CHAPTER-1

INTRODUCTION

Effort estimation is the process of forecasting how much effort is required to develop or maintain a software application. This effort is traditionally measured in the hours worked by a person, or the money needed to pay for this work.

Effort estimation is used to help draft project plans and budgets in the early stages of the software development life cycle. This practice enables a project manager or product owner to accurately predict costs and allocate resources accordingly.

Software cost estimation is one of the most challenging tasks in software engineering. Over the past years the estimators have used parametric cost estimation models to establish software cost, however the challenges to accurate cost estimation keep evolving with the advancing.

This project uses Back-Propagation neural networks for software cost estimation. A model based on Neural Network has been proposed that takes the KLOC of the project as input, uses COCOMO model parameters and gives cost as output. Artificial Neural Network represents a complex set of relationships between the effort and the cost drivers and is a potential tool for estimation. The proposed model estimates the software cost and helps project manager to provide fast and realistic estimates for the project effort and development time that in turn gives software cost.

This dataset contains numerical values as well as categorical data. Within this dataset there are a high percentage of missing values. Due to this, the data mining techniques are used for pre-processing the information. The size estimation is the measuring of the project size, usually in lines of code or equivalent. Since software is a product without physical presence and the main cost is the design and development of the product, the cost is dominated by the cost of the human resources, measuring this effort in man-months.

Finally, the schedule estimation is the amount of time needed to accomplish the estimated effort, considering the organizational restrictions and the parallelism between project tasks. At the end of the process, we can get an economical value for the project cost, multiplying

the number of man-month estimated by unitary cost. So, the project estimation is a forecast of the expected effort to develop a project and the scheduled needed to accomplish it. Because the complexity and variety of factors influencing over the accuracy of the effort estimation, we need to develop analytical models that take consideration of every factor.

CHAPTER-2

LITERATURE SURVEY

2.1 Critical Analysis of the literature

We have considered 20 IEEE transactions for the literature survey published between the year 2019 and 2022. We have included journals based on various artificial intelligence techniques and datasets used. Software Cost Estimation is a hot area of research in the field of development of software projects. Precise estimation of the efforts put on development of software projects, in term of person-month and development time, is an essential earlier startup of projects.

Software project development has become extremely complicated, and the necessary competence in this industry is high, which requires the skills of highly qualified people. In past decades, to complete a project and deliver it to the customer on time, schedule, and budget, project managers had to estimate the cost of the software product, effort, and project duration or defect density. Developing a reliable parametric cost model at the conceptual stage of the project is crucial for project managers and decision makers. Existing methods, such as probabilistic and statistical algorithms have been developed for project cost prediction. However, these methods are unable to produce accurate results for conceptual cost prediction due to small and unstable data samples. Artificial intelligence (AI) and machine learning (ML) algorithms include numerous models and algorithms for supervised regression applications.

The individual analysis of the various proposed systems has given us a lot of insights to frame the design our proposed model. We also came across various performance evaluation metrics to compare the models.

We came across different technologies such as Artificial neural network[ANN],deep learning and meta-heuristic algorithms used to improve the efficiency of the cost prediction model. Some models have been trained and validated over a large number of datasets but performance evaluation is done using a smaller number of metrics.

Some models were not compared with classic models and few systems have been trained and validated against small dataset. Considering all these factors we have designed our model by considering various evaluation metrics against classic models like COCOMO model.

2.2 A Summary table

Authors		Approach	Description	Reaults
[1]	JUNAID ALI KHAN, SAIF UR REHMAN KHAN, TAMIM AHMED KHAN, AND INAYAT UR REHMAN KHAN IEEE - June 2021	An Amplified COCOMO-II Based Cost Estimation Model in Global Software Development Context. .	The current study has proposed a conceptual framework to amplify the algorithmic COCOMO-II model in the GSD domain to accommodate additional cost drivers empirically validated by a systematic review and industrial practitioners.	The results indicate that the model needs further calibration and validation.
[2]	Amrita Sharma and Neha Chaudhary. IEEE - 2020	Linear Regression Model for Agile Software Development Effort Estimation	In this paper, the multiple linear regression model linear regression models are proposed for comparing the best model for agile development. The correlation between the dependent and independent variables are also found out.	The results showed that the proposed model outperforms from the decision tree, stochastic gradient boosting, and random forest.
[3]	Suyash Shukla, Sandeep Kumar and Pravas Ranjan. IEEE - 2019	Analyzing Effect of Ensemble Models on Multi-Layer Perceptron Network for Software	In this paper, MultiLayer Perceptron (MLPNN) and its ensembles are explored in order to improve the performance of	In this work, five different machine learning techniques have been compared for the estimation of effort required to develop any software. For the given datset, it has been

		Effort Estimation	software effort estimation process. Firstly, MLPNN, Ridge-MLPNN, Lasso-MLPNN, Bagging-MLPNN, and AdaBoost-MLPNN models are developed and, then, the performance of these models are compared on the basis of R2 score to find the best model fitting this dataset.	observed that only a minor improvement is achieved in R2 score by using ensembles of MLPNN. The R2 score of AdaBoost-MLPNN is 82.213% , which is highest among all the models, whereas the R2 score of MLPNN is 78.3%.
[4]	Shofiyah Al Idrus, Wahyu Nur Hidayat and Achmad Hamdan. PNBP 2020	NAYA as a Tool of Software Cost Automatic Analysis.	The COCOMO II and UCP methods have a long steps calculation process with a lot of data collection, this obstacle is the main factor in the cost calculation process in the software and also, the lack of information about software cost calculation methods and techniques makes this section often overlooked by software developers.	Cost calculation with UCP has the advantage of implementing the Re-UCP method so that the components used are very comprehensive and complete so that it can remind UCP value and effort. NAYA applications can help software developers to calculate and arrange to finance for a software project quickly and accurately..
[5]	Muhammad Sufyan Khan, CH Anwar ul Hassan, Munam Ali Shah and Azra Shamim. IEEE 2019	Software Cost and Effort Estimation using a New Optimization Algorithm Inspired by Strawberry Plant.	In this paper, we are using a new meta-heuristic algorithm inspired by the strawberry plant for optimization of COCOMO effort estimation method. NASA 93 data set is used in the proposed approach.	The Magnitude of Relative Error (MRE) and Mean Magnitude of Relative Error (MMRE) is evaluated. Experimental results of the proposed method with the COCOMO model shows a decline in MMRE to 23.8%.
[6]	Suyash Shukla and Sandeep Kumar.	Applicability of Neural Network Based	The performance of LR, Support Vector Machines (SVM), KNN, and MLPNN	The results of these models are compared in order to find which model is more feasible

	IEEE 2019	Models for Software Effort Estimation	has been evaluated and it is observed that the MLPNN method is better than the other three models used in this study. The proposed methodology used in this paper consists of four machine learning models namely, LR, SVM, KNN, and MLPNN.	for the estimation of software effort. In addition to these models, we have used the Pearson correlation coefficient to find out the relationship between different attributes and software effort in Desharnais dataset and we have found that the Length, Transactions, Entities, Point Adj, and Non-Point Adjust are the most influential attributes in comparison to other attributes for effort estimation.
[7]	Faiza Tahir and Mahum Adil. IEEE 2018	An Empirical Analysis of Cost Estimation Models on Undergraduate Projects Using COCOMO II.	This paper will present the use of two techniques i.e., expert judgment and Constructive Cost Model (COCOMO) II to enhance the predictability of COCOMO III for the undergraduate projects of universities It calculates the size through function point, use case points or object points metrics and converts them in kilo line of code (KLOC).	This paper has studies the final year projects of undergraduate students of universities of Pakistan. The scheduled time for these projects is usually one year from its inception till deployment. Effort estimation of these projects is conducted through the use of tool Estimator.
[8]	Aman Ullah, Bin Wang, Jinfan Sheng, Jun Long, Muhammad Asim and Faiza Riaz. ICICAS 2019	A Novel Technique of Software Cost Estimation Using Flower Pollination Algorithm.	This study mainly focuses on the optimization of the Constructive Cost Model (COCOMO)-II parameters by implementing Flower Pollination Algorithm (FPA) on standard datasets	The proposed method efficiently optimizes the coefficient of COCOMO-II like A & B for better cost estimation by considering MMRE and MD as a fitness function.

			taken from Turkish Industries software project.	
[9]	ANFAL A. FADHIL , RASHA G. H. ALSARRAJ, AND ATICA M. ALTAIE UM 2020	Software Cost Estimation Based on Dolphin Algorithm.	DOLPHIN BAT ALGORITHM (DOLBAT) PROPOSED TO ESTIMATE SOFTWARE EFFORT In this work has been using dolphin algorithm and Hybridized with bat algorithm to get better coefficients value to predict effort of software, in the initial phase in dolphin algorithm it configure three matrices randomly.	By applying the proposed algorithm the value of MMRE equal (50.2757) for NASA-93, and the value of MMRE equal (14.576) for NASA-60, as has compared with the previously has applied algorithm the proposed algorithm has the lowest MMRE.
[10]	Vishnu Sai Desai and Vishnu Sai Desai. CICT 18	ANN-Cuckoo Optimization Technique to Predict Software Cost Estimation	They have used Artificial Neural Networks with Cuckoo Optimization Algorithm to present as a new model for prediction of Software Cost Estimation We verified this technique on the International Software Benchmarking Standards Group (ISBSG) licensed dataset, which contains 1531 projects.	They have used Artificial Neural Networks with Cuckoo Optimization Algorithm to present as a new model for prediction of Software Cost Estimation We verified this technique on the International Software Benchmarking Standards Group (ISBSG) licensed dataset, which contains 1531 projects.
[11]	Ming Qin, Lianteng Shen, Dahua Zhang ICICAS 2019	Deep Learning Model for Function Point Based Software	The advantage of this method is that this method takes valuable experience of expert, which can suitable for more	A novel approach is proposed in this paper to improve function point analysis function point analysis task in software development.

		Cost Estimation - An Industry Case Study	complex scenario and more general topic it depends too much on the expert, which lacks standardization and can be subjective Challenges raised in Function Point (FP) analysis have been described in many research and industry practice, there are two most obvious barriers to conduct FP analysis, and the first is the rule to define the definition of different type of function point	Experiments shows that the proposed method achieves state-of-the-art on the field of function point analysis function point analysis.
[12]	Kento Korenaga Graduate School of National Science and Technology	Data Smoothing for Software Effort Estimation	This paper proposed a method called data smoothing to mitigate the problem of outliers without reducing the number of data points in a data set The proposed method follows the assumption of Analogy-Based Estimation (ABE) such that "projects with similar features require similar development efforts," and changes the effort values in a data set so as to satisfy this assumption.	They are aware that our experimental result is not strong enough because the significant improvement was seen in only one data set, and we compared data smoothing only with Cook's distance-based outlier removal method, which is a very basic method This is the first step to propose and implement the idea of data smoothing, and we believe that the idea itself has a great potential to improve machine learning in many domain because it can mange with outliers without reducing the number of data points
[13]	Riyanarto Sarno, Johannes	Comparison of Different	This cost drivers consist of Effort	This research has proved that neural network could

	Sidabutar, and Sarwosri. 2015 International Conference on Computer, Control, Informatics and Its Applications	Neural Network Architectures for Software Cost Estimation	Multiplier (EM), Scale Factors (SF), and Line of Code (LOC) This paper shows literature review which describes any information that can help readers to understand the research. It consists of software cost estimation, Constructive Cost Model (COCOMO), and artificial neural network description.	increase the accuracy of COCOMO for both COCOMO 81 dataset and NASA 93 dataset The result is the proposed model increases the accuracy and each model has different result.
[14]	Wilson Rosa, Raymond Madachy, Barry Boehm and Bradford Clark.	.Early Phase Cost Models for Agile Software Processes in the US DoD.	In the United States Department of Defense (US DoD), it is necessary and most critical to estimate software development cost in early lifecycle phases when limited data is available. The model accuracy improves after peak staff and super domain are incrementally added to the model.	Since the data was collected at the Computer Software Configuration Item (CSCI) level, the resulting models may not be appropriate for projects reported at the aggregate level due to the excluded cost of subsystem integration This study only examined the impact of software requirements, peak staff, and super domain on development effort. .
[15]	Safa Mohammed Ahmed Suliman and Gada Kadoda. SCCSIT 2017	Factors that Influence Software Project Cost and Schedule Estimation	As can be drawn from global, regional and local experiences with software project estimation, there is no one overriding factor that causes project failure or overrun, a number of factors are involved in any particular project failure, some	This paper highlighted factors that influence software projects management and sometimes cause the failure of a software project, in particular the paper focused on cost and schedule overrun that are important for software project estimation theory and

			of which interact with each other.	practice
[16]	Saurabh Bilgaiyan, Samaresh Mishra, Madhabananda Das 2016 International Conference on Computational Intelligence and Networks	A Review of Software Cost Estimation in Agile Software Development Using Soft Computing Techniques	Software Project Management (SPM) included a series of interrelated processes which were to be carried out in a specific manner with the development process Recent research and practical outcomes of different software developments have proved that prediction of cost and effort estimation with a high rate of accuracy increases the chance of successful quality product.	This paper presents a systematic literature survey on effort and cost estimation techniques for agile software development by means of soft computing techniques The authors have found that very less work is done in the field of cost and effort estimation of ASD projects using soft computing techniques
[17]	Imam Kurniawan, Arry Akhmad Arman, Sukrisno Mardiyanto School of Electrical Engineering and Informatics Institut Teknologi Bandung Bandung, Indonesia 2019 i	Development of Analogy-Based Estimation Method for Software Development Cost Estimation in Government Agencies.	Analysis and designing using analogy based estimation method, that are Building a new dataset using data collection of previous software projects, Proposing the K-Nearest Neighbor (KNN) with mean and Inverse Distance Weight (IDW) interpolation technique for the effort calculation method, and Designing the cost allocation model based on the regulations of goods and service procurement in	Dataset evaluation for effort estimation using the proposed analogy method shows the best results of Mean Magnitude Relative Error (MMRE) of 0.36 and PRED(0.25) of 0.57 The evaluation of cost estimation from three software projects that were conducted in government agencies have percentages of deviation values of 12.29%, 5.85%, and 3.97% with an average of 7.37%

			government institutions in Indonesia.	
[18]	Mustafa Hammad and Abdulla Alqaddoumi. 3ICT 2018	Features-Level Software Effort Estimation Using Machine Learning Algorithms	The main goal of this study is to evaluate software effort estimation models that are built based on machine learning approaches This paper presents various software effort estimation models using four machine learning algorithm.	Results showed that machine learning approach can be applied to predict the software effort with low Mean Absolute Error (MAE) value Applying filters in the software features data may increase the prediction accuracy due. This can be done by understanding the sensitivity of the prediction model for each dataset.
[19]	Hitesh Kumar Sharma, Ravi Tomar and Ankur Dumka. NGCT-2016	E-COCOMO: An effort estimation model for cleanroom software development approach	The traditional COCOMO need some extended feature for accurate calculation of efforts in these new approaches Bohem evolve the new version of COCOMO in which he considered some more factors involved in effort estimation Detailed COCOMO: Detailed COCOMO focuses on phase wise effort calculation for different phases in SDLC.	We have identified that there is a need of inclusion of the one cost driver i.e. Formal Method Knowledge Capability(FMKC) In the future work the tool will be evolved with some new parameters for some new approaches like Agile Development, Component based design
[20]	Shahab Nadir, Christina Burggraf and Detlef Streitferdt.	Industrial software development s effort estimation model	In order to develop a system which improve the metrics for automotive industry, the square's devil above should be taken in account	In this research They developed an expert system which helps in effort estimation of industrial software change. The support model

			Automotive software is safety-critical so product quality is high priority [3] The iterations during the development cycle will involve many change requests: some to add or edit requirements and features, and others to correct defects which originated in a different phase of the cycle.	describes the system and support the managers with a good overview over the project, and support the project managers (PrjM) with all Key Performance Indicators (KPI) which effected the effort for each CRQ

2.3 Implication and conclusion

Based on the detailed literature survey we infer that we will train and validated our model against ISBSG dataset. The cost prediction model will be COCOMO.

CHAPTER 3

EXISTING SYSTEM

All the methodologies of project management make management of plan and costs in any type of project and in the projects of software. The chosen system to make the estimations has to have the confidence of the project management and to allow to adapt again to the changing necessities of the software.

Nowadays, there are several models available of cost estimation like COCOMO model, Checkpoint, ESTIMACS, SLIM, Knowledge Plan, etc.). Among all of them, COCOMO model is one of most open and well-documented cost estimation models. At present, most of real-world use of cost models is bottom-up rather than top-down.

The historical data summary in the end of the project is essential to update the data base of projects and so that the system can fit its parameters to the changing conditions of software.

There are various hybrid systems proposed which is compared and evaluated against basic models. Few of the research teams have conducted a comparison by investigating 20 different AI techniques but considering only very few performance metrics.

Some models are trained against randomly selected projects from a dataset consisting few projects. Some hybrid models are not compared with classic COCOMO models to prove hoe reliable they are for real world implementation.

Also many models have been trained and validated against standard dataset by ignoring how heterogenous the data is and proper feature selection is not done.

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 Functional Requirements

- Usability: The model should be able to predict effort and cost estimate of all categories of systems and projects
- Reliability: The effort estimated by the model must be accurate and reliable which in turn helps the project managers to proceed.
- Performance: The system should be consistently responsive and with a good performance.
- Robust: To handle complex situations and provide results
- Efficiency: The system should work efficiently while gathering data

4.2 Software Requirements

- Python (3.7.4)
- IDE (Jupyter)

4.3 Hardware Requirements

- Processor: Minimum 1 GHz; Recommended 2GHz or more
- Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)
- Hard Drive: Minimum 32 GB; Recommended 64 GB or more
- Memory (RAM): Minimum 1 GB; Recommended 4 GB or above

CHAPTER 5

ANALYSIS AND DESIGN

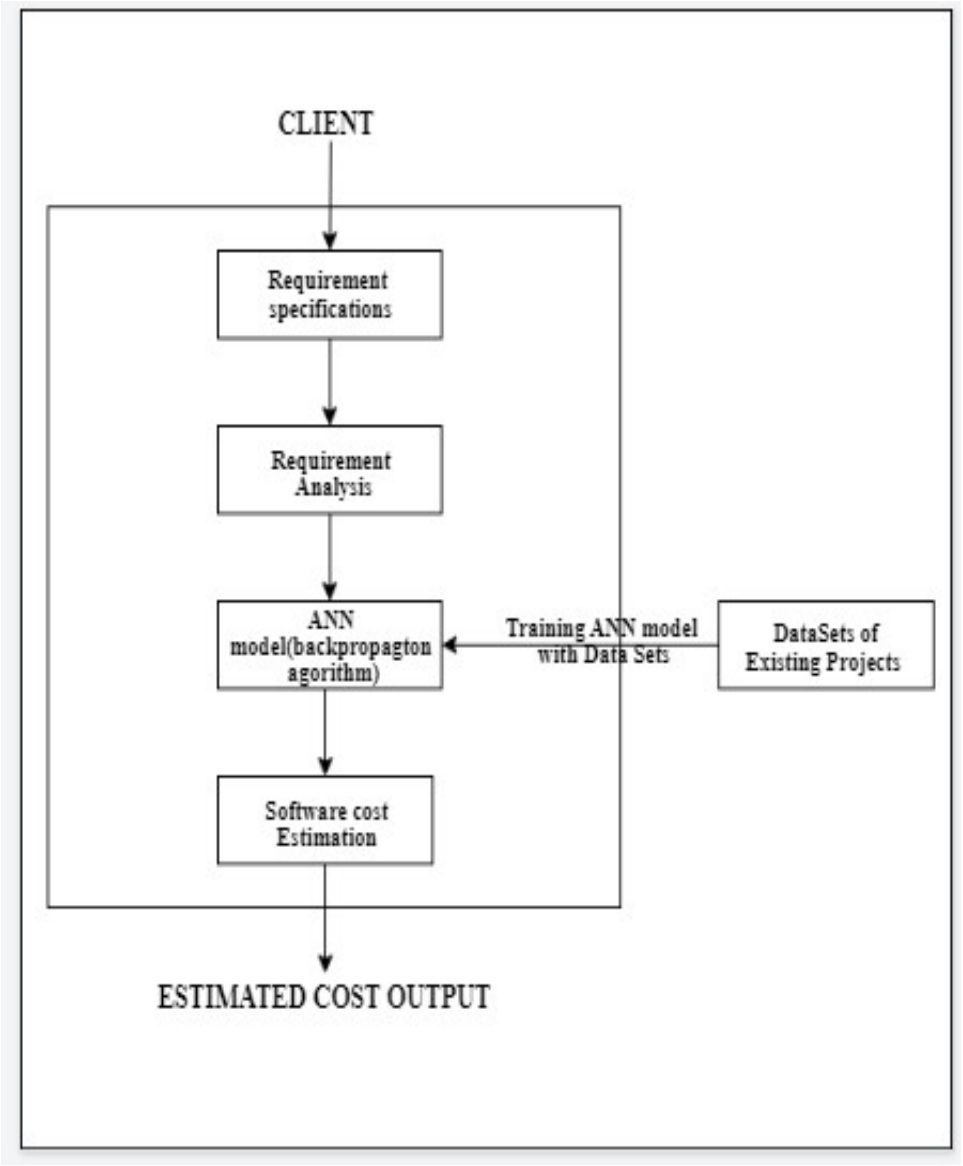

Fig.1: Block Diagram

Steps Involved in estimating the software cost through this project:

- **Step 1**: Getting the requirement specification from the client
- **Step 2**: Analyzing the processing the requirement specifications given by the client
- **Step 3**: Feeding it the created software which is trained with datasets(These data sets contain the already created software details which include the overall cost of that software)
- **Step 4**: The software processes the given data with the back propagation algorithm and the cost is estimated and given as output

5.1 DATA PRE-PROCESSING

It is a data mining technique that transforms raw data into an understandable format. Raw data (real world data) is always incomplete and that data cannot be sent through a model. That would cause certain errors. That is why we need to preprocess data before sending through a model.

Following are the steps that are required to handle missing values for categorical data by most frequent occurrence of element in a column. In machine learning, we usually deal with datasets which contains multiple labels in one or more than one columns.

These labels can be in the form of words or numbers. To make the data understandable or in human readable form, the training data is often labeled in words. Encoding is a technique to solve the problem. Label Encoding refers to converting the labels into numeric form so as to convert it into the machine-readable form.

Machine learning algorithms can then decide in a better way on how those labels must be operated. It is an important pre-processing step for the structured dataset in supervised learning.

These are the steps for pre-processing of data –

1. Import libraries
2. Read data
3. Checking for missing values.
4. Checking for categorical data.

CHAPTER 6

METHODOLOGY

Learning By Human Brain:

The basic building block of the brain and the neural network is the neuron. The Basic human neuron adapted from (Beale and Jackson, 1990) is shown in Fig. .1 As described by (Beale and Jackson ,1990), all inputs to the soma (cell body) of the neuron arrive along with dendrites. Dendrites can also act as outputs interconnecting interneurons.

Mathematically, the dendrite's function can be approximated as a summation. Axons, on the other hand, are found only on output cells. The axon has an electrical potential. If excited past a threshold it will transmit an electrical signal. Axons terminate at synapses that connect it to the dendrite of another neuron. When the electrical input to a synapse reaches a threshold, it will pass the signal through to the dendrite to which it is connected. The human brain contains approximately 10 ° interconnected neurons creating its massively parallel computational capability.

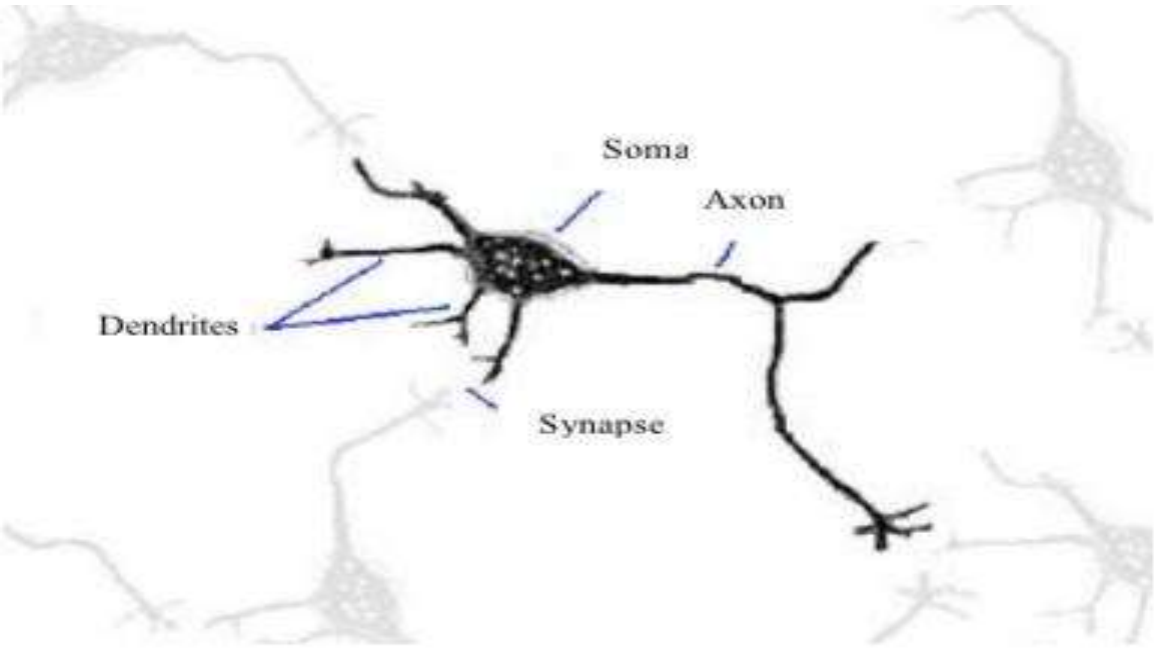

Fig.2: Components of Human Brain cell(Neuron)

From Human Neurons To Artificial Neurons:

The artificial neuron was developed in an effort to model the human neuron. The artificial neuron depicted in Fig. 2 was adapted from (Kartalopoulos and Stamatios, 1996) and (Haykin and Simon, 1994). Inputs enter the neuron and are multiplied by their respective synaptic weights. They are then summed and processed by an activation function. The activation function dampens or bound's the neuron's output. Figure 3 represent common activation functions which also happened to be used by the network tested during this research. The logistic or sigmoid function f(x) = 1/ (I+exp (-X)).

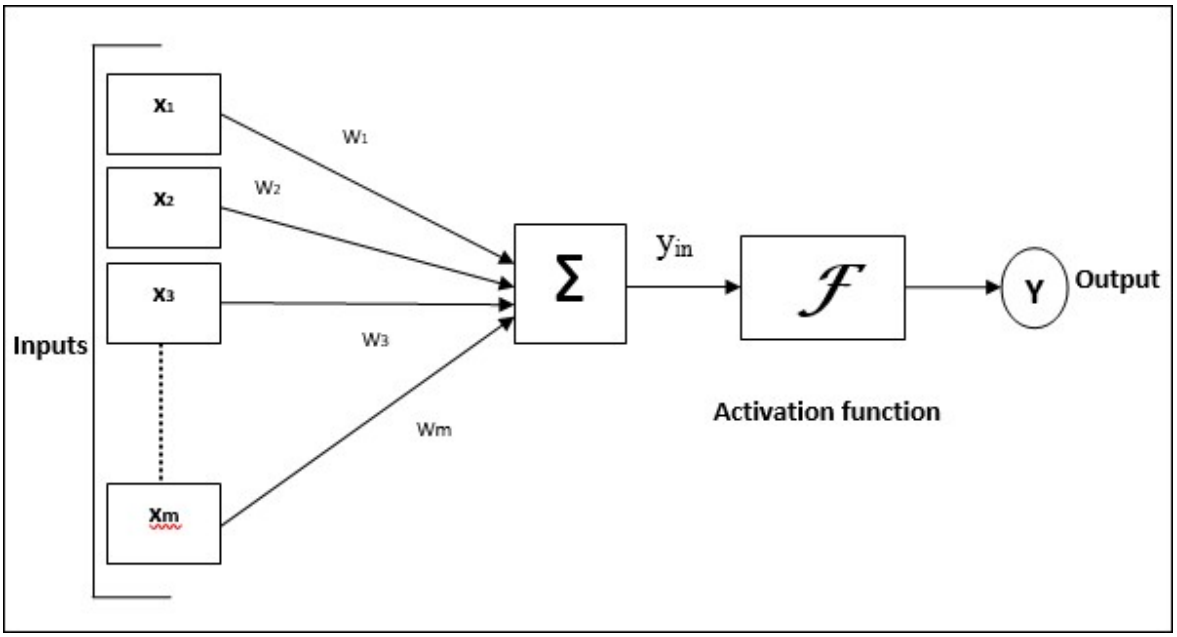

Fig.3:Artificial Neuron Model

To show the general technique of BP training, consider the neuron in the hidden layer H. This neuron has three weights feeding into it, WH.Wh2 and WH3. The training technique applied to this one neuron can be applied to all the other neurons throughout the network. First, an input is applied to the network and the output is calculated. The output is then compared with the target and an error is calculated.

This can be used to calculate Ab y multiplying learning rate ,I then calculate the new for neuron A, as follows Then propagate the value of 8 calculated for the output neurons, back through the weights, to the neurons of the hidden layer and hence calculate a value of 8 for these. This is done by multiplying the function.

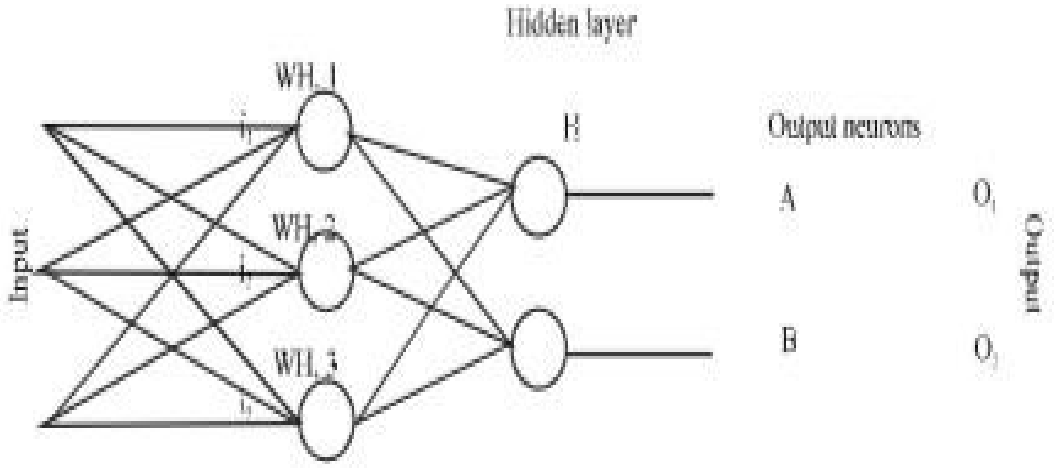

Fig.4: General Multilayer Feedforward Network.

CHAPTER 7

CONCLUSION

This study has compared the forecasting accuracy of neural networks with back propagation algorithm using one, two, three, four and five hidden layers. Results showed that for this software cost estimation problem, the neural networks with back propagation algorithms using two hidden layers has given most accurate prediction in comparison to other layers.Software development cost estimation is a challenging task for both the industrial as well as academic communities. The accurate predictions during the early stages of development of a software project can greatly benefit the development team.There are several effort estimation models that can be used in forecasting software development effort.

FUTURE ENHANCEMENT

The aim in the future is to increase the estimation accuracy of the model by training more datasets using hybrid machine learning models.

To develop a web application for proposed machine learning model.

CHAPTER 8

REFERENCES

[1] JUNAID ALI KHAN, SAIF UR REHMAN KHAN, TAMIM AHMED KHAN, AND INAYAT UR REHMAN KHAN. "An Amplified COCOMO-II Based Cost Estimation Model in Global Software Development Context." IEEE - June 2021, DOI: 10.1109/ACCESS.2021.3089870

[2] Amrita Sharma and Neha Chaudhary."Linear Regression Model for Agile Software Development Effort Estimation" IEEE - 2020, DOI: 10.1109/ICRAIE51050.2020.9358309

[3] Suyash Shukla, Sandeep Kumar and Pravas Ranjan."Analyzing Effect of Ensemble Models on Multi-Layer Perceptron Network for Software Effort Estimation" IEEE 2019, DOI 10.1109/SERVICES.2019.00116

[4] Shofiyah Al Idrus, Wahyu Nur Hidayat and Achmad Hamdan."NAYA as a Tool of Software Cost Automatic Analysis" PNBP 2020, DOI: 978-1-7281-8131-8/20/$31.00 ©2020 IEEE

[5] Muhammad Sufyan Khan, CH Anwar ul Hassan, Munam Ali Shah and Azra Shamim."Software Cost and Effort Estimation using a New Optimization Algorithm Inspired by Strawberry Plant" IEEE 2019.

[6] Suyash Shukla and Sandeep Kumar."Applicability of Neural Network Based Models for Software Effort Estimation" IEEE 2019, DOI 10.1109/SERVICES.2019.00094

[7] Faiza Tahir and Mahum Adil."An Empirical Analysis of Cost Estimation Models on Undergraduate Projects Using COCOMO II" IEEE 2018

[8] Aman Ullah, Bin Wang, Jinfan Sheng, Jun Long, Muhammad Asim and Faiza Riaz."A Novel Technique of Software Cost Estimation Using Flower Pollination Algorithm." ICICAS 2019, DOI 10.1109/ICICAS48597.2019.00142

[9] ANFAL A. FADHIL , RASHA G. H. ALSARRAJ, AND ATICA M. ALTAIE."Software Cost Estimation Based on Dolphin Algorithm." UM 2020, DOI: 10.1109/ACCESS.2020.2988867

[10] Vishnu Sai Desai and Vishnu Sai Desai. "ANN-Cuckoo Optimization Technique to Predict Software Cost Estimation" CICT 18

[11] Ming Qin, Lianteng Shen, Dahua Zhang. "Deep Learning Model for Function Point Based Software Cost Estimation -An Industry Case Study" ICICAS 2019, DOI 10.1109/ICICAS48597.2019.00165

[12] Kento Korenaga Graduate School of National Science and Technology. "Data Smoothing for Software Effort Estimation"

[13] Riyanarto Sarno, Johannes Sidabutar, and Sarwosri."Comparison of Different Neural Network Architectures for Software Cost Estimation" 2015 International Conference on Computer, Control, Informatics and Its Applications

[14] Wilson Rosa, Raymond Madachy, Barry Boehm and Bradford Clark. "Early Phase Cost Models for Agile Software Processes in the US DoD.", DOI 10.1109/ESEM.2017.10

[15] Safa Mohammed Ahmed Suliman and Gada Kadoda."Factors that Influence Software Project Cost and Schedule Estimation" SCCSIT 2017

[16] Saurabh Bilgaiyan, Samaresh Mishra, Madhabananda Das. "A Review of Software Cost Estimation in Agile Software Development Using Soft Computing Techniques " 2016 International Conference on Computational Intelligence and Networks, DOI 10.1109/CINE.2016.27

[17] Imam Kurniawan, Arry Akhmad Arman, Sukrisno Mardiyanto."Development of Analogy-Based Estimation Method for Software Development Cost Estimation in Government Agencies" School of Electrical Engineering and Informatics Institut Teknologi Bandung Bandung, Indonesia 2019,

[18] Mustafa Hammad and Abdulla Alqaddoumi."Features-Level Software Effort Estimation Using Machine Learning Algorithms" 3ICT 2018,

[19] Hitesh Kumar Sharma, Ravi Tomar and Ankur Dumka."E-COCOMO: An effort estimation model for cleanroom software development approach" NGCT-2016

[20] Shahab Nadir, Christina Burggraf and Detlef Streitferdt. "Industrial software developments effort estimation model", DOI 10.1109/CINE.2016.27