Application of Natural Language Processing Techniques in Agile Software Project Management: A Survey

1st Remah Younisse
Departement of Computer Science
Princess Sumaya University for Technology
Amman, Jordan
r.baniyounisse@psut.edu.jo

2nd Mohammad Azzeh
Departement of Data Science
Princess Sumaya University for Technolog
Amman, Jordan
m.azzeh@psut.edu.jo

2023 14th International Conference on Information and Communication Systems (ICICS) | 979-8-3503-0786-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICICS60529.2023.10330468

***Abstract*—Software effort estimation has long been an important task for better software management. Most of the constructed effort estimation models were based on data collected from software projects that had been developed using traditional software development processes. The structure of this data is usually in the form of tabulated data. Recently, the Agile management framework invaded software production lines as an effective and productive management method. It helps software development teams to complete their tasks in a highly effective way. One of the main components of this success is predicting accurate story points from textual user stories. User stories and story points have become an essential component over which the project planning process is built. Machine learning, artificial intelligence, and deep learning are used to enhance the process of using user story context to put a close estimate of the required resources to finish the project. Using these models has become popular and remarkable in the field of effort estimation. The textual nature of user stories directed the research into the natural language processing path. Natural language processing models can be used to understand textual user story context in order to produce effort estimates. This study reviews the usage of natural language processing methodology in the context of Agile project effort estimation based on the contextual content of user stories.**

***Index Terms*—Effort Estimation, Agile Software Development, Story Points, Natural Language Processing.**

I. Introduction

THE successes of the Agile management framework during the past few decades have made it an excellent method for managing software production projects. At the same time, Agile has witnessed dramatic developments, and advancements [1]. Involving artificial intelligence with the Agile process model brought agile to a new, better level in many cases. The textual format of user stories used to state the project requirements has inspired many works to use natural language processing (NLP) methods to extract beneficial information from these user stories.

Predicting effort and story points from user stories has witnessed advancements after using NLP models. The works focused on using NLP methods for effort estimation can be used to enhance the effort estimation process. In this work, we present a review of these works. In this work, we aim to introduce a review of the recent works presented to apply NLP for software effort estimation in agile software development based on user stories and story points. We analyze the weaknesses and strengths of these methods. We also present a review of the datasets used with these models.

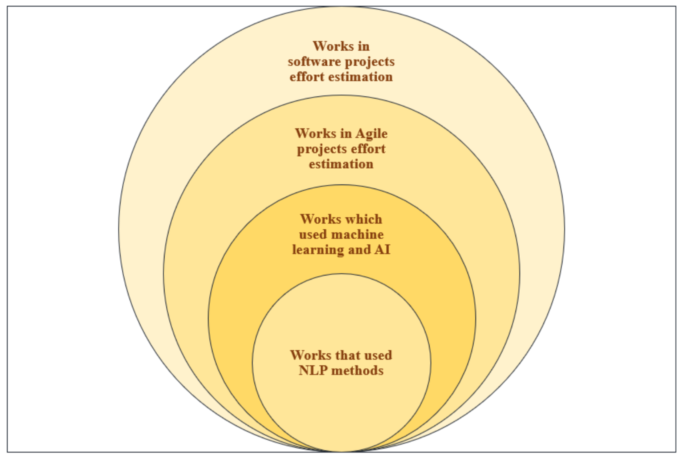

Fig. 1. The areas in software projects effort estimation

The main contribution of this study is to present a review of the usage of NLP in Agile project effort estimation based on user stories. We traverse the new science, the methods presented, the research questions of many influential research papers, and the advancements every research has presented to move the science forward, aiming to produce a gigantic digital effort estimator out of the NLP models. In figure 1, we clarify this study's area of interest. In this study, we focus on the works which used NLP methods to study Agile project effort estimation automated models.

The survey study is driven by the following research questions are:

- **RQ1**. What are the most widely used datasets for esti-

Authorized licensed use limited to: American University of Beirut. Downloaded on August 30,2024 at 07:17:47 UTC from IEEE Xplore. Restrictions apply.

mating effort using NLP models?
- **RQ2**. What are the common NLP models used for estimating effort from user stories and story points?
- **RQ3**. What are the main strengths and weaknesses of the works presented in this field?

II. Background

In this section, we provide the necessary background to develop effort estimation models which use NLP methods.

A. Agile Software Development(ASD)

Agile was introduced in [2] as methods and practices used with software projects. Agile came after other traditional project management methods that have shown many failures. Agile methodologies facilitate suitable management practices which are adapted to the rapidly changing software projects' nature. Generally, Agile projects can be described as built on excellent communication skills among stakeholders including the team leader. The team leader is given a sufficient degree of decision-making. Nevertheless, the decisions are not restricted to him, but to the whole team [3].

Agile Scrum is an agile approach developed to deal with the fact that software project requirements are unpredictable and changeable. Scrum is an organizational pattern that offers a dynamic strategy to manage software projects [4]. According to [5], a Scrum is composed of a sprint, a product backlog, a sprint backlog, and a user story. A sprint is a software product that should be delivered in a certain period of time. The product backlog is basically the requirement set of the software project. The sprint Backlog is a plan for the developers which they will follow during working on the project. And finally, [5] defined user stories as descriptive text for the requirements from users' points of view. User stories tend to change and develop. User stories require an estimation of the required effort, this effort is measured by a numerical value named story point; the more complicated the task the higher the story point value assigned to the user story.

B. User Stories and Story Points

In ASD, project requirements are usually stated in the form of user stories, which are basically a textual description of the project from a user's point of view. User stories are used to describe what the projects should do they are usually written in the form: "As a ¡User¿, I want to¡Have¿so that ¡Benefit¿' [1]. The quality of user stories in ASD is an essential demand to produce high-quality software, user stories quality was discussed in many works such as [6], [7]. The required effort to implement and accomplish a user story is usually measured in story points measures.

Communication between users and developers through user stories can be considered a precise communication method. The style that states the requirement using user stories can be less misleading than the old-school writing the requirements methods. User stories are also easily readable; hence the effort estimation process through user stories can flow smoothly. User stories are expandable, which means that they can be written now and detailed later when the details are needed. This practice is beneficial when non-detailed user stories are used through early communication. [8]

When high-level user stories are expanded, sometimes it is difficult to expand the user story and fit it into a single sprint. When these requirements are divided across many sprints, requirements quality is prone to drop, and traceability challenges are created [9].

Effort estimation is the process of evaluating the required resources to produce the software project described in these user stories. The effort estimation process is usually jeopardized by an inexperienced team, which will result in an overestimation or underestimation situation. The effort estimation process can always reflect the estimator's experience and state [10].

C. Natural language processing (NLP)

Deep learning has proven itself as a helpful tool which improved many fields such as machine vision, speech recognition, and text classification [11]. After 2019, many works were presented to deploy NLP methods in effort estimation missions [12] since NLP is an artificial intelligence sector concerned with interpreting human language.

NLP can be considered a communication method between people and computers. It is widely used in different fields such as management; to demonstrate customers' needs, feedback, and attitudes [13].

Multiple NLP techniques are available and can be used in NLP applications. N-gram for example aims to predict the next word in a sentence based on the previous word mentioned in that sentence. Bag of Words (BoW) is another NLP technique that groups words and measures their relevance based on their term frequency. Meanwhile, word embeddings are neural network-based methods that extract the semantics of words in a specific context. [14]

In deep contextualized models, embeddings from Language Models (ELMo) representation is generated using a function whose input is the entire sentence. This function is prepared using a bidirectional LSTM network trained with a coupled language model. Currently used embedding models can be enhanced by using the ELMo representation. ELMo is effective in extracting sentence information [15]. Using ELMo; a series of pre-trained neural network models for language processing are created such as BERT. BERT and ELMo are now considered very efficient and advanced pre-trained sentence encoders, Which have elevated many NLP applications [16].

III. Related Work

The systematic review presented in [17] considered 80 works presented in the literature to create automated intelligent models to perform effort estimation on software projects, Agile or non-Agile projects. The work revealed limited effort estimation models based on Agile user stories. Most studies use the old-school COCOMO and NASA numerical and statistical non-user story-based datasets. The same study also showed that LSTM is not widely used in effort estimation prediction

Authorized licensed use limited to: American University of Beirut. Downloaded on August 30,2024 at 07:17:47 UTC from IEEE Xplore. Restrictions apply.

models, despite its out-performance compared to many other models, such as feed-forward neural networks. The work in [17] has scanned and organized the works in the context of effort estimation between 2000 and 2020. However, it has not given any special consideration to the Agile projects or the estimations built based on the content of user stories.

The particular consideration to Agile projects effort estimation was given in [18]. The authors investigated the literature and used 11 works to summarize the efforts to create data-driven effort estimation models for Agile projects. The work starts with listing the majorly used datasets in this scope and ends with emphasizing the need to create more. Twelve user stories-based datasets are primarily used in literature. Generally, these datasets are formed of features whose values are textual rather than numerical. The textual nature of the content of user stories has enforced many features in the form of text or strings. The study mentions that 20 ML and AI techniques were used in the context of effort estimation based on story points. Ensemble models were also introduced which is a blend of multiple techniques applied together to solve a certain problem [19] such as effort estimation prediction.

It is convenient to mention that NLP models were often used with user stories. The work presented in [20], for example, used NLP to create a goal model which aims to relate different user stories and create a model reflecting a big picture of the many user stories of the project. The work in [21] was also presented to elevate the quality of the NLP models in extracting a general requirements model and presenting it visually. Meanwhile, [22] used NLP models to transform user stories into a UML diagram.

When NLP methods are used in effort estimation applications and models, user story title and description text is fed into a recurrent neural network capable of analyzing and digesting the text syntax and semantics and transforming this text into a vector representation. The vector is then passed into a deep learning or machine learning model, which can make the effort estimation test. The use of NLP methods in the effort estimation process has shown multiple successes. Therefore, this study presents the approaches used to embed NLP in effort estimation AI models.

IV. Effort Estimation in Agile Development

Effort estimation is an essential step in the project planning process. The time, cost, and manpower required to accomplish a project are measured in the effort estimation process [23]. Misestimation can lead to the project running over budget or being late [24]. Hence many research works have focused on creating efficient effort estimation methodologies. In Agile projects, "user story" is the method to describe what the software project is supposed to do. User stories are the main component to estimate the required effort to accomplish the project.

Poker meetings are a famous method to estimate the required effort for Agile project stories. During Poker meetings, the team members estimate the required effort based on repeated discussions and votes [25]. Discussions and votes are based on the teams' understanding of the project stories and their expertise with similar stories. The more the experts are, the better the estimate, of course. Humans tend to be biased and emotional and might be affected by their managers and peers, while ML and AI are immune to these circumstances [26]. Considering that effort estimation based on story points came into attention after 2010 [27], many doors are to be opened to improve the current fresh, promising methods. Machine learning and artificial intelligence have improved decisions over a large amount of data and complicated calculations in many areas [28]–[30]. Effort estimation also was improved with many ML and AI methods [31]. In these works, the models were trained with datasets such as COCOMO and NASA datasets. Relying on these datasets is considered efficient, but it limits decisions to be built over limited features contained in the Agile user stories. There is much more helpful information carried, yet, neglected because they are not listed in the features of the dataset [26].

V. Review of NLP Methods for Story Point Estimation

Effort estimation in an agile project is built over the content of the project's user stories. Experts and project members sometimes do this estimation, which is done by data-driven models using datasets derived from user stories and other related information to predict the required effort for other user stories within a new project being effort estimated.

The quality of the decisions taken by these models is controlled by the nature of the data being fed into them. The data determines the model to be used as well. For example, the work presented in [32] has used Bayesian networks while fuzzy-based models were used in [33] and decision trees were used in [34]. This continuous research can be described as the process of evolution happening in many areas where ML and AI are considered comfort zones.

At the same time, NLP methods were used in many works to enhance the research area in Agile projects' effort estimation. The work presented in [35] used deep neural networks to predict and estimate the required effort, not for the entire project, but for every story independently. The work focused on creating an estimation dataset and model for every user story. The work in [36] replicated and extended the dataset creation to develop a more general and practical dataset.

The model learns from previously accomplished user stories to predict the effort required for other story points within a project. The proposed model consists of four stages; word embedding, document representation using LSTM, deep representation using Recurrent Higway Net (RHWN), and differentiable regression. The evaluation process took place by answering six valuable questions. The proposed technique in [35] has shown excellent results regarding all six measuring questions. The results were evaluated based on the mean absolute error, Median Absolute Error, and standardized accuracy measurements.

The work in [35] has established a rigid path other researchers can follow to create an NLP effort estimation model.

Authorized licensed use limited to: American University of Beirut. Downloaded on August 30,2024 at 07:17:47 UTC from IEEE Xplore. Restrictions apply.

The work has thoroughly investigated many options. For example, the LSTM model was compared with the previously used other models, such as Doc2V and BagOfWords methods; LSTM was found to be much more beneficial. Frequency-inverse document frequency (TF-IDF), distributed representations of documents (doc2vec) methods were presented in [37] in the context of Agile user story effort estimation.

In the year 2022 and four years after [35] gave a robust estimation NLP and deep neural networks can take prediction estimates to a new level. The work in [38] was re-evaluated with more realistic cases and measurements. Generally, the study concluded that the study's findings in 2018 were very optimistic and biased towards making deep-SE of [35] seem a more powerful estimation tool than it is. The study of [38] has reopened the door to producing an effort estimation tool that can be relied on for Agile story points datasets.

In [39], using NLP and DNN tools for effort estimation was found more efficient by using further information during the prediction process, such as the seniority of employees. The study has confirmed the importance of using NLP as well as the availability for many more improvements in the field in the Agile projects effort estimation process. The work used a small dataset which showed an improvement in the results compared to similar approaches, but the results could have been better.

Hierarchical attention networks (HAN) were used through the effort estimation process in [11]. Story points were passed through different gated recurrent unit neural networks GRU layers to extract the ultimate possible semantic information from the stories datasets. The summarized information was passed into a hierarchical attention network followed by a neural network that aims to classify the stories into three classes. The hierarchical attention networks are used to deal with the input as a sequence of words and sentences once. The decision taken by the neural network was considered as participation in Poker meetings aim to estimate effort for stories of the project. This method has improved the NLP model by using the GRU networks. It also has brought the NLP effort estimation concept into a more realistic position as they dealt with the model as a participant in poker meetings rather than eliminating the experts' judgment; the model was considered a new reliable judge.

The research questions of the study are: "What is the effect of adding an attention layer during the training phase? Can attention mechanism be leveraged to improve software effort estimation? Does the use of Hierarchical Attention Networks provide more accurate story point estimates than using traditional classification technique?" [11]. Adding the attention layer has reduced training time. For the second question, the answer confirmed accuracy improvement.

The work presented in [27] made the estimation process more comfortable. The contextual content of the story points was analyzed using the Latent Dirichlet Allocation (LDA) model to transfer the content into a set of probabilities according to the topics included in user stories. After transforming stories into vectors of numerical values, clustering algorithms were applied to these vectors. The effort estimation was derived according to which cluster every story point relates.

The study has proved that the simple clustering method can be as efficient as more complex artificial effort estimation models. [27]. The result section showed that the three questions' answers were satisfying, and the clustering method applied to the resulted vectors of the NLP mapping process can be considered efficient for effort estimation based on story points.

Authors of [40] have noticed that story point datasets mix long sentences, short pieces of code, and special characters. The nature of the content of story point datasets has inspired them to create a particular heterogeneous graph in which the content of the user stories can be held. The graph is then fed into a heterogeneous graph neural network model to learn and predict the effort required to accomplish new stories within a project.

At the time, many approaches were proposed to utilize NLP in the effort estimation process. The study of [41] can be considered a comparative study that came just in time. The study aims to compare the Random forest and BERT feed-forward linear neural network classifiers in the context of effort estimation NLP-based applications. Feature extraction methods used during the comparison process are Term Frequency–Inverse Document Frequency (TF-IDF) and Bidirectional Encoder Representations from Transformers(BERT). Using BERT as a classifier as well as a feature extractor showed a slight advancement over the other combinations of classifiers and feature extractors. One of the essential findings of the presented study is that machine learning model estimation can be as efficient as an expert estimation. Unlike the other recent related studies to [41], the research questions did not target assessing the performance of their suggested model, but answered general four questions regarding the accuracy of the ML models in predicting effort estimation. The research questions also mentioned the difference between different vectorization. methods such as BERT and TF-IDF. The study also compares experts' predictions to ML methods' prediction results.

The first question answer is that the BERT has outperformed the other ML models, yet the other models gave results very close to BERTs. While for the second question, no significant difference was detected between the two feature extraction models. The third question answer came to show that ML can be as good as experts' estimation in many cases. [42] has compared many ML models in effort estimation and found that among Term Frequency - inverse document frequency (TF-idf), fastText, RNN, LSTM, and BERT, fastText with the pre-trained model has scored the best results then LSTM model did.

Another comparative study in the context of effort estimation is [43]. This study was built to show the effectiveness of pre-trained embedding models in NLP effort estimation models and how much pre-training the model can be helpful to get better results. Pre-trained models were found to be efficient in estimating effort based on requirement text only.

Authorized licensed use limited to: American University of Beirut. Downloaded on August 30,2024 at 07:17:47 UTC from IEEE Xplore. Restrictions apply.

TABLE I
LITERATURE SUMMARY

Ref#	Dataset	Metric	Contribution	Limitations
[35]	They generated their own dataset from 16 different SW projects	Absolute Error (MAE), Median Absolute Error (MdAE), and Standardized Accuracy (SA)	Generating a new dataset for effort estimation. Using LSTM methods for predicting the required effort	The criticism in [38]
[40]	The dataset used in [35]	Absolute Error (MAE), Median Absolute Error (MdAE), and the Standardized Accuracy (SA)	They used a heterogeneous graph in which the content of the user stories can be held	Model's complexity
[39]	A dataset of rated tasks	Accuracy and MMRE	They adapted an estimation model which takes the textual description of the task and available metadata such as employee seniority	MMRE and Accuracy results due to using small datasets
[11]	The dataset used in [35]	Accuracy, Recall, Precision, and F score	They adapted an estimation model which uses two levels of attention mechanisms, and the use of GRU neural networks	MMRE and Accuracy results due to using small datasets
[27]	The largest dataset available collected from 26 projects [38]	Absolute Error (MAE), Median Absolute Error (MdAE), and the Standardized Accuracy (SA)	They adapted an estimation model which analyzes textual features of software issues by using latent Dirichlet allocation (LDA) and clustering methods	The results were similar or slightly better than existing methods, yet their method is considered simpler

VI. RESEARCH QUESTIONS DISCUSSION

In this section, we present the answers to the research question listed in this paper's contribution section.

A. **RQ1**. *What are the most widely used datasets for estimating effort using NLP models?*

The answer to this question is that one of the main datasets is the one used in [11], [35], [40]. This dataset was created by the authors of [35] and collected from 16 different projects.

This dataset was criticized in [38], and another dataset was proposed in this work and used in the work of [27].

B. **RQ2**. *What are the common NLP models used for estimating effort from user stories and story points?*

Most works, such as [35], [40], used LSTM neural networks. While [11] has used networks built with GRU neural networks. We can notice that none of the previous studies used the notion of NLP transformers in their work.

C. *Q3: What are the main strengths and weaknesses of the works presented in this field?*

The main weaknesses found in this field are the lack of datasets that can be used to create and train NLP networks. This weakness leads to the subsequent weakness related to the first one: the lack of proposed works in this area of research despite the elevation these models can present to the quality of the automated effort estimation models for Agile projects.

The main strength of this field is that it is open to further evolutionary steps forward to build robust and considerable models. The presented methods are developing considerably in the quality and the approaches used for estimating effort based on user stories. The present works are also constructing a tough road that can be followed in future works, especially in creating reliable datasets and using them conveniently.

VII. CONCLUSION

The use of NLP methods in Agile projects effort estimation can be described as a promising technique that relies on user stories and textual content to extract the features which are used later to make the estimation. The gradual advancements in the methods used can be noticed from around 2108 until now. However, the need to create more comprehensive datasets to be used with the NLP models is urgent. Works that are presented need more revisions, assessment on the field, and enhancement. The use of NLP methods in Agile projects effort estimation is a newly opened research area are ready to accept more enhancements.

REFERENCES

[1] S. Al-Saqqa, S. Sawalha, and H. AbdelNabi, "Agile software development: Methodologies and trends." *International Journal of Interactive Mobile Technologies*, vol. 14, no. 11, 2020.

[2] T. Dybå and T. Dingsøyr, "Empirical studies of agile software development: A systematic review," *Information and software technology*, vol. 50, no. 9-10, pp. 833–859, 2008.

[3] S. Nerur, R. Mahapatra, and G. Mangalaraj, "Challenges of migrating to agile methodologies," *Communications of the ACM*, vol. 48, no. 5, pp. 72–78, 2005.

[4] A. T. Karabulut and E. Ergun, "A new way of management: A scrum management," *International Journal of Commerce and Finance*, vol. 4, no. 2, pp. 108–117, 2018.

[5] V. Lenarduzzi, I. Lunesu, M. Matta, and D. Taibi, "Functional size measures and effort estimation in agile development: a replicated study," in *International Conference on Agile Software Development*. Springer, 2015, pp. 105–116.

[6] G. Lucassen, F. Dalpiaz, J. M. E. van der Werf, and S. Brinkkemper, "Improving agile requirements: the quality user story framework and tool," *Requirements engineering*, vol. 21, pp. 383–403, 2016.

Authorized licensed use limited to: American University of Beirut. Downloaded on August 30,2024 at 07:17:47 UTC from IEEE Xplore. Restrictions apply.

[7] G. Lucassen, F. Dalpiaz, J. M. E. Van Der Werf, and S. Brinkkemper, "Forging high-quality user stories: towards a discipline for agile requirements," in *2015 IEEE 23rd international requirements engineering conference (RE)*. IEEE, 2015, pp. 126–135.

[8] M. Cohn, "Advantages of user stories for requirements," *InformIT Network*, 2004.

[9] R. Kasauli, E. Knauss, J. Horkoff, G. Liebel, and F. G. de Oliveira Neto, "Requirements engineering challenges and practices in large-scale agile system development," *Journal of Systems and Software*, vol. 172, p. 110851, 2021.

[10] B. Alsaadi and K. Saeedi, "Data-driven effort estimation techniques of agile user stories: a systematic literature review," *Artificial Intelligence Review*, vol. 55, no. 7, pp. 5485–5516, 2022.

[11] H. KASSEM, K. MAHAR, and A. SAAD, "Software effort estimation using hierarchical attention neural network," *Journal of Theoretical and Applied Information Technology*, vol. 100, no. 18, 2022.

[12] N. Surantha, "Model and dataset trend in software project effort estimation–a systematic literature review."

[13] Y. Kang, Z. Cai, C.-W. Tan, Q. Huang, and H. Liu, "Natural language processing (nlp) in management research: A literature review," *Journal of Management Analytics*, vol. 7, no. 2, pp. 139–172, 2020.

[14] R. A. de Morais, "Deep learning based models for software effort estimation using story points in agile environments," 2021.

[15] W. Yang, L. Li, Z. Zhang, X. Ren, X. Sun, and B. He, "Be careful about poisoned word embeddings: Exploring the vulnerability of the embedding layers in nlp models," *arXiv preprint arXiv:2103.15543*, 2021.

[16] I. Tenney, D. Das, and E. Pavlick, "Bert rediscovers the classical nlp pipeline," *arXiv preprint arXiv:1905.05950*, 2019.

[17] P. S. Kumar, H. S. Behera, A. Kumari, J. Nayak, and B. Naik, "Advancement from neural networks to deep learning in software effort estimation: Perspective of two decades," *Computer Science Review*, vol. 38, p. 100288, 2020.

[18] B. Alsaadi and K. Saeedi, "Data-driven effort estimation techniques of agile user stories: a systematic literature review," *Artificial Intelligence Review*, pp. 1–32, 2022.

[19] A. Mohammed and R. Kora, "An effective ensemble deep learning framework for text classification," *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 10, pp. 8825–8837, 2022.

[20] T. Güneş and F. B. Aydemir, "Automated goal model extraction from user stories using nlp," in *2020 IEEE 28th International Requirements Engineering Conference (RE)*. IEEE, 2020, pp. 382–387.

[21] M. Robeer, G. Lucassen, J. M. E. Van Der Werf, F. Dalpiaz, and S. Brinkkemper, "Automated extraction of conceptual models from user stories via nlp," in *2016 IEEE 24th international requirements engineering conference (RE)*. IEEE, 2016, pp. 196–205.

[22] M. Elallaoui, K. Nafil, and R. Touahni, "Automatic transformation of user stories into uml use case diagrams using nlp techniques," *Procedia computer science*, vol. 130, pp. 42–49, 2018.

[23] P. Srivastava, N. Srivastava, R. Agarwal, and P. Singh, "Estimation in agile software development using artificial intelligence," in *Proceedings of Trends in Electronics and Health Informatics*. Springer, 2022, pp. 83–93.

[24] E. Kula, E. Greuter, A. Van Deursen, and G. Gousios, "Factors affecting on-time delivery in large-scale agile software development," *IEEE Transactions on Software Engineering*, vol. 48, no. 9, pp. 3573–3592, 2021.

[25] M. Usman, E. Mendes, F. Weidt, and R. Britto, "Effort estimation in agile software development: a systematic literature review," in *Proceedings of the 10th international conference on predictive models in software engineering*, 2014, pp. 82–91.

[26] P. Chongpakdee and W. Vatanawood, "Estimating user story points using document fingerprints," in *2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, 2017, pp. 149–152.

[27] V. Tawosi, A. Al-Subaihin, and F. Sarro, "Investigating the effectiveness of clustering for story point estimation." IEEE, 2022.

[28] M. Arora, A. Sharma, S. Katoch, M. Malviya, and S. Chopra, "A state of the art regressor model's comparison for effort estimation of agile software," in *2021 2nd International Conference on Intelligent Engineering and Management (ICIEM)*. IEEE, 2021, pp. 211–215.

[29] E. Wińska, E. Kot, and W. Dabrowski, "Reducing the uncertainty of agile software development using a random forest classification algorithm," in *International Conference on Lean and Agile Software Development*. Springer, 2021, pp. 145–155.

[30] G. Fischer, "End-user development: Empowering stakeholders with artificial intelligence, meta-design, and cultures of participation," in *International Symposium on End User Development*. Springer, 2021, pp. 3–16.

[31] Y. Mahmood, N. Kama, A. Azmi, A. S. Khan, and M. Ali, "Software effort estimation accuracy prediction of machine learning techniques: A systematic performance evaluation," *Software: Practice and Experience*, vol. 52, no. 1, pp. 39–65, 2022.

[32] O. Malgonde and K. Chari, "An ensemble-based model for predicting agile software development effort," *Empirical Software Engineering*, vol. 24, no. 2, pp. 1017–1055, 2019.

[33] J. M. Alostad, L. R. A. Abdullah, and L. S. Aali, "A fuzzy based model for effort estimation in scrum projects," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 9, 2017. [Online]. Available: http://dx.doi.org/10.14569/IJACSA.2017.080939

[34] I. Hussain, L. Kosseim, and O. Ormandjieva, "Towards approximating cosmic functional size from user requirements in agile development processes using text mining," in *International Conference on Application of Natural Language to Information Systems*. Springer, 2010, pp. 80–91.

[35] M. Choetkiertikul, H. K. Dam, T. Tran, T. Pham, A. Ghose, and T. Menzies, "A deep learning model for estimating story points," *IEEE Transactions on Software Engineering*, vol. 45, no. 7, pp. 637–656, 2018.

[36] V. Tawosi, A. Al-Subaihin, R. Moussa, and F. Sarro, "A versatile dataset of agile open source software projects," *arXiv preprint arXiv:2202.00979*, 2022.

[37] V.-S. Ionescu, H. Demian, and I.-G. Czibula, "Natural language processing and machine learning methods for software development effort estimation," *Studies in Informatics and Control*, vol. 26, no. 2, pp. 219–228, 2017.

[38] V. Tawosi, R. Moussa, and F. Sarro, "Deep learning for agile effort estimation have we solved the problem yet?" *arXiv preprint arXiv:2201.05401*, 2022.

[39] I. Dan, R. Cătălin, and O. Oliver, "An nlp approach to estimating effort in a work environment," in *2020 International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*. IEEE, 2020, pp. 1–6.

[40] H. Phan and A. Jannesari, "Heterogeneous graph neural networks for software effort estimation," *arXiv preprint arXiv:2206.11023*, 2022.

[41] M. Alhamed and T. Storer, "Evaluation of context-aware language models and experts for effort estimation of software maintenance issues," in *2022 IEEE International Conference on Software Maintenance and Evolution (ICSME)*. IEEE, 2022, pp. 129–138.

[42] S. A. Raza, "Predicting the duration of user stories, machine learning for agile planning," 2020.

[43] E. M. D. B. Fávero, D. Casanova, and A. R. Pimentel, "Se3m: A model for software effort estimation using pre-trained embedding models," *Information and Software Technology*, vol. 147, p. 106886, 2022.

Authorized licensed use limited to: American University of Beirut. Downloaded on August 30,2024 at 07:17:47 UTC from IEEE Xplore. Restrictions apply.