International Journal of Application on Sciences, Technology and Engineering
(IJASTE)
Volume 1, Issue 4, 2023. ISSN:2987-249

JOINT K-MEANS AND MODIFIED KNN FOR FAULT RESOLVING TIME PREDICTION OF TELECOMMUNICATION TROUBLE TICKET

Indri Yani Berutu

Faculty of Information Technology, Tarumanagara University
Email : Indriyaniberutu22@gmail.com

Submitted: 27-09-2023, Revised: 27-10-2023, Accepted: 11-12-2023

ABSTRACT

Efficiently resolving telecommunication trouble tickets is crucial for maintaining network reliability and customer satisfaction. This paper proposes a novel approach that combines the power of K-Means clustering and a modified K-Nearest Neighbors (KNN) algorithm to predict the fault resolving time of telecommunication trouble tickets. By leveraging K-Means clustering, the trouble tickets are grouped into clusters based on similarity, allowing for more accurate fault resolution time predictions within each cluster. The modified KNN algorithm further refines these predictions by considering the historical performance of similar tickets. Experimental results demonstrate that the joint K-Means and modified KNN approach significantly enhances the accuracy of fault resolving time predictions, thereby improving service quality and operational efficiency in telecommunication networks.

1. Introduction

In the telecommunications industry, when a telecommunication device encounters issues such as disconnection or disruptions, companies or service providers receive problem reports through two methods: automated and manual, known as Trouble tickets. Automated Trouble tickets are generated based on alarms sent by telecommunication devices to a database. This automated process allows for quicker and more efficient issue responses. On the other hand, manual Trouble tickets are created by Field Maintenance Engineers (FMEs) who are on-site experts responsible for maintaining and resolving issues with the devices. When FMEs encounter problems or failures, they manually create Trouble tickets to record and track the issue resolution process. However, a crucial aspect of issue handling is accurately estimating the resolution time of Trouble tickets. Precise and timely resolution time estimates are vital for swift issue responses and prioritization.

Estimating the resolution time for Trouble tickets can be complex due to diverse information, including timestamps, site and alarm IDs, and fault levels. Typically, complex systems use a multiple model learning technique, which involves creating a set of local prediction models and analyzing the system in parts. Data groups are predicted using clustering techniques, and a prediction model is crafted for each group. This approach enhances prediction accuracy. Therefore, to improve the accuracy of Trouble ticket resolution time estimates, a combination of clustering and prediction approaches is utilized.

K-Nearest Neighbor (KNN) is a prediction technique that determines the label of a new data point based on the labels of the K data points with the highest similarity to the new data point. KNN offers advantages such as ease of implementation, noise handling, and no need for model training. However, it has limitations like sensitivity to the K parameter, computational overhead, and difficulty handling class imbalances. To address these limitations, Modified K-Nearest Neighbor (MKNN) was developed to handle class imbalances and enhance accuracy.

K-Means is an iterative algorithm used to cluster datasets into K subgroups (clusters). Each iteration assigns data points to clusters with high similarity and homogeneity. The process involves calculating the distance between each data point and the centroid of its cluster, where the centroid is the average value of all data points in the cluster. The iteration continues until the cluster members remain the same as in the previous iteration.

In the telecommunications industry, a system for estimating Trouble ticket resolution times is developed by combining the Modified K-Nearest Neighbor (MKNN) prediction method and K-Means clustering. The resulting system aims to assist telecommunication service providers in improving the quality of telecommunication repair services by enhancing Trouble ticket resolution time estimates.

2. Method

This section will explain some of the methods used and also evaluation calculations from the models that have been made. Then It will be explained about the data used and finally, the analysis stages of this paper will be explained.

2.1. Trouble ticket

A Trouble Ticket, also referred to as a Trouble Report, is a system used within organizations to monitor, report, and manage various complex situations. It initially began as a paper-based issue reporting system but has evolved into a primarily web-based system integrated with Customer Relationship Management (CRM) platforms. These platforms are commonly used in call centers, e- commerce websites, and advanced technology environments like Network Operations Centers (NOCs).

In this research, data collected from Trouble Tickets includes 10 properties, such as Ticket ID, Title, Ticket Status, Business Status, Created On, Closed On, Site ID (Create TT), Fault Level (Create TT), Alarm Name (Create TT), and Alarm ID (Create TT). Each property serves a crucial role. Ticket ID provides unique identification, Title offers a description of the issue, and Ticket Status and Business Status track the ticket's status and its impact on the business. Created On records the issue detection time, Closed On indicates when the issue was resolved, while Site ID (Create TT) and Site Name (Create TT) help pinpoint the issue's physical location. Fault Level (Create TT) and Alarm Name (Create TT) are used to gauge the severity and identify the source of the issue within the Trouble Ticket.

For this study, only four properties are utilized: Created On, Closed On, Site ID (Create TT), and Alarm Name (Create TT). These properties were selected due to their significant influence on the analysis and modeling of Trouble Tickets and their relevance in predicting resolution times.

2.2. K-Means

The K-Means method is one of the techniques in Cluster Analysis used to group data into several clusters based on the similarity between each data point. In this method, the number of clusters is initialized beforehand, and each data point is assigned to the cluster with the nearest Centroid or cluster center.

K-Means is a versatile method that can be applied to various types of data, such as sales data, social data, and biological data. Particularly in business data analysis, pattern recognition, and machine learning, the K-Means method is often used as an effective and efficient clustering algorithm.
The steps to perform the K-Means algorithm are as follows:

1. Determine the desired number of clusters (K) and the distance metric to be used for calculating dissimilarity between data points. If necessary, also set a threshold for changes in the objective function and the centroid positions.

2. Select K data points from the dataset as the initial centers (centroids) for each cluster.

3. Allocate each data point to the nearest cluster based on the predefined distance metric.

4. Recalculate the centroid positions based on the data points assigned to each cluster.

5. Repeat steps 3 and 4 until convergence is achieved, which happens when (a) the change in the objective function is smaller than the predefined threshold, or (b) no more data points change clusters, or (c) the change in centroid positions is smaller than the specified threshold.

Distance formula for each data to the K-Means centeroid:

$$D(X_1, X_1) = \|x1 - x2\| = \sqrt{\sum_{j=1}^{p} |X_{2j} - X_{1j}|^2}$$

Explanation:
1. $X_1$and X_2 are two vectors or points in a p-dimensional space (where p is the number of variables possessed by those vectors).
2. $|X_{2j} - X_{1j}|$represents the difference between the value of variable j in X_2 and the value of variable j in X_1.
3. $\sum_{j=1}^{p} |X_{2j} - X_{1j}|^2$ is the result of summing the squares of the differences in variable values

between X_2 and X_1 for all variables (j) from 1 to p.

4. √(...) is the square root of the sum of squared differences in variable values between the two vectors. This corresponds to the definition of Euclidean distance, which is the distance (Euclidean distance) between two points in Euclidean space.

So, $D(X_1, X_2)$ measures the Euclidean distance between two points, X_1 and X_2, in a p-dimensional space. A smaller value of $D(X_1, X_2)$ indicates that the two points are closer in the p-dimensional space, while a larger value of $D(X_1, X_2)$ indicates that the two points are farther apart in the p-dimensional space.

2.3. Modified K-Nearest Neighbor (MKNN)

The Modified K-Nearest Neighbor (MKNN) algorithm is a variation of the K-Nearest Neighbor (KNN) method that involves two additional steps: weight calculation and validity evaluation. In contrast, the KNN algorithm is a relatively straightforward approach that classifies new data by considering only the nearest K values [10].

Modified K-Nearest Neighbor (MKNN) is a strategy for assigning class labels to new data based on predefined validation data, using the K-Nearest Neighbor (KNN) approach that takes into account weights. For distance calculation, Hamming Distance is used to measure how different two data points (in the form of feature vectors) are from each other. This method is useful in determining the nearest neighbors in KNN, which are data points with the smallest Hamming Distance to the data point being predicted. The Hamming Distance formula is as follows:

$$\text{Hamming Distance} = \sum_{i=1}^{n} \partial(x_1, y_1)$$

N: The number of features or attributes in each data point.
Xi: The value of the i-th feature in the first data point.
yi: The value of the i-th feature in the second data point.
δ(xi, yi): Delta function (1 if xi and yi are different, 0 if they are the same).

2.4. DBI

The Davies-Bouldin Index (DBI) is an evaluation method used to measure the quality of clustering in data analysis. DBI is obtained by comparing the average distance between data points within one cluster to the average distance between data points in different clusters. A smaller DBI value indicates a better quality of clustering. DBI is useful for determining the optimal number of clusters in clustering [5].

DBI serves as one of the quality evaluation indicators for a clustering result. The optimal DBI value depends on the number of clusters used. The primary purpose of using DBI is to find the optimal number of clusters. This can be achieved by comparing DBI values among different clustering models that use different numbers of clusters. The clustering model that produces the smallest DBI value can be considered the best clustering model [4].

$$SSW = \frac{1}{N} \sum_{i=1}^{N} \|x_1 - C_{pi}\|^2$$

Explanation:

1. SSW (Sum of Squares Within) is the sum of squared distances between each data point (x_1) and the cluster center (C_{pi}) of the same cluster as that data point. In this context, SSW measures how far each data point is from the cluster center within the same cluster.
2. N is the total number of data points in the dataset.
3. x_1 is the i-th data point in the dataset.
4. C_{pi} is the i-th cluster center calculated based on the average of all data points belonging to that cluster.
5. $\|x_1 - C_{pi}\|^2$is the squared Euclidean distance between the data point $x1$ and the cluster center C_{pi}.
6. $\sum_{i=1}^{N}$ ||| represents the average of the sum of squared distances between each data point and the cluster center within the same cluster.

2.5. MAPE

Mean Absolute Percentage Error (MAPE) is a statistical metric used to measure the level of prediction or forecast error in forecasting methods. MAPE quantifies the average error as a percentage of the actual values in the dataset. This method is commonly used to assess prediction accuracy and can be easily understood and applied by many individuals. Here is the formula:

$$\text{MAPE} = \frac{1}{n} \sum_{t=1} \left| \frac{At - Ft}{At} \right|_{\text{N}}$$

MAPE	: Mean Absolute Percentage Error
At	: Actual value at time t in the dataset.
Ft	: Forecasting or prediction results at time t in the dataset.
N	: The total number of observations or times in the dataset.

3. Discussion

3.1. Combining K Means and MKNN for Prediction

Figure 1 illustrates the data processing stages regarding the integration of K-Means clustering with MKNN. Firstly, the trouble ticket data will undergo data preprocessing. The trouble ticket data will be transformed and formatted to meet the analysis requirements. Then, the K-Means Clustering algorithm will be applied to group the trouble ticket data into multiple clusters. This process begins with the selection of an appropriate value for k (the number of clusters). Once the clusters are formed, classification is performed for each cluster. In this stage, Modified K-Nearest Neighbor (K-NN) is applied to predict the resolution time of trouble tickets within each cluster. Subsequently, model evaluation is carried out using MAPE to measure the effectiveness of the generated model in predicting trouble ticket resolution times.

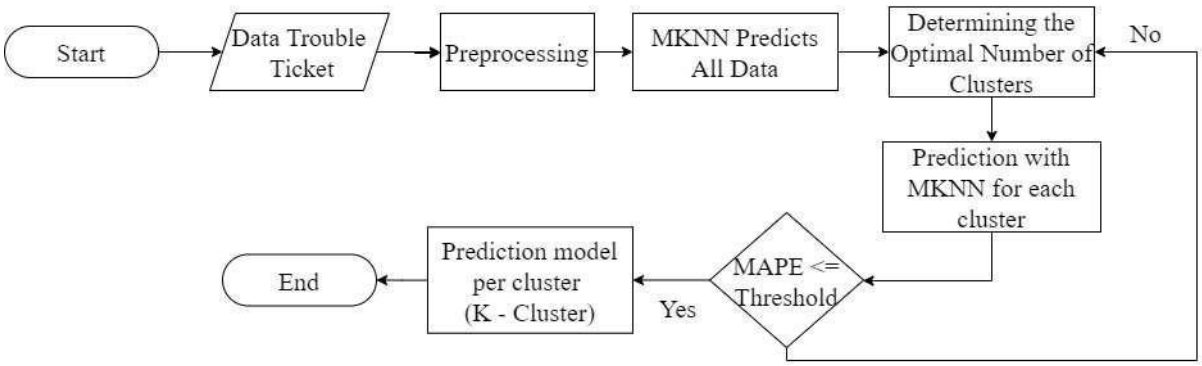

Figure 1. Data Processing Stages

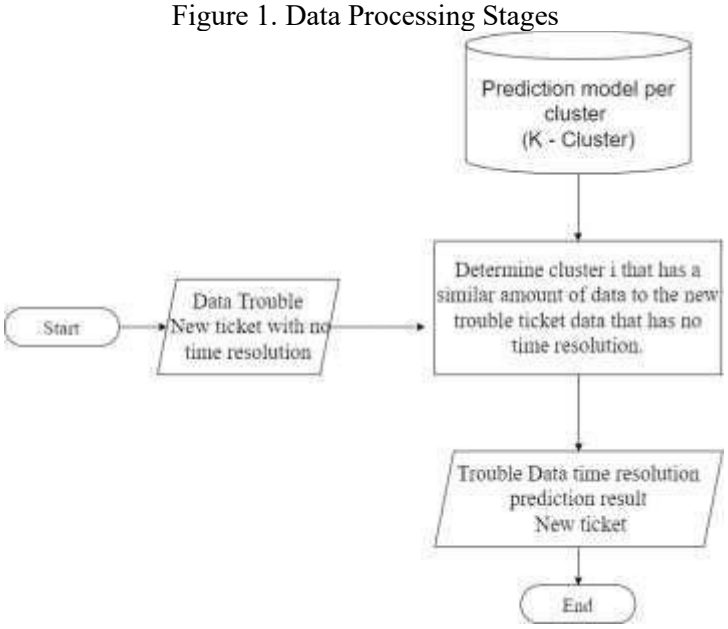

Figure 2. Prediction Stages

4. Conclusion

In the telecommunications industry, trouble tickets can be generated automatically based on alarms sent by devices or manually by Field Maintenance Engineers (FMEs). Estimating the resolution time of trouble tickets is a crucial aspect of problem handling. Estimating the resolution time can be complex as it involves various pieces of information such as creation time, closure time, site ID, alarm ID, and fault level. Therefore, accurate prediction techniques are needed. The approach of combining clustering (K-Means) and prediction (Modified KNN) can enhance the accuracy of estimated trouble ticket resolution times. This overall prediction system is expected to assist telecommunications service providers in improving the quality of telecommunications repair services by providing more accurate estimates of trouble ticket resolution times.

References

[1] A. Z. Siregar, "Implementasi Metode Regresi Linier Berganda Dalam Prediksi Tingkat Pendaftaran Mahasiswa Baru," Kesatria J. Penerapan Sist. Inf. (Komputer dan Manajemen), vol. 2, no. 3, pp. 133–137, 2021.

[2] R. E. Pranata, I. Gunawan, and Sumarno, "Algoritma Backpropagation Dalam Melakukan Prediksi Penjualan Beras Pada CV Hariara Pematangsiantar," J. Comput. Syst. Informatics, vol. 2, no. 2, pp. 210–221, 2021.

[3] M. Ardhiansyah, "Penerapan Model Rapid Application Development pada Aplikasi Helpdesk Trouble ticket PT. Satkomindo Mediyasa," J. Teknol. Sist. Inf. dan Apl., vol. 2, no. 2, p. 43, 2019, doi: 10.32493/jtsi.v2i2.2759.

[4] E. Muningsih, I. Maryani, and V. R. Handayani, "Penerapan Metode K-Means dan Optimasi Jumlah Cluster dengan Index Davies Bouldin untuk Clustering Propinsi Berdasarkan Potensi Desa," J. Sains dan Manaj., vol. 9, no. 1, pp. 95–100, 2021.

[5] R. Muliono and Z. Sembiring, "Data Mining Clustering Menggunakan Algoritma K-Means Untuk Klasterisasi Tingkat Tridarma Pengajaran Dosen," CESS (Journal Comput. Eng. Syst. Sci., vol. 4, no. 2, pp. 2502–714, 2019.

[6] H. D. Wijaya and S. Dwiasnati, "Implementasi Data Mining dengan Algoritma Naïve Bayes pada Penjualan Obat," J. Inform., vol. 7, no. 1, pp. 1–7, 2020, doi: 10.31311/ji.v7i1.6203.

[7] P. A. Rahayuningsih, "Analisis Komparasi Algoritma Klasifikasi Data Mining," J. Tek. Inform. Kaputama, vol. 3, no. 1, 2019.

[8] S. Febriani and H. Sulistiani, "Analisis Data Hasil Diagnosa Untuk Klasifikasi Gangguan Kepribadian Menggunakan Algoritma C4.5," 89Jurnal Teknol. dan Sist. Inf., vol. 2, no. 4, pp. 89–95, 2021.

[9] D. A. I. C. Dewi et al., "Penerapan Metode K-Means dan Optimasi Jumlah Cluster dengan Index Davies Bouldin untuk Clustering Propinsi Berdasarkan Potensi Desa," J. Sains dan Manaj., vol. 9, no. 1, pp. 95–100, 2019, doi: 10.31940/matrix.v9i3.1662.

[10] Y. Pinanda, W. Firdaus Mahmudy, and E. Santoso, "Klasifikasi Risiko Penyakit pada Ibu Hamil menggunakan Metode Modified K-Nearest Neighbor (MKNN)," J. Pengemb. Teknlogi Inf. dan Ilmu Komput., vol. 6, no. 5, pp. 2116–2121, 2022.