

# FRAMEWORK FOR MACHINE LEARNING BASED TASK ALLOCATION IN DASD ENVIRONMENT

Madan Singh<sup>1\*</sup>, Naresh Chauhan<sup>2</sup>, Rashmi Popli<sup>3</sup>

<sup>1</sup>Research Scholar, <sup>2</sup>Professor, <sup>3</sup>Assistant Professor

Department of Computer Engineering

J.C. Bose University of Science & Technology, YMCA, Faridabad

\*KIET Group of Institutions, Ghaziabad

**Abstract:** - Agile software development (ASD)[1], in today's time has turn out to be one of the main torrent methodologies. In order to hunt for bigger markets and cheaper labor, software development companies are switching towards Distributed development environments. In Agile terms this approach is well understood as distributed Agile software development (DASD). In DASD environment allocation of tasks has been an vicinity of concern because ad – hoc based approaches for task allocation may lead to customer dissatisfactions leading to project failures. Many research fellows have operated on different techniques for task allocation. In the paper, a novel approach for quantitative task allocation in DASD has been wished-for The approach seems to be promising for allocation of tasks in DASD environment.

**Keywords:** - Software Development (SD), Agile Software Development (ASD), Task Allocation (TA), Distributed Agile Software Development (DASD).

## INTRODUCTION

Little by little process to produce good quality products is an art termed as Software Engineering [1] which provides different procedures for software development. Traditional Software Development (TSD) [1] methods and techniques are the oldest among them. In this, development process streams through requirement analysis phase, design phase, implementation phase, coding phase, testing and maintenance phases one after the other. They focus on comprehensive planning, heavy documentation, and pre-defined requirements. These methods have been in demand for years due to the nature of being straightforward, methodical, and structured in nature. Key problem areas of these methods include requirement of complete set of customer needs prior to design, likelihood to be over budget, behind schedule and inflexibility to initiate changes as per

the customer demands at middle of development due to market needs or business values.

Agile Software Development (ASD) [1][28][29], is aa approach for software development where continuous customer communication and interactions are kept at top priority. There always exists scope of improvement in the project due to the presence of customer or one of his representatives with development team who always provide feedback according to the market trends or business values. A few of trendy agile methods are Scrum, Extreme Programming (XP), Crystal, and Adaptive Software Development (ASD) etc. Agile techniques can be classified as iterative and incremental developments that have adaptability to change throughout the systems development life cycle. In 2001, Agile Manifesto was created which emphasizes on the Agility based four principles. After lots of research it has been identified that agile methods have the capacity to deliver customer satisfaction, smaller development cycle, and a faster adjustment to fast changing business needs. So main emphasis remains on in -person communication and progresses of the project is measured by getting working sub - modules ready from the team and deliver them to the customer at the end of a time quantum termed as Sprint [3]. Agile methods rely on lesser documentation and better results.

Population explosion in twenty first century has caused migration of people from their native places to other places in the hunt of good higher education, better living hood and larger competitive markets. The locations of these markets depend on level of demand of different products globally. The software development industry also follows the same norms for its growth. In this process, high growth of software industry in recent years has been possible due to distributed software development strategies. The main reasons behind distributed development are – finding new customers around the world, reduction in development cost and effort, production of quality

software and expansion of personnel capacity [5]. The high expansion of software development companies due to distributed development has caused several difficulties like – deficiency of trained professionals, intricacy of coordinating and controlling projects, language, and time zone issues [6]. Task allocation is one of those compound activities. It is crucial for the realization of the project. The residual of paper is prepared as follows: - Section I provides introduction to three basic task - allocation approaches in DASD environment, in section II requirement of intelligent tools for automated task allocation has been analyzed based upon the study of literature survey, in section III machine learning based model for task allocation in DASD [14][15], environment has been proposed.

### **TASK ALLOCATION APPROACHES IN DASD ENVIRONMENT**

Task Allocation remains to be a decisive activity for each project. It becomes more critical in distributed environment. In general, task allocation is performed by three means – Ad – hoc manner, dedicated developer per story and story swarming [28][29][30][31]. In Ad – hoc approach, any task can be assigned to any developer, which he / she likes to do on that day based upon their interest, energy level and other factors. This approach improves flexibility with lesser possibility of any sort of deadlocked tasks. It is easier to accommodate absenteeism on a specific day and maximizes team accountability as all the members of the team are equally responsible for the complete project work and its associated tasks. The approach also addresses certain weaknesses like – inability to fit the best team members in all situations, increased number of backlog tasks and inability to directly fit any individual accountable. When everyone is responsible then it sounds like no one is responsible. In the approach termed as dedicated developer per story, a developer or a pair of developers is signed up for any individual story and are kept responsible for the completion of the story. This approach is least feasible as it demolishes any real feeling of team liability. Major advantages of the approach are that each developer or pair is at least accountable and responsible to complete at least one user story, and they understand overall design of the system. These sorts of approaches also work better with teams who cannot, or do not have to willingness

to collaborate. A distributed team across different time zones may be more effective with this method. It might also dress suit to those teams who are new to Agile based projects and work separately. The approach has disadvantages like – there exist no common goals or collaboration on day-to-day basis. If one of stories is blocked then the developer is also blocked and it cannot afford much absenteeism i.e., due to sickness, vacations, production emergencies etc.

In story swarming strategy, each team member swarms each story. Though it is not feasible for all the team members to swarm each story, and to work only on a single story, the number of stories can be reduced to two or three at a time. Advantage of the approach is increased level of accountability. One more major benefit of the technique is that the problem of several stories being finished at the same time and at the end of the iteration can be avoided. It requires higher collaboration among team members and can accommodate unexpected absenteeism. The approach also reduces the bunch of almost but not quite ready or half complete stories. The approach can also handle iterations with every iteration delivering a single or bunch of stories. The approach also involves some disadvantages like – addressing more than one story at a time may involve risk of late delivery of some of the stories.

### **LITERATURE SURVEY - REQUIREMENT OF AN INTELLIGENT TOOL FOR TASK ALLOCATION**

Besides these techniques there exist certain factors which drive task allocation which lie under organizational, personal, environmental factors and other factors like socio – cultural factors [14]. Plentiful studies on task allocation in DASD environment have been conducted aiming to refine the process to obtain improved results in stipulations of project managing faster delivery, customer satisfaction etc. Shen M. et al., 2003, [2], projected task assignment model with multiple criteria within workflow management system. Their model used fuzzy numbers-based scale which describes qualitative dealings like person based capabilities with linguistic scales. The model is flexible and easily comprehensible. Shim J. et al., 2003, [3], projected methodology that is model-based and it allows sculpting of task assignment strategies. A strategy is required to determine to allocate a task to

a specific person or set of persons. Their line of attack uses UML based Meta models. The UML based models are used for describing four essentials of task assignment – “organization, process, work product and project”. Duggan J. et al., 2004, [4], projected an optimized method. The method was used for allocating tasks through inception and construction phase of software development higher productivity, good quality and less time frame. They used multi - objective evolutionary algorithm and genetic algorithm. Genetic algorithm includes fitness assignment, crossover, and mutation phases. They applied genetic algorithm to task assignment problem. While the technique is effectual to estimate price and fix schedules, there may be problems when applied to execute allocation of tasks during the implementation of project. Mak D.K.M. et al., 2006, [5], projected an approach to resolve task allocation problems faced in Agile based DSD environment. The process uses analytic hierarchy process (AHP) methodology. It is very useful for multi criteria assessment in which decision maker and project leaders are needed to examine certain factors like - quantitative i.e. time and resources as well as qualitative i.e. team member skills and experience based factors. Task coordination and allocation model firstly prioritizes set of tasks then finds best person for allocation. A suitably ranking team member is allocated with each available task. Lamersdorf A. et al., 2008, 2009, 2010[6][17][18], analyzed the approaches used for task allocation and proposed a set of parameters for it. The authors suggested a decision support system to be used in task allocation. The model considered many criteria for making decisions. They identified among goals of development projects i.e. cost, time and quality as well as distinctiveness for dispersed development that has a blow on the goals. The authors urbanized a model named as TAMRI (Task allocation based on multiple criteria). The model is used for decision support in task allocation. It makes reuse of a technique of distributed systems as well as Bayesian networks in order to put forward an ordered model. Marques A. B. et al., 2013 [7], projected a domain ontology based approach in order to show allocation of tasks in dispersed teams. The anticipated ontology was useful for giving a common thought about task allocation aspects to teams. The ontology brought alertness to project managers about the factors related

to task allocation planning. Lin J. et al., 2013, 2014 [8][9], identified experiential conclusion task allocation and making decisions along with teamwork and team self-esteem. The results were based on twelve week long coursework project having one hundred twenty five undergraduate, software engineering students working in twenty one scrum teams on Agile platform module HASE. Using the platform HASE, the authors address task allocation decision making, collaboration, team morale like issues empirically.

Filho M.S. et al., 2015, 2016.2017 [23][24][25][26], projected allocation of task set to team members residing in remote locations as a major challenge where project manager has to take in to account technical expertise and cultural similarities under consideration. Verbal decision analysis method ZAPROS III-I had been used by authors for rank ordering eleven parameters that project managers used to keep in their mind at the time for task allocation. Verbal decision analysis uses multi criteria problem solving mechanism which takes in to account criteria’s subjectivity. Task allocation is performed quantitatively to the team members. Masood Z. et al., 2017[11], proposed different techniques for assessing flow of work across teams like- “team independent, team dependent, skill set / module dependent, hybrid workflow”. They identified five types of task allocation strategies namely “individual-driven, manager-driven, team-driven, manager-assisted, team- assisted”. The distributions are helpful for project managers in making successful distribution and allocation of tasks. Adnan M. et al., 2017[12], proposed a model for software effort estimation. The model performs knowledge management for applying scrum methodology. The model improves and increases effort estimation of the software as well as management of knowledge. The focus lies upon scrum practices. It uses ontology based model for multi agent estimation system. The projected technique comprises five components, namely – “multi agent effort estimation system, ontology knowledge-base, knowledge creation, knowledge tree maintenance and facility to software companies to map and integrate project data with the system to make quality estimates”.

Aslam W. et al., 2018 [13], identified task allocation as a major challenge for project scheduling in

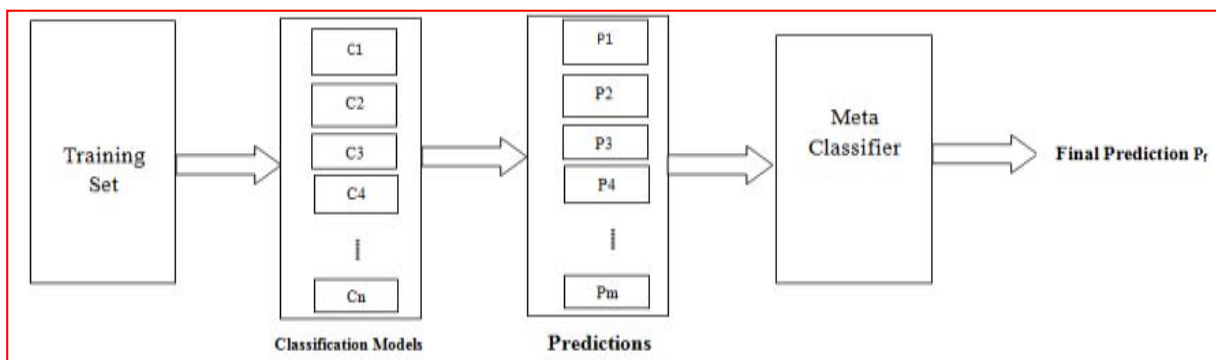
distributed Agile software development. They suggested a task allocation framework for identifying factors which control task allocation as well as provided quantitative method for task allocation for team members who most excellent contest for task requirements articulated as capabilities having different aspects- Technical, Personal and environmental. In the paper authors have suggested task allocation structure having two phases- Identify factors as well as dependencies which persuade task allocation decisions and they suggested a quantitative method for task allocation for task requirements. They emphasized that dispersed teams take 2.5 times longer that collocated teams for task completion. Three types of dependencies were identified namely flow, fit and sharing dependencies. The proposed model assigned role to different team members based on capabilities and practice. Based upon the literature review, we have identified that numerous techniques for task allocation have been applied by the researchers. They have also identified a number of factors responsible for quality improvement of the product and customer satisfaction but still human role in assignment of tasks exists. Though human involvement is better under certain cases, need of an automated process always exists. In the next section, we have shown the factors identified by Filho et. al., 2018 [26] and applied machine learning procedures on those factors in order to automate task allocation.

**MACHINE LEARNING BASED MODEL FOR TASK ALLOCATION**

Filho et. al.,2018 [26], identified different factors to allocation tasks in DASD environment. These factors

involve – “Technical Expertise, Expertise in business, Project Manager Maturity, Proximity to client, Low turnover rate, Availability, Site Maturity, Personal Trust, Time Zone, Cultural similarities, Willingness at site”. The factors are used as alternatives for their research. Following these factors based upon three criteria named as – “Facility to carry out research remotely, Time for the project and cost for the project” [26], they identified the most influential factors based on the ordering ZAPROS – III – i. For decision making ARANAÚ tool [26], was used for alternative definition and result generation. However, this approach can be improved if we apply machine learning approach for task allocation. Based on these thoughts we have proposed a framework for task allocation that will use different machine learning techniques like clustering and classification, etc. Based on these methods we have found that the task allocation has become more efficient and it has produced better results in terms of improved throughput rate and higher customer satisfaction.

In the paper, an approach for enhancing the accuracy of selection of best suited person based on prediction by machine learning methods has been introduced. The model has been trained by many methods and constructed model to guess ratings of a employee to whom the task needs to be allocated. First using k-means and LDA, clusters were created of the employee data that was available in the form of eleven factors provided by Filho *et. al.* [26],to predict the ratings. Afterwards, low, neutral and high ratings were produced to be used in to task allocation.



**Figure 1 - Proposed Framework for Machine learning based task allocation**

Three different approaches have been used, namely “binary classification, multi-class classification and logistic regression”. Binary classification is a simple technique. It is a baseline to start the investigation. It

allows project leaders to compare employees who are best at doing their assigned task and complete it timely. Using multi-class classification or logistic regression can refine the analysis. It tells the team

leaders on how good the employee is to handle any new project (scale from one to five), which is an extreme precious information when team lead wants to compare several employees.

## CONCLUSION

Working with distributed and dispersed teams has been a great challenge over the years. Task allocation in the midst of remote teams is again one of the challenging tasks. Based on different factors the project managers should take a smart decision towards allocation of critical tasks to different team members. In the paper machine learning based framework for task allocation has been proposed. Authors have identified that a Meta – classifier based prediction model can work in line to the task allocation. It works on training the model first with appropriate data sets and factors. As an outcome the model works on predicting best employee for any specific task allocation, selected on the basis of different factors.

## REFERENCES

- [1] M. Fowler and J. Highsmith, "The Agile Manifesto," in *Software Development*, vol. 9, 2001, pp. 28-35.
- [2] M. Shen, G.-H. Tzeng, and D.-R. Liu, "Multi-criteria task assignment in workflow management systems," in *Proc. 36th Annu. Hawaii Int. Conf. Syst. Sci.*, Jan. 2003, pp. 1-9.
- [3] J. Shim, S. Lee, and C. Wu, "A Unified Approach for Software Policy Modeling: Incorporating implementation into a Modeling Methodology," *Springer Lecture Notes in Computer Science*, vol. 2813, pp. 118-130, 2003.
- [4] J. Duggan, J. Byrne, and G. J. Lyons, "A Task Allocation Optimizer for Software Construction," in *IEEE Software*, vol. 21, 2004, pp. 76-82.
- [5] D. K. M. Mak and P. B. Kruchten, "Task coordination in an Agile distributed software development environment," in *Proc. Can. Conf. Elect. Computer Eng.*, May 2006, pp. 606-611.
- [6] A. Lamersdorf, J. Münch, and D. Rombach, "A decision model for supporting task allocation processes in global software development," in *Proc. 10th Int. Conf. Product-Focused Software Process Improvement (PROFES)*, Oulu, Finland, Jun. 2009, pp. 332-346.
- [7] A. B. Marques, J. R. Carvalho, R. Rodrigues, T. Conte, R. Prikladnicki, and S. Marczak, "An ontology for task allocation to teams in distributed software development," in *Proc. IEEE 8th Int. Conf. Global Softw. Eng.*, Aug. 2013, pp. 21-30.
- [8] J. Lin, "Context-aware task allocation for distributed Agile team," in *Proc. 28th IEEE/ACM Int. Conf. Autom. Softw. Eng. (ASE)*, Nov. 2013, pp. 758-761.
- [9] J. Lin, H. Yu, Z. Shen, "An Empirical Analysis of Task Allocation in Scrum-based Agile Programming", *Intl. school of computer science*, 2014.
- [10] Filho M.S., Pinheiro P.R., Albuquerque A.B., "Applying verbal decision analysis in distributed software development: Rank ordering the influencing factors in task allocation", *11th Iberian Conference on Information Systems and Technologies (CISTI)*, 2016.
- [11] Masood Z., Hoda R., Blincoe K., "Exploring workflow mechanisms and task allocation strategies in Agile software teams", *International Conference on Agile Software Development, XP 2017: Agile Processes in Software Engineering and Extreme Programming*, pp 267-273, 2017.
- [12] Adnan M., Afjal M., "Ontology based Multi agent Effort Estimation System for Scrum Agile Method", *IEEE explore*, 2017, pp.-25993-26005.
- [13] Aslam W., Ijaz F., "A Quantitative Framework for Task Allocation in Distributed Agile Software Development", *IEEE explore*, vol. 6, 2018, pp.- 15380-15390.
- [14] George Coulouris, Jean Dollimore, Tim Kindberg, "Distributed Systems: Concepts and Design", 4/E, Pearson Education Ltd, 5th edition.
- [15] A.S. Tanenbaum and M.V. Steen, "Distributed Systems: Principles and Paradigms", Prentice – Hall, 2002.
- [16] Bokhari S. H., "A shortest tree algorithm for optimal assignments across space and time in a distributed processor system", *IEEE Transactions on Software Engineering*, Vol:7(6), 1981, pp. -583–589.
- [17] Lamersdorf A, Münch J, Rombach D, "Towards a Multi- Criteria Development Distribution Model: An Analysis of Existing Task Distribution Approaches", *IEEE International Conference on Global Software Engineering, ICGSE*, 2008, pp. – 109 -118.
- [18] Lamersdorf A, Münch J, Rombach D, "A survey on the state of the practice in distributed software development: criteria for task allocation", *Proceedings of the fourth IEEE international conference on Global Software Engineering, ICGSE*, 2009, pp. - 41–50.
- [19] Lamersdorf A, Münch J., "TAMRI: a tool for supporting task distribution in global software development projects", *International workshop on tool support development and management in distributed software projects, Proceedings of the fourth international conference on global software engineering*, 2009, pp. - 322–327.
- [20] Lamersdorf A, Münch J., "A multi- criteria distribution model for global software development projects", *The Brazilian Computer Society*, 2010, pp. - .
- [21] Larichev O.I., Moshkovich H.M., "Verbal Decision Analysis for Unstructured Problems", Boston: Kluwer Academic Publishers, 1997.
- [22] Bokhari, S. H., "Assignment Problems in Parallel and Distributed Computing", *The Kluwer International Series in Engineering and Computer Science*, 1987, pp. - .
- [23] Simao Filho, M, Pinheiro, P.R., Albuquerque, A.B., "Task Allocation Approaches in Distributed Agile Software Development: A Quasi systematic Review", *4th Computer Science On-line Conference, Proceedings of the 4th Computer Science On-line Conference (CSOC2015)*, Vol. 3: Software Engineering in Intelligent Systems, 2015, pp. - 243-252.
- [24] Simao Filho, M, Pinheiro, P.R., Albuquerque, A.B., "Task Allocation in Distributed Software Development aided by Verbal Decision Analysis", *5th Computer Science On-line Conference 2016 (CSOC2016), Proceedings of the 5th Computer Science On-line Conference 2016*, pp.- 127-137.
- [25] M. Simão Filho, P.R. Pinheiro, A.B. Albuquerque, "Applying verbal decision analysis to task allocation in distributed development of software", *28th Int. Conf. on Software Engineering and Knowledge Engineering (SEKE2016)*, 2016, pp.- 402-407.
- [26] M. Simão Filho, P.R. Pinheiro, A.B. Albuquerque, "Task assignment to distributed teams aided by a hybrid methodology of verbal decision analysis ", *Software IET, Wiley online library*, vol. 11, no. 5, 2017, pp.- 245-255.
- [27] Hole S, Moe N.B., "A case study of coordination in distributed Agile software development", *Communications in Computer and Information Science, Software Process Improvement, Part 5*, 2008, pp. -189–200.
- [28] Madan Singh, N. Chauhan and R. Popli, "A Framework For Transitioning Of Traditional Software Development Method To

Distributed Agile Software Development," 2019 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), GHAZIABAD, India, 2019, pp. 1-4. (IEEE - Scopus Indexed) - <https://ieeexplore.ieee.org/document/8977654>.

[29] Madan Singh, N. Chauhan, R. Popli," A Technique for Transitioning Of Plan Driven Software Development Method to Distributed Agile Software Development", Proceedings of International Conference on Innovative Computing and Communications (ICICC-2020), Delhi University, India, 2020.

[30] Madan Singh, N. Chauhan, R. Popli," Machine Learning based Backlog Prioritization Techniques in Distributed Agile Software Development", International Journal of Advanced Science and Technology, 2020, Vol. 29, No. 6, ISSN: 2005-4238, pp. - 8699-8704.

[31] Madan Singh, N. Chauhan, R. Popli," Implementation of Machine Learning Based Task Allocation in Distributed Agile Software Development", International Journal of Grid and Distributed Computing, 2020, Vol. 13, No. 2, ISSN: 2005 – 4262, pp. - 454-46.

[32] M. Simão Filho, P.R. Pinheiro, A.B. Albuquerque, "Verbal Decision Analysis Applied to the Prioritization of Influencing Factors in Distributed Software Development" Springer International Publishing AG 2018, pp. – 49 – 66.

[33] Kotsiantis, S.B., Zaharakis, I. and Pintelas, P., 2007. Supervised machine learning: A review of classification techniques. Emerging artificial intelligence applications in computer engineering, 160, pp.3-24.