = np.mean(df.iloc[:, 5])
33
34      median1 = np.median(df.iloc[:, 1])
35      median2 = np.median(df.iloc[:, 2])
36      median3 = np.median(df.iloc[:, 3])
37      median4 = np.median(df.iloc[:, 4])
38      median5 = np.median(df.iloc[:, 5])
```

Figure 17: Data processing

4.2.4. DATA VISUALIZATION

After the data processing is completed, the data needs to be visualized. Linear graph, bar graph and pie graph of data are needed in the research.

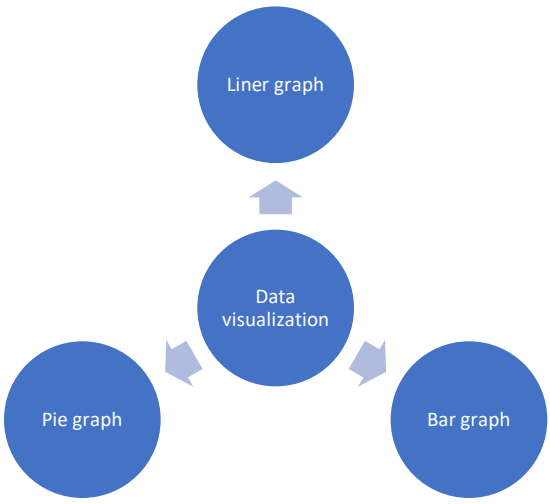

Figure 18: The flow of data visualization

The "plot()" method can be used to complete the drawing work using the plot() method of the matplotlib toolkit in python. The "plot.show()" method displays the data in the data table as a linear graph. The "plot.bar()" method is used in the data table. The data is displayed as a bar chart, and the "plot.pie()" method displays the data as a pie chart.

4.2.5. TOOL CREATION

In the research design of Chapter 3, the third stage is to make a data analysis tool to make it easier for users to analyze similar data and draw conclusions. The main function of this tool is to read the data file, generate a linear graph of the original data, and then generate a bar chart and a pie chart that can be compared by optimizing the algorithm. Finally, the user can generate a table with reference value. The specific functions of the data analysis tool are as follows:

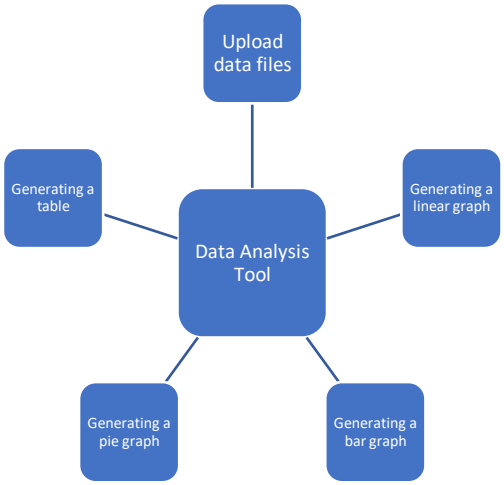

Figure 19: Main function of tool

To implement the data analysis tool, we need to use Python's Tkinter module. Tkinter is the interface of Python's standard Tk GUI toolkit. It can be used under most Unix platforms. It can also be used in Windows and Mac OS, so users This data analysis tool is available on any platform.

4.2.6. THE TOOL USAGE FLOW

When using this data analysis tool, users should first upload the data file to be analyzed. The data file should be a csv file with a .csv suffix. After uploading the data file, the user can choose to generate a linear graph of the original data, a bar graph of the optimized data, a pie chart of the best

March - April 2020
ISSN: 0193-4120 Page No. 307 - 321

data, and a table with analytical value. The specific flow chart is as follows:

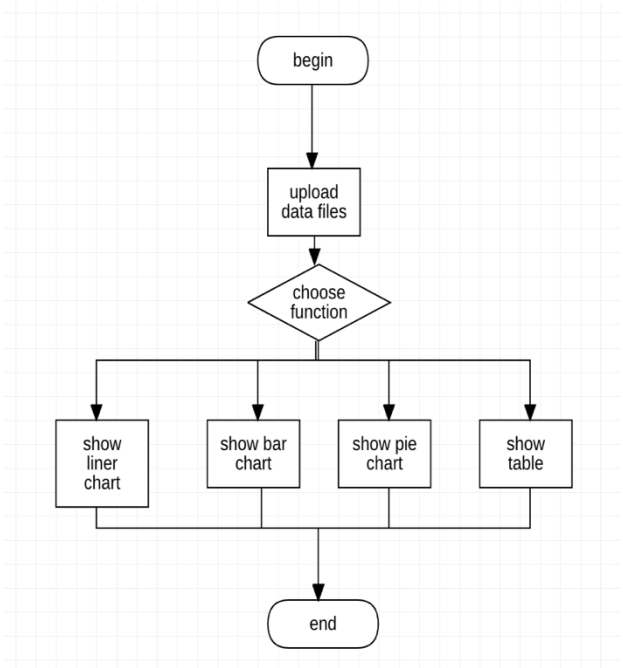

Figure 20：Flow chart of tool

4.2.7. THE TOOL SAMPLE INTERFACE

The interface of the data analysis tool is composed of five buttons and a text display area. The five buttons control the uploading data, the production of the linear graph, the generation of the bar graph, the generation of the pie chart and the generation of the table.

After clicking the upload file button, the selected file will be displayed. The file processed by the data analysis tool is a csv file, and the user needs to upload a corresponding csv file, as shown below:

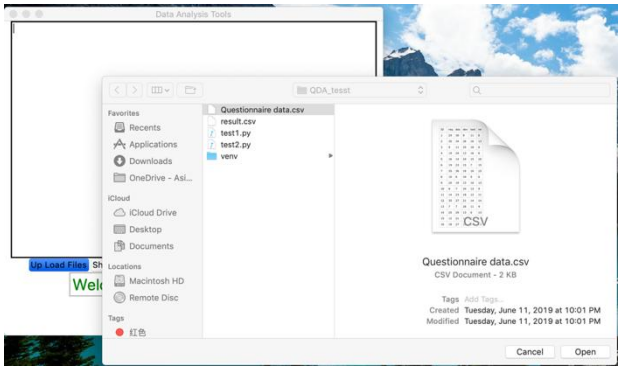

Figure 21: Upload function of tool

After uploading the data of the recalled questionnaire, you can choose to view the linear graph made of the 100 data. Click the show liner chart button to display the linear graph, as shown below:

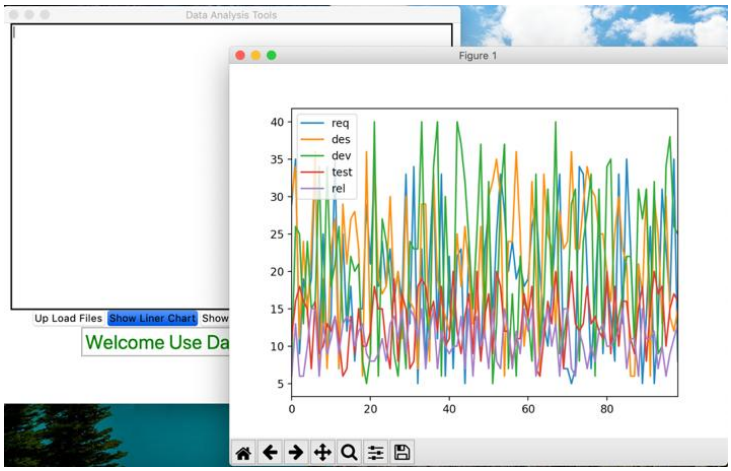

Figure 22: Show liner chart of tool

After viewing the linear graph, the data analysis tool can extract the maximum, minimum, average, intermediate values and the best values after using the optimization algorithm in the data and make a bar chart. The user clicks the show bar chart button to display a bar graph of the above data, as shown below:

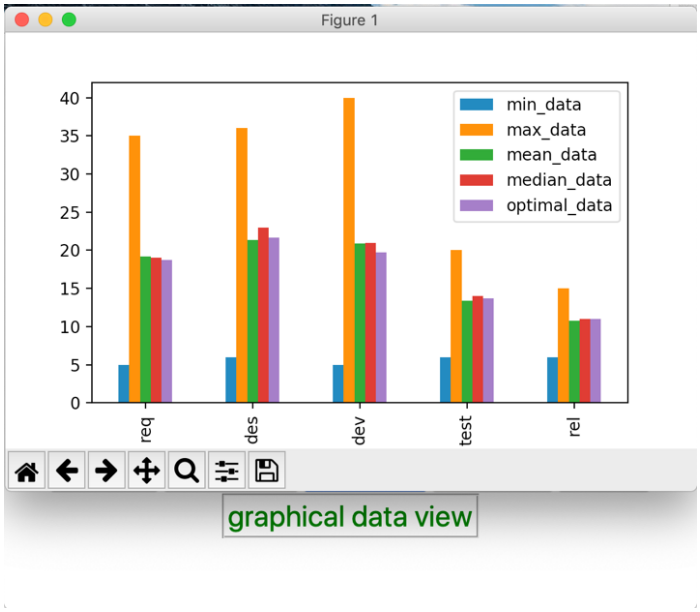

Figure 23: Show bar chart of tool

When the user clicks the show pie chart button, the pie chart can be displayed after an optimized

March - April 2020
ISSN: 0193-4120 Page No. 307 - 321

algorithm. The user can visually see the proportion of time spent in the software development phase, as shown below:

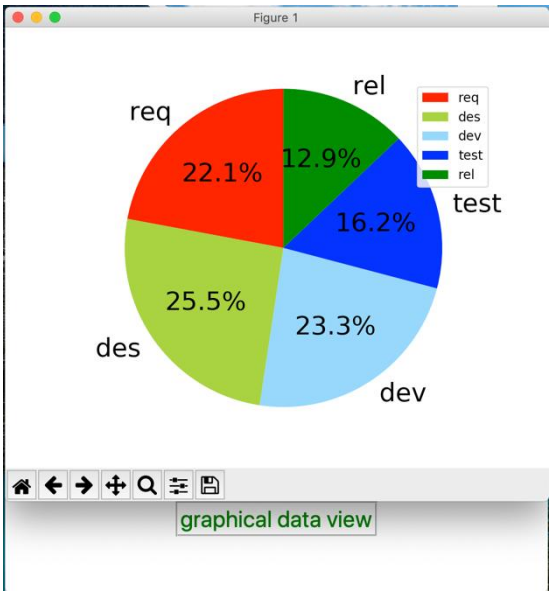

Figure 24: Show pie chart of tool

In addition to showing the user an intuitive graph, this tool can finally generate a table with the maximum, minimum, average, intermediate value and best data feedback to the user, the user can use the data in the table to carry out deeper Analysis, the table displayed by the tool is as follows:

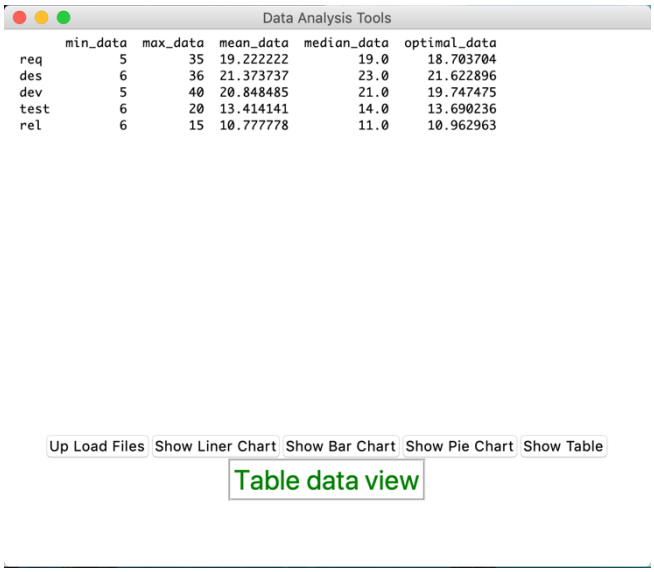

Figure 25: Show table of tool

4.3. GET RESULTS

After data processing, users can get five predictive models, each of which can predict the specific time an employee needs in the five phases of software development. Here are five models listed one by one.

Predictive model of software requirement step:

$$\text{days} = -3.16235 * years + 36.61878$$

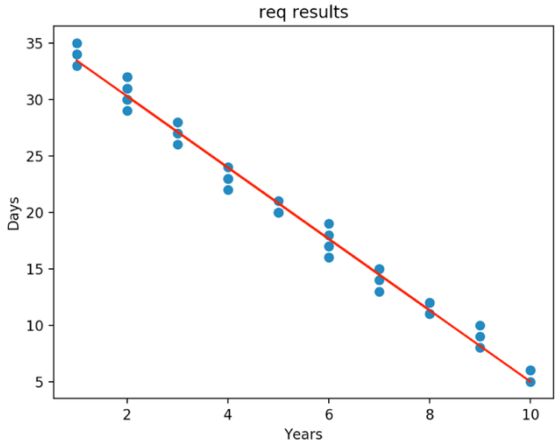

Figure 26: The req predict model

Predictive model of software design step:

$$days = -3.18912 * years + 36.83046$$

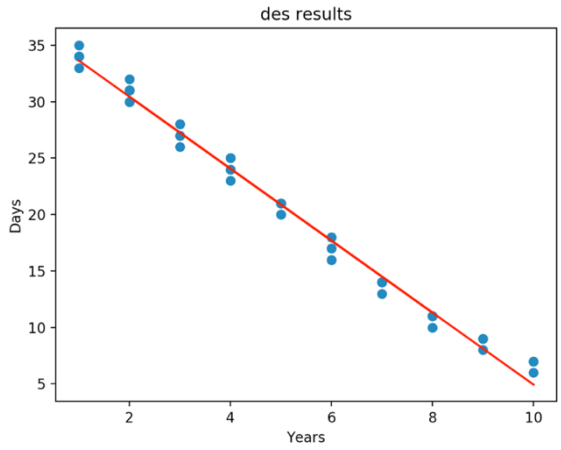

Figure 27: The des predict model

Predictive model of software development step:

$$days = -2.97335 * years + 41.02287$$

March - April 2020
ISSN: 0193-4120 Page No. 307 - 321

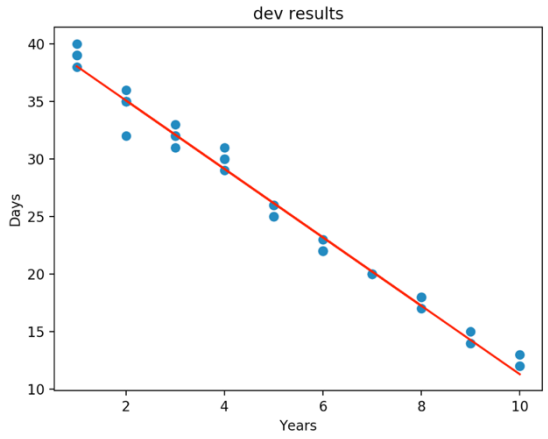

Figure 28: The dev predict model

Predictive model of software testing step:

$$days = -1.45521 * years + 19.07963$$

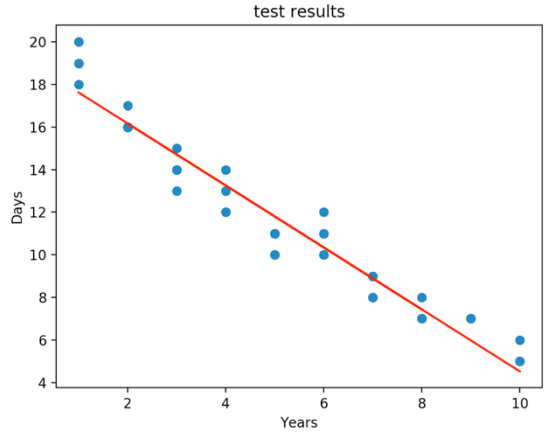

Figure 29: The test predict model

Predictive model of software release step:

$$days = -1.18746 * years + 16.72230$$

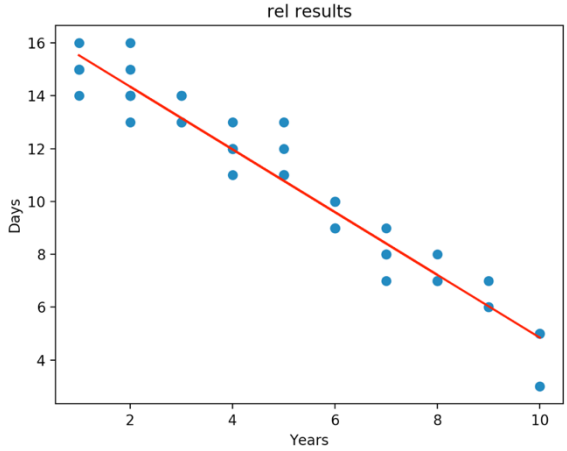

Figure 30: The rel predict model

Through the above five sets of predictive models, it can be seen that the longer the employee's working age, the shorter the time required in each stage of software development. After obtaining the above five groups of models, users can predict the time required by employees in all stages of software development. Then, the prediction results of several employees are imported into the data analysis tool. After analysis by the data analysis tool, three graphs and one table can be obtained. The three graphs are a linear graph, a bar graph and a pie graph. The data shown in the linear graph is a linear arrangement of unprocessed data, and the bar graph and pie graph show the data after data processing. The pie chart can be used as the final data conclusion, which contains the percentage of time required for the five phases of software development, as shown below:

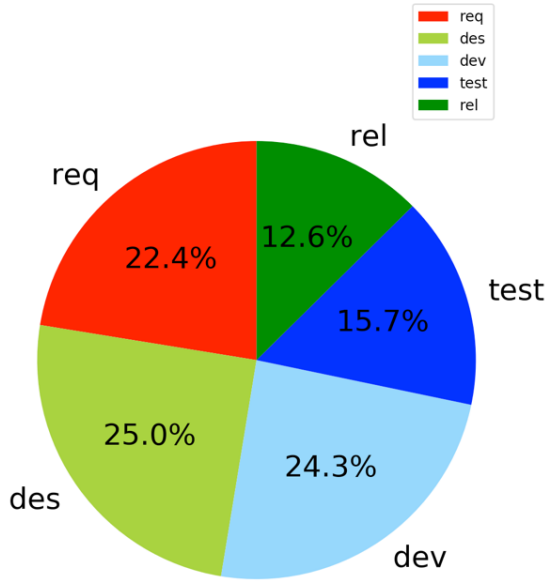

Figure 31: The graph results

The resulting table contains the minimum, maximum, average, intermediate values of the data and the optimized values obtained using the optimization algorithm. After obtaining the data, the user can manage the progress of the software project according to the actual situation of the current company. The contents of the form are as follows:

March - April 2020
ISSN: 0193-4120 Page No. 307 - 321

result

	max_data	mean_data	median_data	min_data	optimal_data
req	35	19.2222222222222	19	5	19.2063492063492
des	36	21.3737373737374	23	6	21.381364667079
dev	40	20.8484848484848	21	5	20.8147804576376
test	20	13.4141414141414	14	6	13.4225932797361
rel	15	10.7777777777778	11	6	10.7834467120181

Figure 32: The table results

5. FINDINGS & DISCUSSION

5.1. THE FINDINGS OF THE ORIGINAL DATA

The following figure is a scatter plot generated by python using raw data. The x-axis of the graph represents the age of the respondent, and the y-axis represents the time required to complete the requirements analysis phase. In the figure, it can be seen intuitively that the longer the length of time the respondent is working, the shorter the development time. This is very realistic, because the development experience is so rich that the development work will be more efficient.

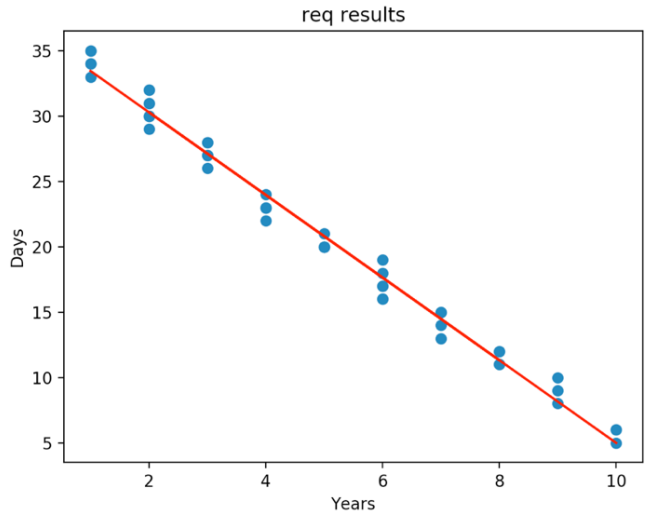

Figure 33: The results of req data

The following figure is a linear graph generated by using python from raw data. It can be seen from the figure that the original data is almost irregular, and the values of the data are also scattered. The maximum value is taken to be 40, and the minimum value is taken as 5. Through these raw data, it can be concluded that the professional level of the questionnaire object is not uniform, which can also be linked to the current level of professionalism of employees of small and medium-sized companies. Therefore, it can be considered that the data obtained through this questionnaire is informative.

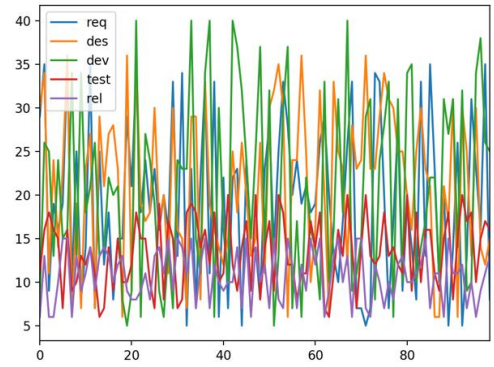

Figure 34: Linear graph of original data

Using python, take the maximum, minimum, average, and intermediate values of the five stages of software development and sort them into the following table:

Table 2: Original data results

	Max value	Min value	Mean value	Medium value
Requirements step	35	5	19.222222	19
Design step	36	6	21.373737	23
Development step	40	5	20.848485	21
Testing step	20	6	13.414141	14
Release step	15	6	10.777778	11

Through the comparison of the maximum and the minimum, it is more intuitive to understand that the professional level of the practitioners in the software development industry is still uneven.

March - April 2020
ISSN: 0193-4120 Page No. 307 - 321

Developers with a high level of professionalism can complete a development phase in a short period of time, and developers with a relatively low professional level need more time to complete a certain development phase. In the current small and medium-sized software companies, due to the limited funds, junior programmers must be hired to assist in the completion of the project, so this aspect should be taken into account in the software project schedule management. Establish a reasonable time to avoid the phenomenon that the quality of the product will decline due to the progress of a certain development stage.

The mean directly reflects the average of the subjects surveyed in this study, but the average is susceptible to the maximum and minimum values. From the average data, the data results are much higher than the minimum, which shows that the highly specialized programmers in this survey are only a small part. As shown in the figure below, the cutting-edge talents of all walks of life are only a small part of this industry, and most people need to be considered when planning the project schedule management.

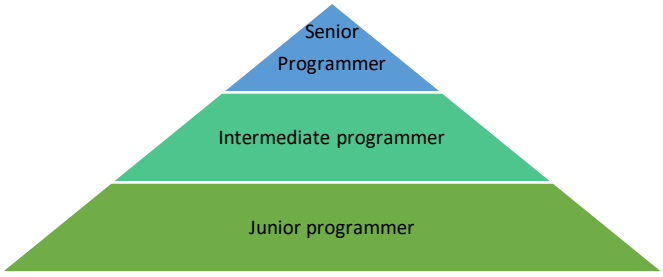

Figure 35: Programmer level chart

The intermediate value can intuitively reflect the intermediate level of the survey object. The data of the intermediate value in the table is roughly similar to the data of the average value, which can affirm the intermediate level of the data to a greater extent.

5.2. THE FINDINGS OF PROCESSED DATA

Since the extreme value in the original data has a relatively large influence on the overall data, this study needs to simply process the original data, remove a maximum value and a minimum value of each set of data, and then calculate the average value of the remaining samples. The specific calculation formula is as follows:

The advantage of using removing a maximum and a minimum and then averaging is to reduce the effect of the extremes on the average, which is often the case in our lives, such as ten judges giving a speaker for the score, the final result of the speaker is also to remove the highest score and the lowest score and then average the score. The scores thus obtained are fair and equitable, because the judges' scores sometimes have personal factors, and the scores are too high or too low, and the average score will eventually affect the overall score. This factor also needs to be excluded in this research, and the average of the overall data will not be affected by the data filled by a single investigator. This will increase the confidence of the average. If the sample data is larger, consider removing more extremes, such as removing three maximums and three minimums, or more. This needs to be determined based on the number of samples.

After calculating all the optimized data, use Python to organize all the data into a bar chart, as shown below:

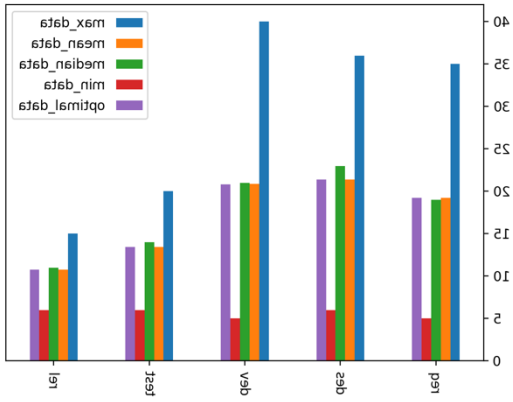

Figure 36: Bar chart generated using python

The bar chart can visually reflect the specific data in each group, and it is easy to compare the differences between the data. As can be seen from the figure, the median, average and optimized data

are approximate. The project management personnel can refer to this data to determine the intermediate level of the time required for each opening phase, and then combine the actual management of the project schedule.

After calculating the optimized data in a stage, continue to use Python to draw the optimized data of each stage into a pie chart, as shown below:

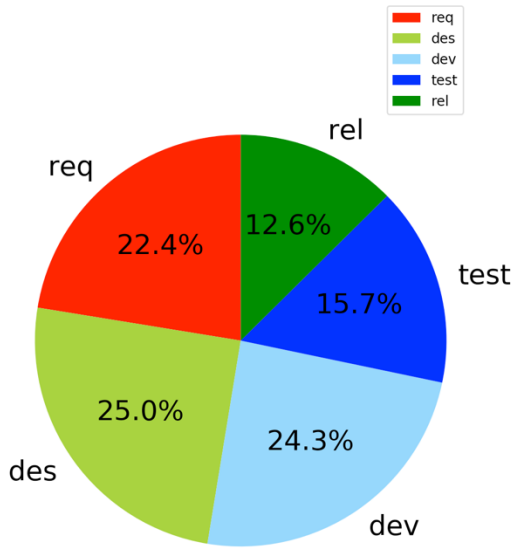

Figure 37：Pie chart generated using python

In the pie chart, it can be seen intuitively that the five steps of software development account for the total time. The most occupied time is the design phase, and the least is the release phase. It can also be seen from the figure that the demand analysis phase, the software design phase and the software development phase occupy almost the same time. The project manager can combine the total time of each step in the pie chart with the optimized average data in the bar chart to customize a reasonable management plan for the project and customize the reasonable project delivery time. Some project customers have already set the delivery time for the development company. This total time cannot be changed. The project manager is ready to allocate this time reasonably. You can refer to the pie chart generated above and combine the team. The actual ability of the members to allocate time to the five phases of software development ultimately enables the product to be delivered within the time specified by the customer.

6. Conclusion

The main objective of this research on project schedule management intends to be still based on the management function and cannot give users appropriate advice. Applying machine learning technology to software project schedule management, it can give users appropriate advice, which can help managers reduce the difficulty of management decision-making. This research uses a linear regression model in machine learning to create a predictive model that can be used to predict the development time required by each developer. Data analysis can then provide effective advice to software project managers. In this research, it still has some shortcomings, too few data samples, and the results of the prediction model are not accurate enough. Second, there is no complete automation and manual data collection and analysis is required. I believe that in the near future, machine learning technology will make a qualitative change in project schedule management.

REFERENCES

[1] M. E. R. Wang Wei, "Software Project Schedule Management Using Machine Learning & Data Mining," INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH, vol. 8, no. 09, 2019.

[2] A. Ahmed, Software Project Management A Process-Driven Approach, New York: Auerbach Publications, 2016.

[3] O. K. B. B. Frank F. Tsui, Essentials of Software Engineering, kennesaw: JONES & BARTLETT LEARNING, 2016.

[4] E. Camilleri, Project Success Critical Factors and Behaviours, London: Routledge, 2016.

[5] T. L. Young, Successful Project Management, Kogan Page, 2016.

[6] E. P. S. B. I. S. A. S. Andreou, "An investigation of effort distribution among development phases: A four-stage progressive software cost estimation model," journal of software, 2017.

March - April 2020
ISSN: 0193-4120 Page No. 307 - 321

[7] O. S. J. R. J. W. Pekka Abrahamsson, "Agile Software Development Methods: Review and Analysis," Cornell University, 2017.

[8] P. ,. M. H. ,. K. ,. P. P. Rajesh H. Kulkarni, "Investigating Agile Adaptation for Project Development," International Journal of Electrical and Computer Engineering, vol. 7, no. 2088-8708, pp. 1278-1285, 2017.

[9] V. H. Vitae, " The Resonance Of Machine Intelligence: Implications For Now And Into The Future For The World Of The Orthodox Human," honors theses, no. 192, 2018.

[10] R. J. G. Randall Englund, Creating an Environment for Successful Projects, Business & Economics, 2019.

[11] L. K. RashinaHoda, "Multi-level agile project management challenges: A self-organizing team perspective," Journal of Systems and Software, vol. 117, pp. 245-257, 2016.

[12] J. M. L. GabrielCepeda Carrión, "Prediction-oriented modeling in business research by means of PLS path modeling: Introduction to a JBR special section," Business Research, vol. 69, no. 10, pp. 4545-4551, 2016.

[13] J. Frank E. Harrell, Regression Modeling Strategies: With Applications to Linear Models, Logistic and Ordinal Regression, and Survival Analysis, New York: Springer, 2015.

[14] Z. Ghahramani, "Probabilistic machine learning and artificial intelligence," Nature International journal of science, vol. 521, pp. 452-459, 2015.

[15] D. A. I. Dr. Krish Narayanan, "The Software Development Life Cycle and Its Application," SENIOR HONORS THESES, vol. 589, 2018.

[16] DaDa, "How long does it take to develop an app?," 06 02 2018. [Online]. Available: https://36kr.com/p/5117575.

[17] D. Sarkar, "Text Analytics with Python," in Python Refresher, Apress, 2016, pp. 51-106.

[18] T. L. Young, Successful Project Management, Kogan Page Publishers, 2016.

[19] F. G. K. J. O. Howard Lei, A statistical analysis of the effects of Scrum and Kanban on software development projects, Elsevier, 2017.

[20] T. M. M. M. I. Jordan, "Machine learning: Trends, perspectives, and prospects," Science, pp. 25-30, 2015.

[21] S. M. J. Riana Steyn, "The Use of a Learning Management System to Facilitate Student-Driven Content Design: An Experiment," in International Symposium on Emerging Technologies for Education, Emerging Technologies for Education, 2017, pp. 75-94.

[22] E. F. M. A. H. Ian H. Witten, Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann, 2016.

[23] E. G. W. M. S. William S. Cleveland, "Local Regression Models," in Statistical Models in S, New York, Routledge, 2017, p. 68.

[24] J. Fox, Applied Regression Analysis and Generalized Linear Models, SAGE, 2015.

[25] Y. Huang, "Applied Research on the Progress of Software Development Projects," silicon valley, ShenZhen, 2011.