UDC 004.85
IRSTI 06.81.29

https://doi.org/10.55452/1998-6688-2026-23-1-107-116

[1]*Aitim A.K.,
MSc., ORCID ID: 0000-0003-2982-214X,
*e-mail: a.aitim@iitu.edu.kz
[1]Sembina G.K.,
Cand. Tech. Sc., ORCID ID: 0000-0003-2920-1490
e-mail: g.sembina@iitu.edu.kz

[1]International Information Technology University, Almaty, Kazakhstan

ENSEMBLE MACHINE LEARNING APPROACHES FOR IT PROJECT COST ESTIMATION UNDER DATA SCARCITY CONDITIONS

Abstract

Accurate prediction of IT project costs is crucial for successful project planning, budgeting, and resource allocation. However, typical cost estimation methods, such as Function Point Analysis, or expert-based evaluations, frequently fail to produce trustworthy conclusions, especially in developing countries like Kazakhstan where previous project data is few or incomplete. This study looks into how ensemble machine learning algorithms, notably Random Forest and Gradient Boosting, can be used to predict IT project costs when there is insufficient data available. To solve data shortage, this study applies synthetic data creation techniques, which result in extended datasets that model various project scenarios while retaining statistical features observed in real-world cases. The presented models use essential project variables, such as team size, project complexity, development process, and project size, as inputs for cost prediction. Experimental results show that ensemble approaches outperform standard estimating techniques in terms of predictive accuracy. Random Forest achieved the lowest mean absolute error (MAE = 0.09) and highest coefficient of determination (R^2 = 0.603). Furthermore, feature importance analysis shows that project size and development time are the most important elements in cost estimation. The findings demonstrate ensemble learning's usefulness in dealing with complicated, nonlinear connections among project variables, as well as providing a feasible approach for improving cost estimation techniques in the absence of high-quality historical data. This work adds to the development of intelligent decision support systems and offers practical insights for IT project managers and policymakers in emerging economies who want to improve IT project budgeting and planning.

Key words: ensemble learning, random forest, gradient boosting, machine learning, data scarcity, decision support systems.

Received: July 2, 2025; revised: January 21, 2026; accepted: February 17, 2026.

Introduction

In the world of information technology (IT), being able to accurately estimate project expenses is still a key part of good management. Cost estimates are used to manage budgets, allocate resources, analyze risks, and negotiate contracts. Even though this work is very important, classic cost estimation methods like expert judgment, parametric models like COCOMO, and Function Point Analysis (FPA) typically have trouble making accurate projections in today's IT systems. These old-fashioned methods usually use deterministic formulas and need a lot of historical data to set model parameters and improve estimates. Unfortunately, it is hard to get good historical data on IT projects in many developing areas, including Kazakhstan. This makes it hard to accurately predict costs [1]. The fact that current IT projects are naturally complicated and varied makes this problem even worse. There are several factors that make the correlations between project features and their final prices very nonlinear and dynamic. These include the size and complexity of the project, the makeup of the team,

the development methodologies used (such as Agile, Waterfall, or Hybrid), and the technological stacks used. Because of this, typical estimation models that presume linear or simple correlations often miss the real cost drivers, which can cause projects to go over budget and be delayed. Machine learning (ML) techniques have come up as promising new ways to get around the problems with traditional methods. Ensemble learning approaches like Random Forest and Gradient Boosting have showed a lot of promise in working with complicated, high-dimensional datasets and finding latent nonlinear correlations between variables. These methods can simulate complex relationships between project elements in a way that changes over time, which makes them perfect for the cost estimation problem. But there is still a big problem: ML algorithms usually need a lot of high-quality training data to work well, which is often not available in new IT industries. The goal of this study is to find out how well ensemble machine learning methods, like Random Forest and Gradient Boosting, can be used to estimate the cost of IT projects when there isn't much data available. It also looks into how synthetic data augmentation might help models work better when there isn't enough real data or when it is broken apart. The results shown here give useful information about how to develop data-driven decision support systems for managing IT projects in emerging markets, where being able to accurately estimate costs is important for digital transformation and economic growth.

The novelty of the study is therefore contextual and methodological, focusing on (i) the structured integration of synthetic data generation with ensemble learning, and (ii) its application to real-world public-sector IT projects characterized by fragmented and incomplete historical records.

Recent research has shown that ensemble learning methods, especially Random Forest and Gradient Boosting, are strong options for modeling the complicated connections that are common in IT project data. Li et al. (2021) showed that Random Forest models are always better at predicting the costs of software development than linear regression [2]. They found that the mean absolute error (MAE) was significantly lower across datasets with different project attributes, such as team size, complexity, and development methodology. Wang et al. (2022) confirmed again that Gradient Boosting methods, especially more complex ones like XGBoost and LightGBM, are better at making predictions, especially when working with datasets that have a lot of dimensions and complicated feature interactions [3]. People are starting to realize that these ensemble methods are quite strong, can help prevent overfitting, and can find small, nonlinear patterns that regular models generally miss. As ensemble learning has gotten better, using synthetic data has become an important way to deal with the ongoing problem of not having enough data in software engineering. Kumar and Singh (2020) looked at the benefits of adding synthetic records to tiny datasets [4]. They found that machine learning models trained on hybrid datasets were more accurate and able to generalize better. In the same way, Patki et al. (2020) showed that synthetic data may accurately mimic the statistical distributions of real project data while keeping the data private [5]. This makes it especially useful in fields where privacy or proprietary considerations make it hard to share data. These results show that utilizing synthetic data to fill in gaps in historical records is possible in real life. This is a typical problem in developing markets like Kazakhstan [6]. Another big change in the last five years has been the use of explainable AI (XAI) techniques with ensemble learning to make predictive models more open and trustworthy. Chen et al. (2023) used SHapley Additive exPlanations (SHAP) with Gradient Boosting models to estimate IT costs [7]. They showed how certain project variables affect cost projections. This level of interpretability is important for practitioners because it lets project managers and stakeholders understand and check model outputs instead of depending on "black-box" forecasts [8]. Even with these technological developments, there is still a void in the literature when it comes to the specific use of ensemble ML techniques and synthetic data creation for estimating the costs of IT projects in contexts with limited data [9]. While studies from throughout the world show that these strategies work, not many have looked at how to use them in places like Kazakhstan and other emerging economies where there isn't a lot of data, there are a lot of different types of projects, and the market works in its own way [10]. This study tries to fill in the gaps by looking at how Random Forest and Gradient Boosting models trained on synthetically enriched datasets can make cost projections for IT projects more accurate and reliable when there isn't enough data.

Materials and methods

This study's main goal is to find a way to train ML models to anticipate IT project costs in Kazakhstan and other emerging countries. One of the biggest problems is that there aren't enough high-quality, complete datasets that can be used for this. The thesis research shows that the fragmented nature of accessible data makes it hard to apply typical cost estimation models. This means that other methods are needed to create accurate predictive models [11]. The first step was to gather real data from a number of sources: Websites like the Kazakhstan public procurement site gave information about IT projects, such as the names of the projects, short descriptions, budget amounts, and contractors [12]. But a lot of the records didn't have the detailed technical information that is important for ML, including lines of code, team structure, or development approach. The "Smart Almaty" digital transformation program was a key source of information about 49 ICT projects that were meant to improve the city's infrastructure [13]. Projects were in different areas, such as transportation, health care, and keeping an eye on the environment. Even while they were useful, many entries didn't have consistent data fields, which made them less useful for ML. Vinchi Interactive and other software companies published public reports and technical case studies that went into great depth about the different stages of a project, the time frames, the sizes of the teams, and sometimes even the costs. These stories gave us important information on how to set realistic ranges and dependencies in the development of synthetic data [14].

Real project data were collected from multiple sources, including public procurement platforms, the Smart Almaty digital transformation program, and industrial case reports. Synthetic data were subsequently generated to preserve observed statistical distributions and inter-variable dependencies, following established practices in the literature. This approach enables controlled experimentation while mitigating overfitting risks in small-sample settings.

Table 1 shows the main factors employed in this investigation, both in genuine and fake datasets.

Table 1 – Key variables used in the study

Variable	Type	Description	Range / Categories
Project Size (LOC)	Numerical	Size of project measured in Lines of Code	5,000–120,000
Team Size	Numerical	Number of people involved in the project	2–15
Development Methodology	Categorical	Methodology used in project development	Waterfall, Agile, Hybrid
Complexity	Categorical	Complexity level of the project	Simple, Moderate, Complex
Complexity Factor	Numerical	Numeric factor associated with project complexity	40 (Complex) – 80 (Simple)
Methodology Factor	Numerical	Coefficient adjusting estimated development time	1.0–1.1
Development Time	Numerical	Estimated duration in person-days	Computed from size & factors
Estimated Cost	Numerical	Predicted project cost	$1,000–$80,000 (synthetic range)
Actual Cost	Numerical	Realized cost of the project (if available)	$1,000–$80,000 (observed range)

About 400 fake records were made, which greatly increased the size of the dataset while yet keeping it statistically realistic [15]. Before modeling, both actual and synthetic data went through a lot of preprocessing to make sure they were consistent and could be used with ML algorithms: The mean values of the available data were used to fill in the gaps where numbers were missing [16].

The mode was employed for categorical variables. One-hot encoding turned category variables like project complexity and development approach into binary features. This stopped the model from getting ordinal relationships wrong when they didn't exist [17]. We used min-max normalization to change all the numbers, like project size, team size, anticipated cost, and actual cost, to a range of 0 to 1 [18]. This step helped keep learning algorithms stable and kept variables with large absolute values from taking over.

To split the merged dataset into two parts: one for training (80%) and one for testing (20%) [19]. We used grid search and five-fold cross-validation to find the best model configurations by tweaking the hyperparameters. The parameters that were changed were the number of trees (n_estimators), the maximum depth of the trees (max_depth), the minimum number of samples needed for node splits and leaves (min_samples_split, min_samples_leaf), and the learning rate (for GB) [20]. The Table 2 shows the ranges of hyperparameters that were tested while adjusting the models for both the Random Forest and Gradient Boosting methods. This makes sure that the findings are fully optimized and can be repeated.

Table 2 – Hyperparameter ranges for model tuning

Parameter	Random Forest	Gradient Boosting
n_estimators	50–500	50–500
max_depth	3–15	3–15
min_samples_split	2–10	2–10
min_samples_leaf	1–5	1–5
learning_rate	–	0.01–0.3
subsample	–	0.5–1.0
max_features	sqrt, log2, None	sqrt, log2, None

The used typical regression metrics to check how well the model worked: Mean absolute error (MAE) tells you how accurate your predictions are by showing you the average absolute difference between the projected and actual expenses. Root mean squared error (RMSE) gives greater weight to larger errors, useful for identifying significant prediction discrepancies. The coefficient of determination (R^2) shows how much of the difference in actual costs the model can explain [21].

Feature importance was also analyzed to identify the most significant drivers influencing project cost predictions. The thesis results showed that project size and expected development time were the best predictors of costs. Complexity and methodology were also important, but they had less of an effect on costs. This detailed strategy made it possible to create and test ML-based cost estimation models that work well even when there isn't a lot of data, which is common in the Kazakhstani IT sector.

Results

Data pretreatment includes getting rid of missing data, turning categorical features (like development approach) into numbers using One-Hot Encoding, and scaling procedures to make numeric features (like team size and cost) more even. To get the data ready for machine learning, the following steps were taken: Use imputation or drop incomplete records to deal with missing values. To make sure machine learning algorithms can use your data, normalize numeric features to a predefined range, like 0 to 1. To can use one-hot encoding or label encoding to turn categorical variables (such development technique or project difficulty) into numbers. An example of data preprocessing for the dataset used to estimate the cost of an IT project is to use the holdout testing set, which was 20% of the combined actual and synthetic dataset, to see how well the ensemble machine learning models worked. MAE, RMSE, and R^2 were some of the most important evaluation measures. In the Table 3 shows the data after dealing with missing values.

Table 3 – Using one code for the Methodology and Complexity columns.

Project Size (LOC)	Complexity	Team Size	Methodology	Estimated Cost (USD)	Actual Cost (USD)
15,000	Moderate	5	Agile	120,000	125,000
25,000	Complex	10	Waterfall	250,000	265,000
19,333	Simple	3	Agile	80,000	78,000
18,000	Moderate	6	Waterfall	150,000	95,000

Table 4 shows the data after the variables were coded.

Table 4 – Data after the variables

Project Size (LOC)	Team Size	Estimated Cost (USD)	Actual Cost (USD)	Complexity_ Moderate	Complexity_ Complex	Complexity_ Simple	Methodology_ Agile	Methodology_ Waterfall
15,000	5	120,000	125,000	1	0	0	1	0
25,000	10	250,000	265,000	0	1	0	0	1
19,333	3	80,000	78,000	0	0	1	1	0
18,000	6	150,000	95,000	1	0	0	0	1

Use minimum and maximum scaling to give quantitative attributes (such project size, team size, expected cost, and actual cost) a range from 0 to 1 (1):

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

The last dataset that has been preprocessed. The dataset is now ready to train machine learning models because it has been cleaned, encoded, and normalized in Table 5.

Table 5 – Normalized data

Project Size	Team Size	Estimated Cost	Actual Cost	Complexity_ Moderate	Complexity_ Complex	Complexity_ Simple	Methodology_ Agile	Methodology_ Waterfall
0.00	0.40	0.27	0.65	1	0	0	1	0
1.00	1.00	1.00	1.00	0	1	0	0	1
0.65	0.00	0.00	0.54	0	0	1	1	0
0.35	0.60	0.47	0.00	1	0	0	0	1

The Random Forest model was the most accurate at making predictions out of all the ones that were evaluated. It had a mean absolute error of 0.09 and a root mean squared error of 0.13. The model's R^2 score of 0.603 means that it could explain about 60.3% of the differences in the actual costs of the project. These results show that Random Forest is good at dealing with nonlinear relationships and interactions between project features while avoiding overfitting through ensemble averaging.

Gradient Boosting was very good in making predictions, however it made a few more mistakes than Random Forest. It had a mean absolute error of 0.10 and a root mean squared error of 0.14. Its R^2 value was 0.557. In the Figure 1 shows a feature correlation matrix that was made using the Random Forest model. The values show the Pearson correlation coefficient between two features. Blue tones show positive associations, whereas red shades show negative correlations. When two

category variables can't be true at the same time, such complexity levels or techniques, they tend to be strongly correlated.

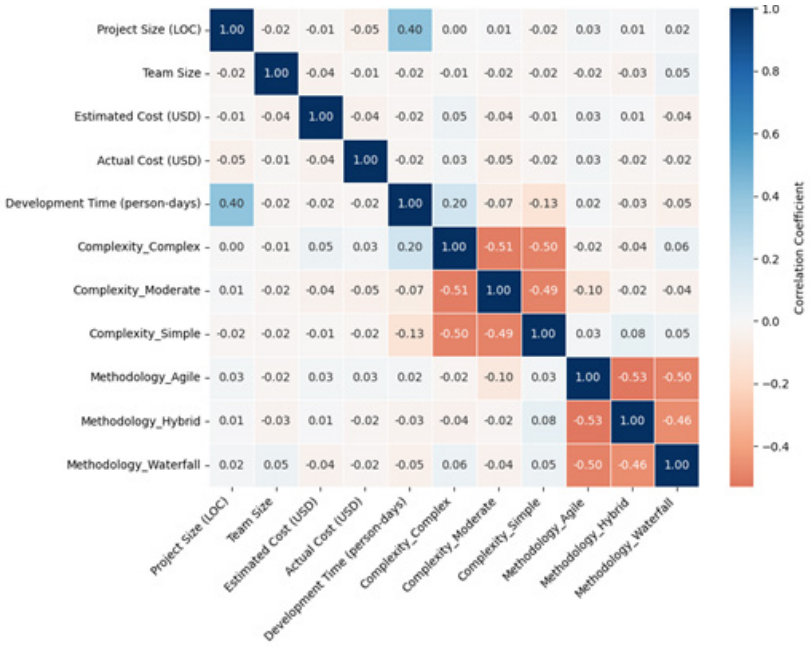

Figure 1 – Feature correlation matrix (Random Forest)

Project size (LOC) has a weak positive association (0.40) with Development Time (person-days), which means that bigger projects usually take longer to develop. There are modest correlations between project size (LOC) and both predicted cost (USD) and actual cost (USD). Most of the time, the correlations between variables are modest (between -0.10 and 0.10), which means that many attributes don't depend on each other in a direct, linear way. The Gradient Boosting model created the feature correlation matrix that is shown in Figure 2.

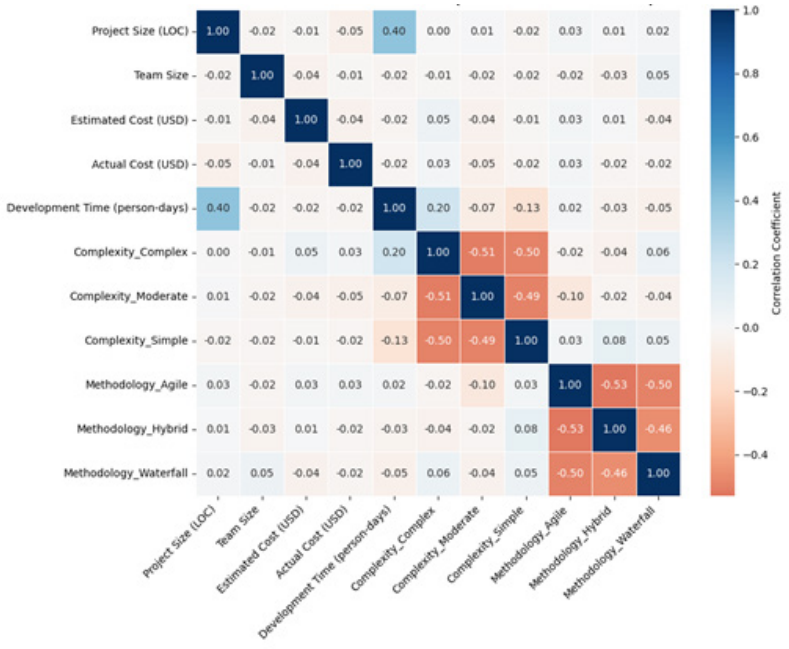

Figure 2 – Feature correlation matrix (Gradient Boosting)

Project size (LOC) has a modest positive association (0.40) with Development Time (person-days), which means that bigger projects usually take more work and time to finish. There aren't high correlations between project size (LOC) and either estimated cost (USD) or actual cost (USD). This means that size alone isn't a good direct predictor of expenses without taking into account other things like complexity and methodology. The complexity variables (Complexity_complex, Complexity_moderate, Complexity_simple) have substantial negative correlations with each other (about –0.50), which makes sense because these variables are binary indicators for categories that can't overlap. There are also negative correlations between the methodology variables (Methodology_agile, Methodology_hybrid, Methodology_waterfall), which means that these characteristics can't be used together. Most of the other correlations are rather low (between about –0.10 and 0.10), which means that a lot of the attributes aren't very strongly related to each other in a purely linear way. The fact that many features have poor correlations shows how complicated the relationships are when estimating the cost of an IT project. This is why ensemble learning approaches like Gradient Boosting are useful; they can find nonlinear interactions that simple linear models could overlook. In general, the matrix helps us understand how different project features are connected and helps us choose features and understand models.

Discussion

This study looked into how well ensemble machine learning methods like Random Forest and Gradient Boosting work for estimating the costs of IT projects when there isn't a lot of data available. This is a prevalent problem in emerging economies like Kazakhstan. The results show that both ensemble models are far better at making predictions and explaining things than classic estimating approaches like COCOMO and Function Point Analysis. The Random Forest model did the best overall, with a R^2 value of 0.603 and the lowest mean absolute error (MAE = 0.09). This means that Random Forest is especially good at handling noisy, high-dimensional datasets and figuring out how project features interact with each other in complicated ways. Gradient Boosting also did well (R^2 = 0.557, MAE = 0.10), but it was more sensitive to hyperparameter adjustment and had a somewhat larger chance of overfitting. These results are in keeping with previous studies that showed how strong ensemble approaches are when it comes to predictive modeling tasks where the relationships between variables are complicated and not linear. The study has some flaws, even though the results are promising. First, synthetic data increases the amount and variety of training data, but it is still an approximation and may not represent all the subtleties of real-world projects. Second, the models were tested on a relatively small combined dataset. More testing on larger datasets that are just real-world would make the results more generalizable. Lastly, the models made big improvements over traditional methods, but their R^2 values show that a lot of the differences in project costs are still unexplained. This shows how hard it is to estimate costs for software development projects. Future studies could look into adding more elements, like organizational aspects, team experience levels, and technology stacks, which could make predictions even more accurate. Also, using ensemble learning with sophisticated explainability tools could help project managers trust and use the system more, since they would be able to not only estimate costs but also comprehend what makes such predictions.

Conclusion

The study investigated how to use Random Forest and Gradient Boosting, two ensemble machine learning methods, to estimate the costs of IT projects in places where there isn't much data, like Kazakhstan's growing IT sector. The results showed that ensemble models are far superior to standard cost estimation methods, even when there aren't always enough data to make them work. They have lower error rates and stronger explanatory power. One of the most important new ideas in this research was using synthetic data production to add to small real-world datasets. This method

helped the models learn more general patterns and cut down on overfitting. It was a useful option for situations where historical project data is missing or hard to get to. Feature importance research showed that project size and development time are the most important factors in determining IT project costs, which is in line with accepted software engineering principles. But the moderate overall R^2 values show that more research is needed to fully understand the many factors that affect costs in software development projects. In general, the results show that using ensemble machine learning methods with synthetic data could improve the accuracy of cost estimates in situations where there isn't much data available. This work gives project managers and decision-makers useful information about how to find more accurate cost forecasting tools, which will help them plan and allocate resources better in IT projects.

REFERENCES

1 Bach, M.P., Topalovic, A., Krstic, Z., Ivec, A. Predictive maintenance in industry 4.0 for the SMEs: A decision support system case study using open-source software. Designs, 7, 98 (2023). https://doi.org/10.3390/designs7040098

2 Suleiman, Z., Shaikholla, S., Dikhanbayeva, D., Shehab, E., Türkyılmaz, A. Industry 4.0: Clustering of concepts and characteristics. Cogent Engineering, 2034264 (2022). https://doi.org/10.1080/23311916.2022.2034264

3 Çakır, M., Güvenç, M.A., Mıstıkoğlu, S. The experimental application of popular machine learning algorithms on predictive maintenance and the design of IoT-based condition monitoring system. Computers & Industrial Engineering, 151, 106948 (2021). https://doi.org/10.1016/j.cie.2020.106948

4 Sembina, G., Aitim, A., Shaizat, M. Machine learning algorithms for predicting and preventive diagnosis of cardiovascular disease. In: 2022 International Conference on Smart Information Systems and Technologies (SIST), 1–5 (2022). https://doi.org/10.1109/sist54437.2022.9945708

5 Sarker, I.H. Machine learning: Algorithms, real-world applications and research directions. SN Computer Science, 5 (2021). https://doi.org/10.1007/s42979-021-00592-x

6 Fernandes, M., Corchado, J.M., Marreiros, G. Machine learning techniques applied to mechanical fault diagnosis and fault prognosis in the context of real industrial manufacturing use-cases: A systematic literature review. Applied Intelligence, 52, 14246–14280 (2022). https://doi.org/10.1007/s10489-022-03344-3

7 Frankó, A., Hollósi, G., Ficzere, D., Varga, P. Applied machine learning for IoT and smart production – Methods to improve production quality, safety and sustainability. Sensors, 22, 9148 (2022). https://doi.org/10.3390/s22239148

8 Kane, A.P., Kore, A.S., Khandale, A.N., Nigade, S.S., Joshi, P.P. Predictive maintenance using machine learning. arXiv, 2205.09402 (2022). https://doi.org/10.48550/arxiv.2205.09402

9 Arboretti, R., Ceccato, R., Pegoraro, L., Salmaso, L. Design of experiments and machine learning for product innovation: A systematic literature review. Quality and Reliability Engineering International, 38, 1131–1156 (2021). https://doi.org/10.1002/qre.3025

10 Aitim, A., Sembina, G. Modeling of human behavior for smartphone using machine learning algorithm. News of the National Academy of Sciences of the Republic of Kazakhstan. Physico-Mathematical Series, 4, 17–28 (2024). https://doi.org/10.32014/2024.2518-1726.304

11 ëSarker, I.H. Machine learning: Algorithms, real-world applications and research directions. SN Computer Science, 2, 5 (2021). https://doi.org/10.1007/s42979-021-00592-x

12 Fernandes, M., Corchado, J.M., Marreiros, G. Machine learning techniques applied to mechanical fault diagnosis and fault prognosis in real industrial manufacturing use-cases: A systematic literature review. Applied Intelligence, 52, 14246–14280 (2022). https://doi.org/10.1007/s10489-022-03344-3

13 Srivastava, P., Srivastava, N., Agarwal, R., Singh, P. NEMAEP: A novel ensemble machine learning framework for accurate effort estimation in software projects. Journal of Advanced Research in Technology and Engineering, 102(24), 9112–9125 (2024).

14 Aitim, A. Developing methods for automatic processing systems of Kazakh language. KazATC Bulletin, 133(4), 254–265 (2024). https://doi.org/10.52167/1609-1817-2024-133-4-254-265

15 Mansoor, F., Alim, M.A., Jilani, M.T., Alam, M.M., Su'ud, M.M. Enhancing software cost estimation using feature selection and machine learning techniques. Computers, Materials & Continua, 79(3), 12345–12367 (2024). https://doi.org/10.32604/cmc.2024.057979

16 Akumba, B.O., Ogala, E., Agaji, I., Akumba, B.T., Blamah, N.V., Otor, S.U. Improving software effort estimation accuracy with a Kalman filter-driven ensemble model. International Journal of Computer Applications, 186(58), 45–59 (2024).

17 Seilo, J. Artificial intelligence in software project cost estimation. Bachelor's thesis, Lappeenranta–Lahti University of Technology LUT, 33 p. (2025).

18 Alhazmi, O.H., Khan, M.Z. Software effort prediction using ensemble learning methods. Journal of Software Engineering and Applications, 13(7), 143–158 (2020). https://doi.org/10.4236/jsea.2020.137010

19 Aitim, A. Building a high-quality annotated corpus for Kazakh NLP: A pipeline approach. Vestnik KazUTB, 4(29) (2025). https://doi.org/10.58805/kazutb.v.4.29-1092.

20 Ahmed, B.M. Predicting software effort estimation using machine learning techniques. In: 2018 8th International Conference on Computer Science and Information Technology, 249–256 (2018). https://doi.org/10.1109/CSIT.2018.8486222

21 Zubair, K.M. Particle swarm optimisation based feature selection for software effort prediction using supervised machine learning and ensemble methods: A comparative study. Invertis Journal of Science & Technology, 13, 33–50 (2020).

[1]*Әйтім Ә.Қ.,
магистр, ORCID ID: 0000-0003-2982-214X,
*e-mail: a.aitim@iitu.edu.kz
[1]Сембина Г.К.,
т.ғ.к., ORCID ID: 0000-0003-2920-1490,
e-mail: g.sembina@iitu.edu.kz

[1]Халықаралық ақпараттық технологиялар университеті,
Алматы қ., Қазақстан

ДЕРЕКТЕР ТАПШЫЛЫҒЫ ЖАҒДАЙЫНДА IT-ЖОБАЛАРДЫҢ ҚҰНЫН БАҒАЛАУҒА АРНАЛҒАН МАШИНАЛЫҚ ОҚЫТУДЫҢ АНСАМБЛЬДІК ӘДІСТЕРІ

Аңдатпа

IT-жобалардың құнын дәл болжау сәтті жоспарлау, бюджеттеу және ресурстарды бөлу үшін шешуші мәнге ие. Алайда COCOMO моделі, функционалдық нүктелерді талдау немесе сараптамалық бағалау сияқты дәстүрлі бағалау әдістері, әсіресе Қазақстан сияқты дамушы елдерде, алдыңғы жобалар бойынша деректердің жеткіліксіздігі немесе толық еместігі жағдайында, әрдайым сенімді нәтижелер бере бермейді. Осы зерттеуде бастапқы деректердің тапшылығы жағдайында IT-жобалардың құнын болжау үшін машиналық оқытудың ансамбльдік алгоритмдерін, атап айтқанда Random Forest және Gradient Boosting әдістерін қолдану қарастырылады. Деректердің жетіспеушілігі мәселесін шешу мақсатында синтетикалық деректерді генерациялау әдістері пайдаланылды, бұл нақты жағдайларда байқалатын статистикалық сипаттамаларды сақтай отырып, жобалардың әртүрлі сценарийлерін модельдейтін кеңейтілген деректер жиындарын қалыптастыруға мүмкіндік береді. Ұсынылған модельдер шығынды болжау үшін жобаның негізгі параметрлерін, атап айтқанда команда мөлшерін, жобаның күрделілігін, әзірлеу әдіснамасын және жоба көлемін кіріс деректері ретінде пайдаланады. Эксперименттік нәтижелер ансамбльдік тәсілдердің болжау дәлдігі бойынша стандартты бағалау әдістерінен жоғары екенін көрсетті. Random Forest моделі ең төмен орташа абсолюттік қателікті (MAE = 0,09) және ең жоғары детерминация коэффициентін (R^2 = 0,603) көрсетті. Сонымен қатар, белгілердің маңыздылығын талдау жоба көлемі мен әзірлеу уақыты шығынды бағалауда ең маңызды факторлар екенін анықтады. Алынған нәтижелер жобалар параметрлері арасындағы күрделі, сызықтық емес тәуелділіктермен жұмыс істеуде ансамбльдік оқытудың тиімділігін растайды және сапалы тарихи деректердің болмауы жағдайында шығынды бағалау әдістерін жетілдіруге арналған практикалық құрал ұсынады. Зерттеу интеллектуалдық шешімдерді қолдау жүйелерін дамытуға үлес қосады және IT-жобаларды бюджеттеу мен жоспарлауды жетілдіруге мүдделі дамушы экономикалардағы жоба менеджерлері мен шешім қабылдаушыларға арналған практикалық ұсынымдар береді.

Тірек сөздер: ансамбльдік оқыту, random forest, gradient boosting, машиналық оқыту, деректер тапшылығы, шешімдерді қолдау жүйелері.

[1]*Айтим А.К.,
магистр, ORCID ID: 0000-0003-2982-214X,
*e-mail: a.aitim@iitu.edu.kz
[1]Сембина Г.К.,
к.т.н., ORCID ID: 0000-0003-2920-1490,
e-mail: g.sembina@iitu.edu.kz

[1]Международный университет информационных технологий,
г. Алматы, Казахстан

АНСАМБЛЕВЫЕ МЕТОДЫ МАШИННОГО ОБУЧЕНИЯ ДЛЯ ОЦЕНКИ СТОИМОСТИ ИТ-ПРОЕКТОВ В УСЛОВИЯХ НЕХВАТКИ ДАННЫХ

Аннотация

Точное прогнозирование стоимости ИТ-проектов имеет ключевое значение для успешного планирования, бюджетирования и распределения ресурсов. Однако традиционные методы оценки, такие как COCOMO, анализ функциональных точек или экспертные оценки, часто не дают надежных результатов, особенно в развивающихся странах, таких как Казахстан, где предыдущие данные о проектах скудны или неполны. В настоящем исследовании рассматривается использование ансамблевых алгоритмов машинного обучения, в частности Random Forest и Gradient Boosting, для прогнозирования стоимости ИТ-проектов в условиях недостатка исходных данных. Для решения проблемы нехватки данных применяются методы генерации синтетических данных, позволяющие формировать расширенные наборы данных, моделирующие различные сценарии проектов при сохранении статистических характеристик, наблюдаемых в реальных случаях. Представленные модели используют ключевые проектные параметры, такие как размер команды, сложность проекта, методология разработки и размер проекта, в качестве входных данных для прогнозирования стоимости. Экспериментальные результаты показывают, что ансамблевые подходы превосходят стандартные методы оценки по точности прогнозирования. Модель Random Forest продемонстрировала наименьшую среднюю абсолютную ошибку (MAE = 0,09) и наибольший коэффициент детерминации (R^2 = 0,603). Кроме того, анализ важности признаков показал, что размер проекта и время разработки являются наиболее значимыми факторами в оценке стоимости. Полученные результаты подтверждают эффективность ансамблевого обучения для работы со сложными, нелинейными зависимостями между параметрами проектов и предлагают практический инструмент для совершенствования методов оценки стоимости в условиях отсутствия качественных исторических данных. Данное исследование вносит вклад в развитие интеллектуальных систем поддержки принятия решений и предоставляет практические рекомендации для менеджеров ИТ-проектов и лиц, принимающих решения в развивающихся экономиках, заинтересованных в улучшении бюджетирования и планирования ИТ-проектов.

Ключевые слова: ансамблевое обучение, random forest, gradient boosting, машинное обучение, нехватка данных, системы поддержки принятия решений.