Project Management Ticket Category Classification using Machine Learning Method: A Comparative Study

Julius Alekhine
Department of Informatics
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
6025231036@student.its.ac.id

Ratih Nur Esti Anggraini
Department of Informatics Institut
Teknologi Sepuluh Nopember
Surabaya,Indonesia
ratihnea@if.its.ac.id

Riyanarto Sarno
Department of Informatics
Institut Teknologi Sepuluh Nopember
Surabaya,Indonesia
riyanarto@if.its.ac.id

Abullah Faqih Septiyanto
Department of Informatics Institut
Teknologi Sepuluh Nopember
Surabaya, Indonesia
7025221022@student.its.ac.id

Agus Tri Haryono
Department of Informatics Institut
Teknologi Sepuluh Nopember
Surabaya,Indonesia
7025231020@student.its.ac.id

2024 IEEE International Conference on Artificial Intelligence and Mechatronics Systems (AIMS) | 979-8-3503-5052-4/24/$31.00 ©2024 IEEE | DOI: 10.1109/AIMS61812.2024.10512660

***Abstract*—JIRA Software is used to monitor projects and track errors or bugs. Generally, JIRA tickets issued by the operational team consists of 2 : EI tickets to report any issues and EACR tickets intended for enhancements or developing new report requesting by clients. However, in its usage, there is often mistakes made in classifying JIRA tickets, which impacts the quality and time delivery to clients. It can be ascertained that the errors were solely human errors, not system errors. To predict categorization errors, we have attempted to apply machine learning using several classifiers and compare the results. From various literature studies, we have tried the 6 most commonly used classifiers for classification and prediction purposes. The classifiers we used are: Multinomial Naive Bayes, Random Forest, Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Extra Trees. Meanwhile, the training and test data comparison was set at a ratio of 70:30. The valuation score from the experiment shows that among other classifier, the Extra Trees classifier produces the highest level of accuracy, F1 score, and recall with values of 0.9950, 0.9950, and 0.9899 respectively. To improve the classifier predictive accuracy , we proposed Extra Trees parameter adjustment.**

Keywords : JIRA, machine learning,Extra Trees

I. Introduction

Company X, Limited uses the project management tool application JIRA to create a workflow that facilitates monitoring task assignments and the progress of a project. JIRA is known as an efficient tools to tracking bugs/issue during the process [1] and increase productivity in the long-run[2].

One of the main goal of using JIRA is to deliver products or services on time. Fig 1 shows the simplified JIRA process. In practice, there are often issues where clients send complaint notes due to late delivery according to agreements.

There are several contributing factors, for example: lack of resources, knowledge skill gap and misclassification of assignment in JIRA. The research task focuses on 'Misclassification of assignment tickets in JIRA'.

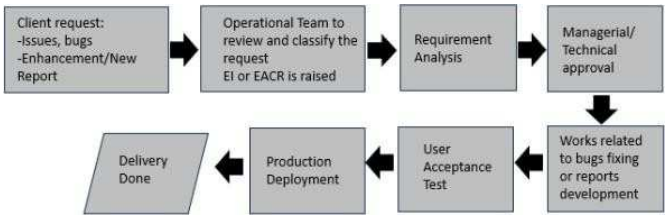

Fig. 1. Simplified JIRA Work Flow

The JIRA ticket assignment consists of 2 categories:

-EI: Used to address issues, bugs in reports, programs, and must be delivered within 72 hours (3 working days).

-EACR: Used for enhancement requests or new requirements from the client. Delivery time is more flexible and longer than using EI.

The client's requests are handled by the operational team before being forwarded to the development team. Hence the operational team is the first party to issue the tickets, either EI or EACR. For various reasons, the operational team might incorrectly classify the ticket, and the problem starts from here. Misclassifying tickets—assigning EACR as EI or vice versa—can significantly impact development time, ultimately resulting in delayed or undelivered products. The risk of delays includes penalties to be paid to the client, but the worst consequence is the company's reputation.

Due to the time-consuming works in the manual requirement analysis, we propose using machine learning to analyze requirement descriptions and predict the categorization of each requirement.

Several challenges faced include: the accuracy level of predictions may not be as high as expected due to incomplete summary descriptions, ambiguous meanings, syntax errors, and the use of abbreviations by the ticket raiser. There have been

several previous researches conducted related to the classification management cases, one of the research implementing supervised machine learning models such as Support Vector Machine, Naive Bayes, Random forest, SGD and XGBoost [3]. Another research with background of misclassifying ticket when using various type of open-source tools like Bugzilla, Gnats and JIRA found the misclassifications is costly[4]. In this research, we proposed to use the Extra Trees algorithm to classify the requirement description. In the conclusion , we also proposed to modify its parameters to enhance the accuracy of text prediction.

II. Proposed Method

The flowchart of the research methodology as shown in Fig 2. The process starts with collecting the JIRA dataset followed by data preprocessing, including duplication removal, category evaluation, and labeling for each JIRA requirement. The data then undergoes cleaning, tokenization, lemmatization, and TF- IDF vectorization technique to gain relevance of words to documents. Next comes the training data phase, where multiple algorithms are applied, and their results are evaluated. Finally, the process moves to prediction testing based on the evaluation. Here is the breakdown.

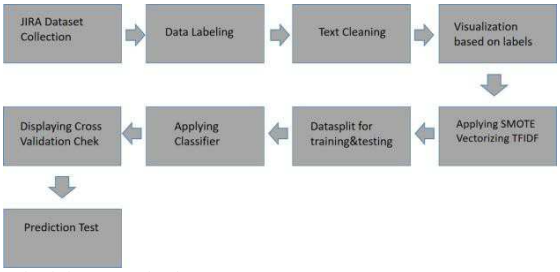

Fig. 2. Methodology Work Flow

A. Jira Dataset Collection

We took samples of JIRA data between July and October 2023. The example JIRA list below, see Table I, demonstrates a categorization error where an EI ticket should have been placed in the EACR category (data in 3rd row). This is because the client's request analysis mandates the developer team to modify the existing program (coding enhancement).

TABLE I. SAMPLE TABLE OF JIRA LIST

Category	Requirement Description	Label
EI	Ricoh PH:Incorrect system calculation of tax	Correct
EI	Palo Alto AU-Incorrect Normal amount in EPay	Correct
EI	**Cognos Report Modification**	**Incorrect**
EACR	60012-Global Blue MY-Create new PE in PDA	Correct
EACR	MSD Policy update	Correct

After done with sorting and removing all duplicates data using SQL , total we managed to obtain 7157 data entries.

B. Data Labeling

During the labeling stage, an analysis was conducted, and a label column was added to indicate whether the data was correct or incorrect based on how well the requirement description aligns with the ticket category.

For instance, a description containing the phrase 'Need to do enhancement in this existing report' should be categorized under EACR, and the JIRA ticket would be EACR-xxxxx (where x is a numeric ticket number). From interpreting the description, the user is requesting changes or modifications to an existing report/application. Another example, a description containing sentences like 'Please check the result...' or 'The amount in the salary report is not correct', the interpretation tends to categorize these descriptions as errors or bugs in reports or computational processes, hence the user should place these requests under 'EI'. To train the data, as mentioned earlier, we have added a label column to mark the correctness or incorrectness of our categorization in the dataset.

Label 1 indicates a description aligned with the JIRA category, hence it is correct. Label 0 denotes misclassification, where a requirement that should belong to the EACR category is placed under EI or vice versa . The bar-chart in Fig 3 depicts the total of correct, and incorrect data

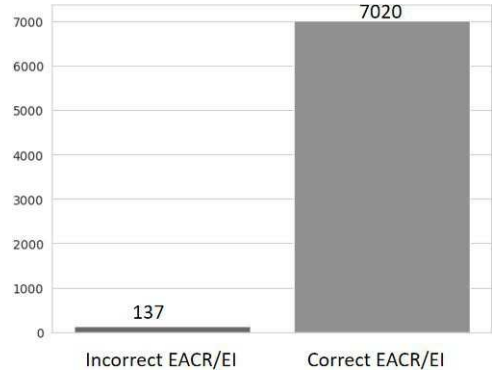

Fig. 3. JIRA Data after labeling

Out of a total of 7157 valid data entries, we identified a total of 7020 correct entries labeled as 1 and 137 incorrect entries labeled as 0 (categorized as EI when it should be EACR, or vice versa).

Furthermore, for data training purposes, we conducted further observation and found the following facts: among the incorrect data, nearly 80 percent contained descriptions related to 'Cognos report' or 'Cognos.' It's important to note that the company utilizes a 3rd party reporting tool i.e. IBM-Cognos, integrated with the payroll application. Typically, users raise JIRA tickets related to Cognos reports under the categories of new development or enhancement. However, there's often confusion, leading them to categorize these tickets under EI, when they should actually be categorized under EACR. Other descriptions include 'to develop' or 'development,' which exhibit similar misclassification patterns as the previous 'Cognos' descriptions."

C. Text Cleaning

The text cleaning stage involves removing stopwords, unnecessary characters, tokenization, and lemmatization. Finally, we checked for any NaN-type data and simultaneously removed it from the dataset when found.

D. Visualization Based On Labels

Using the Wordcloud method to visualize words extracted from JIRA descriptions, where the size of each word depends on how frequently it appears in the overall text. Visualization in Fig 4 and Fig 5 is for the correct and incorrect JIRA categories

respectively. The visualization of Correct JIRA, Fig 4, shows the dominance of words like 'report,' 'payslip,' and 'request.' In the test predictions, all assignments containing these words tend to be categorized under EACR.

On the other side, see Fig 5 , the visualization of Incorrect JIRA is dominated by 'Cognos,' 'new,' and 'report.' With the exception of the word 'report,' which is evenly distributed between correct and incorrect categorization, predictions involving the word 'report' should also consider other words present in the same description

Fig. 4. Correct JIRA Words

Fig. 5. Incorrect JIRA Words

E. Applying Smote

Because the gap between correct and incorrect data is quite wide, thus we conducted oversampling using the SMOTE method to generate synthetic data. This helps balance the quantity of correct and incorrect data, thus improving prediction accuracy [5]. Fig 6 shows the number of data before and after applying SMOTE

	Total Correct Data	Total Incorrect Data
Before SMOTE	7020	137
After SMOTE	7020	7020

Fig. 6. Before and After SMOTE

F. Vectorizing Tf-Idf

The TF-IDF vectorization algorithm is used to assign weights to words in relation to the entire document [3]. The weight of the relationship between a word and a document will be high if the frequency of that word's occurrence is relatively high. Technically TF-IDF will transform text data into a numerical format that machine learning algorithms can work with effectively. It's a technique that represents the importance of words in a document within a larger collection. According to Shahzad et al. [6] the greater or higher occurrence of a word in documents will give higher term frequency.

G. Applying Classifier

We used the following:

1. **Multinomial Naive Bayes (MNB)** : It is a probabilistic classification algorithm that's commonly used in natural language processing (NLP) and text classification tasks[7].
2. **Random Forest (RF)** : Usually used for both classification and regression purposes. It is an ensemble model that generates several trees before performs voting between them.
RF is consider one of the prominent technique for handling imbalanced data[8]
3. **Logistic Regression (LR)**: It is a statistical method used for binary classification and to be used to obtain the odds ratio in the presence of more than one explanatory variable[9]
4. **Support Vector Machine (SVM)**: SVM is a linear model, usually used for sentiment analysis. [10] [11]. It is basically an algorithm that allows determining the best decision boundary between vectors that belong to a specific group and those that do not [12]]
5. **K-Nearest Neighbors (KNN)**: KNN is a simple and effective supervised learning algorithm used for both classification and regression tasks in various fields[13][14].
6. **Extra Trees(ET)**: It's a short for Extremely Randomized Trees or Extra-Trees, is an ensemble learning method based on decision tree algorithms. It's similar to Random Forests but with some notable differences in how it builds the individual trees[15].

III. Experiments and Results

At first, we split the dataset into training and testing data. The ratio between training and testing data is 70:30 We set all parameters for the six classifiers as shown in Table II.

TABLE II. PARAMETERS SETTING FOR EACH MODEL

Classifier	Parameters
MNB	alpha=1.0, fit-prior=True
RF	n-estimators=100,max-depth=None,max-features=auto
LR	penalty = l2, max-iter = 100
SVM	kernel = linear, random state = 0
KNN	n-neighbors = 5
ET	n-estimators=150, random-state=50

We were using classification report to produce the metric result for each classifier we test. The result were evaluated using following method.

Recall: Represents the ability of the classifier to identify all relevant instances. It is the ratio of correctly predicted positive observations to the all observations in the actual class.

F1-score: It's the harmonic mean of precision and recall. It gives a balance between precision and recall. It is very useful metric when there is imbalanced class (correct and incorrect JIRA).

Support: Indicates the number of actual occurrences of the class in the specified data set.

The classification score can be seen in Table III, and Fig 7 is the display of combined confusion matrix in addition to each classifier.

TABLE III. CLASSIFICATION SCORE FOR EACH CLASSIFIER

Classifier	Metric	precision	recall	f1-score	support
Multinomial Naive Bayes	JIRA incorrect	0.9245	1.0000	0.9608	2132
	JIRA correct	1.0000	0.9163	0.9563	2080
	accuracy			0.9587	4212
	macro avg	0.9623	0.9582	0.9586	4212
	weighted avg	0.9618	0.9587	0.9586	4212
Random Forest	JIRA incorrect	0.9893	1.0000	0.9946	2132
	JIRA correct	1.0000	0.9889	0.9944	2080
	accuracy			0.9945	4212
	macro avg	0.9947	0.9945	0.9945	4212
	weighted avg	0.9946	0.9945	0.9945	4212
Logistic Regression	JIRA incorrect	0.9586	1.0000	0.9789	2132
	JIRA correct	1.0000	0.9558	0.9774	2080
	accuracy			0.9782	4212
	macro avg	0.9793	0.9779	0.9781	4212
	weighted avg	0.9791	0.9782	0.9781	4212
Support Vector Machine	JIRA incorrect	0.9731	1.0000	0.9864	2132
	JIRA correct	1.0000	0.9716	0.9856	2080
	accuracy			0.9860	4212
	macro avg	0.9865	0.9858	0.9860	4212
	weighted avg	0.9864	0.9860	0.9860	4212
K-Nearest Neighbors	JIRA incorrect	0.8850	1.0000	0.9390	2132
	JIRA correct	1.0000	0.8668	0.9287	2080
	accuracy			0.9342	4212
	macro avg	0.9425	0.9334	0.9338	4212
	weighted avg	0.9418	0.9342	0.9339	4212
Extra Trees	JIRA incorrect	0.9902	1.0000	0.9951	2132
	JIRA correct	1.0000	0.9899	0.9949	2080
	accuracy			0.9950	4212
	macro avg	0.9951	0.9950	0.9950	4212
	weighted avg	0.9951	0.9950	0.9950	4212

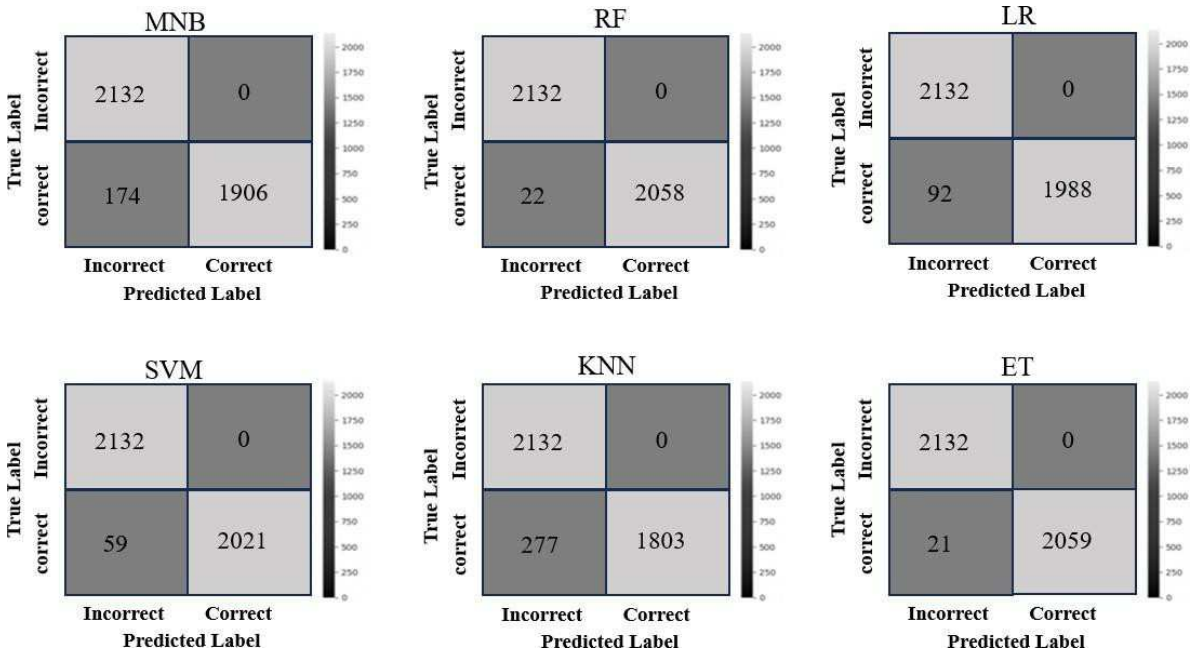

Fig. 7. Confusion Matrix in addition to each Classifier

As the final phase, we used Cross Validation Check as it provides the ability to estimate model performance on unseen data not used while training. Result as seen in Table IV, it is rounded to the nearest 4 decimal places.

TABLE IV. CROSS VALIDATION CHECK

Model	Accuracy	ROC-AUC	Precision	Recall	F1-Score
MNB	0.9587	0.9581	1	0.9163	0.9563
LR	0.9781	0.9779	1	0.9558	0.9774
RF	0.9940	0.9940	1	0.9880	0.9940
SVM	0.9860	0.9858	1	0.9716	0.9856
KNN	0.9342	0.9334	1	0.8668	0.9287
ET	**0.9950**	**0.9950**	**1**	**0.9899**	**0.9950**

Using the Cross Validation Check we can assess how well the model performs across different subsets of the data. As the result of the six models we experimented with , Extra Trees performs its superiority over the other models.

H. The Prediction Test Result

Based on the metric results, we conducted prediction tests using Extra Trees. The result , although Extra Trees showed high predic- tion scores, during the testing some text containing 'Cognos', ET providing some unexpected prediction. It returned the category under EI ticket, whereas we expected ET to be assigned under EACR. Consequently, we repeated the experiment specifically for ET by adjusting the parameters to n-estimators=300 and random- state=100 instead of its default set i.e. n-estimators=150 and random-state=50.

Despite no change in the metric result, but it is able to provide prediction as expected. Table V provides the default and proposed new parameter for ET.

TABLE V. PROPOSED ET PARAMETERS ADJUSTMENT

Classifier	Extra Trees
Default Parameters	n-estimators=150, random-state=50
Proposed New Parameters	**n-estimators=300, random-state=100**

The n-estimators controls the number of decision trees in the ensemble, having more estimators can lead to better performance but increase computational cost. While random-state ensures reproducibility by setting the seed for random number generation.

IV. Conclusion and Future Work

In this study we analyzed usage of JIRA dataset for ticket classification, as we found the human error made during the classification phase. We initiated to implementing machine learning to gain for early error detection in classification. Using the machine learning to help in finding the possibility of errors. We chose six different models to determine a prediction.From the classifier application results, it was found that the Extra Trees (ET) and Random Forest (RF)classifier exhibited the higher accuracy compared to other classifiers. Basically both ET and RF using the same method approach, namely the ensemble method. Despite RF shows promising outcome yet ET produce more robust result. ET is an ensemble machine learning approach that trains numerous decision trees and aggregates the results from the group of decision trees to output a prediction. We have chosen ET classifier to test the category prediction based JIRA description request.Higher score may not in line with the expectations from the prediction. Some of the obstacles that affects prediction results is the writing of descriptions that are too brief, the use of uncommon abbreviations, writing errors, and biased interpretations. Hence, to have better result in prediction we propose to adjust ET parameter as seen in the experiment section. Considering that one of the challenges is dealing with the ambiguity of a requirement description meaning, therefore our future work will include the BERT (Bidirectional Encoder Representa- tions from Transformers) method to reduce misinterpretation. As BERT is designed to understand the context of words in a sentence by considering both left and right context. This two-way method enables the model to grasp more subtle connections among words. Eventually this will result in achieving higher accuracy values.

References

[1] M. Ortu, G. Destefanis, M. Kassab, and M. Marcesi, ”Measuring and Under- standing the Effectivenees of JIRA Developers Community”, ResearchGate, Conference Paper May 2015

[2] A. Arnautović,”Managing Project Using JIRA Software”,Serbian Journal of Engineering Management,Vol. 7, No. 2, 2022

[3] S. Gopirajan Chitra,”Classification of low-level tasks to high-level tasksusing JIRA data”,Universidade do Porto (Portugal),2021

[4] N. Pandey, D. K. Sanyal, A. Hudait, and A. Sen,”Automated classifi- cation of software issue reports using machine learning techniques: an empirical study”,Innovations in Systems and Software Engineering,13,279– 297,Springer,2017

[5] N. V. Chawla and K. Bowyer, ”SMOTE: Synthetic Minority Over-sampling Technique”, Journal of Artifial Intelligence Research - June 2022

[6] S. Qaiser and R. Ali, ”Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents”, International Journal of Computer Applications (0975-8887) Vol 181-No 1 July 2018

[7] S. Ruan, H. Li, C. Li, and K. Song,”Class-Specific Deep Feature Weighting for Na¨ıve Bayes Text Classifiers”,IEEE Access ,10.1109/AC-CESS.2020.2968984, 2020

[8] N. Jalal, A. Mehmood, G. S. Choi, and I. Ashraf,”A novel improved random forest for text classification using feature ranking and optimal number of trees”, ScienceDirect,Journal of King Saud University-Computer and Infor- mation Science, 34, 2022

[9] A. Pal,”Logistic Regression: a Simple Primer”, Cancer Research, Statistics, and Treatment ,4(3):p 551-554, Jul–Sep 2021.

[10] N. Zainuddin and A. Selamat,”Sentiment analysis using support vector machine”. In: 2014 international conference on computer, communications, and control technology (I4CT). p. 333–337,IEEE; 2014

[11] M. Fikri and R. Sarno,”A comparative study of sentiment analysis using SVM and SentiWordNet”, Indonesian Journal of Electrical Engineering and Computer Science. vol. 13, no 3. p. 902–909, 2019.

[12] S. Al Hasan, M. G. Hussain, J. Protim, M. M. Rahman, N. Fahim, M. Z. Chowdhury, and A. I. Pritom,”Classification of Multi-Labeled Text Articles with Reuters Dataset using SVM”,International Conference on Science and Technology (ICOSTECH), IEEE, 2022

[13] G. Guo, H. Wang, D. Bell, Y. Bi,and K. Greer,”KNN Model-Based Approach in Classification”,Researchgate,2004

[14] G. Vinayakumar,”A Comparison of KNN Algorithm and MNL Model for Mode Choice Modelling”,European Transport Trasporti Europei ,Issue 92, Paper n 3, ISSN 1825-3997, 2023

[15] A. Kharwar and D. Thakor,”An Ensemble Approach for Feature Selection and Classification in Intrusion Detection Using Extra-Tree Algorithm”, ResearchGate Journal, January 2022