



Unsupervised Learning For Managing Safety Accidents In Railway Stations Using Topic Modeling And Clustering Techniques

¹D.Jayasoniya, ²Chunduru Reshma, ³Yamparala Gnana Prasanna,

⁴Sriram Venkata Prasanna Nithin Kumar, ⁵Maddula Yuva Vardhini, ⁶Are Saila Priya

¹Assistant Professor, ^{2,3,4,5,6}UnderGraduate

^{1,2,3,4,5,6} CSE-Data Science Department, St. Ann's College of Engineering & Technology, Chirala, Andhra Pradesh

Abstract: Railway stations are high-traffic environments prone to safety incidents ranging from equipment failures to passenger-related emergencies. Efficient analysis and management of such incidents are crucial for improving operational safety, reducing risk, and informing preventive strategies. This paper presents an unsupervised learning-based approach for extracting actionable insights from unstructured accident logs recorded in railway stations. Natural Language Processing (NLP) techniques are applied to clean, preprocess, and vectorize textual incident reports. The core of the methodology involves applying Latent Dirichlet Allocation (LDA) for topic modeling and K-Means clustering for grouping similar incidents. Dimensionality reduction is achieved using Principal Component Analysis (PCA) to enhance interpretability and visualization. The resulting system enables the identification of recurring safety issues, latent risk factors, and thematic clusters of incidents. A lightweight Streamlit web interface is also developed to support interactive exploration of incident categories and their severity levels. The proposed unsupervised pipeline facilitates effective monitoring and analysis of safety events in complex transit environments without relying on labeled data, offering significant value for railway safety departments and decision-makers.

Index Terms - Unsupervised Learning, Topic Modeling, K-Means Clustering, LDA, Railway Safety, NLP, Streamlit.

I. INTRODUCTION

Railway stations represent critical infrastructures in urban transit systems, managing high volumes of passengers, operational staff, and machinery on a daily basis. Due to this complexity, these stations are also susceptible to various types of safety-related incidents, including slips and falls, crowd surges, technical failures, and environmental hazards. Timely identification and categorization of such incidents are essential for effective risk mitigation, preventive action planning, and policy formulation.

Traditionally, safety incident logs at railway stations are recorded in free-text formats as part of internal reporting systems. These logs often contain valuable insights regarding the nature of incidents, location-specific risks, and severity levels. However, their unstructured nature makes them difficult to analyze using conventional statistical methods or rule-based approaches. Manual inspection of these records is time-consuming, subjective, and infeasible at scale, particularly for large railway networks that generate thousands of incident reports annually.

To address this challenge, this study explores the use of **unsupervised learning** techniques for the automatic categorization and analysis of safety incidents in railway stations. Unlike supervised learning approaches, which require annotated datasets, unsupervised models can identify latent structures and patterns from unlabeled data. This makes them particularly suitable for safety log analysis, where labeled incident categories are often unavailable or inconsistently defined.

The methodology leverages **Natural Language Processing (NLP)** to preprocess raw textual data, including tokenization, stopwords removal, stemming, and TF-IDF vectorization. Subsequently, **Latent Dirichlet Allocation (LDA)** is used for topic modeling to extract dominant themes across incident records, while **K-Means clustering** is applied to group similar incident reports. Dimensionality reduction using **Principal Component Analysis (PCA)** enhances cluster interpretability and supports visualization. To facilitate accessibility and real-time analysis, the entire pipeline is deployed via a user-friendly **Streamlit** interface.

This research provides a practical, scalable, and interpretable approach for incident trend discovery in transit environments. By surfacing recurring accident types and uncovering emerging risk factors, the system supports proactive safety management without requiring labeled datasets or predefined categories. It also serves as a foundational step toward integrating AI-based incident monitoring systems in transportation networks.

The major contributions of this work are summarized as follows:

- Development of an end-to-end unsupervised learning pipeline for safety incident analysis.
- Application of LDA and K-Means to extract dominant topics and clusters from unstructured log data.
- Dimensionality reduction for visualization and interpretation of incident patterns.
- Deployment of a Streamlit-based interface for interactive exploration and reporting.

II. LITERATURE REVIEW

Analyzing safety incidents in complex environments such as railway stations is a critical task in transportation management. Traditional safety auditing methods often rely on manual analysis of incident reports, which can be time-intensive and susceptible to human error. Recent advances in Natural Language Processing (NLP) and unsupervised machine learning have enabled automated analysis of unstructured textual data, offering new possibilities for safety event mining and risk mitigation.

A. Safety Analytics in Transportation

Research on transportation safety has historically focused on structured data, including sensor readings, accident frequencies, and maintenance logs. However, **textual incident reports** remain underutilized despite containing rich context about operational hazards. Studies by Bhattacharyya et al. [1] demonstrated the usefulness of textual analytics in metro systems for identifying recurring risk patterns. Similarly, Singh et al. [2] applied statistical models to railway incident data to predict high-risk zones, though their approach required manually labeled categories.

B. Topic Modeling for Text Mining

Latent Dirichlet Allocation (LDA), introduced by Blei et al. [3], has become a popular technique for discovering latent themes in unstructured corpora. In safety applications, LDA helps uncover hidden structures in incident logs, such as frequently co-occurring terms or event categories. Li et al. [4] applied LDA to aviation safety reports to identify systemic issues in flight operations. LDA has also been used in healthcare, legal, and industrial domains for unsupervised categorization of free-text records.

LDA is particularly suitable for domains lacking labeled data, offering interpretability by assigning probabilities to topics for each document. Its use in railway incident data remains limited, presenting an opportunity for novel applications.

C. Clustering Techniques for Unlabeled Data

Clustering is another core unsupervised learning method used to group data points based on similarity. Among clustering algorithms, **K-Means** is widely used due to its simplicity and efficiency. In the context of textual data, clustering supports the discovery of similar event types or patterns across a dataset. Previous work by Zhao et al. [5] demonstrated the effectiveness of K-Means in categorizing industrial incident reports based on term frequency-inverse document frequency (TF-IDF) vectors.

Although K-Means assumes spherical clusters and may require manual tuning of the number of clusters (K), it performs well when paired with dimensionality reduction techniques such as **Principal Component Analysis (PCA)** for visual analysis.

D. NLP Preprocessing in Safety Domains

Preprocessing plays a critical role in extracting meaningful features from raw text. Common steps include lowercasing, punctuation removal, stopwords elimination, stemming or lemmatization, and vectorization using techniques like TF-IDF or Word2Vec. These preprocessing steps have been shown to significantly improve the quality of downstream topic modeling and clustering. Hasan et al. [6] emphasized the impact of domain-specific stopwords lists and n-gram features in enhancing clustering coherence in safety-critical applications.

E. Streamlit for Interactive Safety Dashboards

While much of the academic work focuses on back-end analysis, few studies integrate unsupervised learning with **interactive dashboards**. The use of **Streamlit**, a Python-based web app framework, allows researchers to create real-time analytical tools with minimal development overhead. Integration of NLP, LDA, and clustering outputs into a web interface offers immediate utility for non-technical users in operational settings.

This review highlights the potential of unsupervised learning techniques—particularly LDA and K-Means—for extracting structured insights from unstructured incident logs. While these techniques have been applied in related domains, their use in railway station safety management remains underexplored. This research fills that gap by presenting an integrated framework that combines these techniques with a web-based interface for practical deployment.

III. PROPOSED METHODOLOGY

This study proposes a structured unsupervised learning framework to analyze unstructured safety incident data collected from railway stations. The methodology transforms raw textual logs into interpretable clusters and topics using NLP and machine learning techniques. The major stages include data preprocessing, vectorization using TF-IDF, topic modeling with Latent Dirichlet Allocation (LDA), clustering using K-Means, dimensionality reduction via Principal Component Analysis (PCA), and deployment of an interactive analysis interface using Streamlit.

A. System Workflow Overview

The overall workflow begins with importing raw text-based safety incident logs, which are typically brief summaries of events recorded in accident registers. The system proceeds through a multi-stage pipeline that cleans and processes these texts, extracts features, identifies underlying themes, and organizes the data into interpretable categories. The final system outputs visualizations and topic summaries for actionable insights.

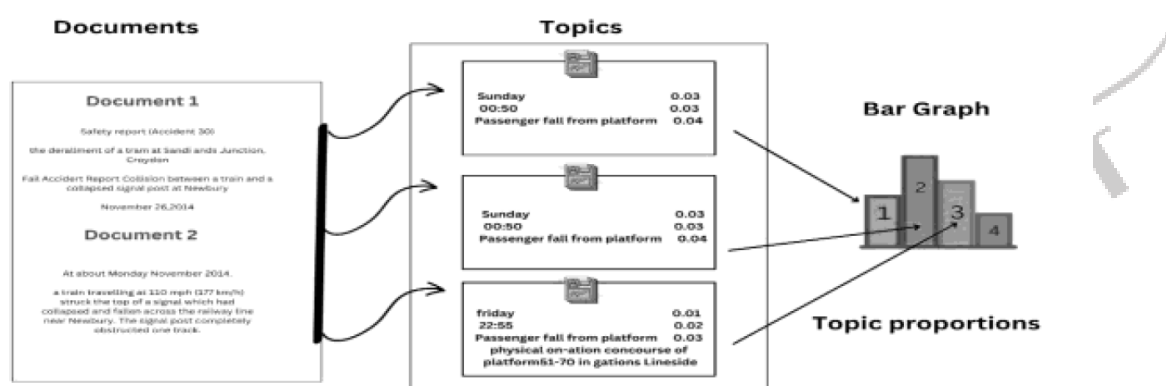


Fig 1. Block Diagram of the Topic Modelling in NLP

B. Data Collection and Preprocessing

The dataset consists of safety incident reports collected from different railway zones. Each report contains a short natural language description of an incident, such as “passenger slipped on wet platform” or “signal failure caused delay.” These records are imported from structured CSV files.

Text preprocessing is carried out using Python’s `re`, `nlTK`, and `pandas` libraries. The preprocessing steps include:

- Lowercasing all characters for normalization.
- Removing punctuation, special characters, and numeric tokens using regular expressions.
- Tokenizing the text into words.
- Removing stopwords using NLTK’s English stopwords list, with custom additions for domain-specific words like “station,” “zone,” or “area.”
- Applying lemmatization using WordNetLemmatizer to reduce words to their base form while retaining semantic meaning.

The cleaned and normalized text is then ready for feature extraction.

C. Feature Extraction Using TF-IDF

After preprocessing, the text is converted into a numerical format suitable for machine learning models using the Term Frequency-Inverse Document Frequency (TF-IDF) vectorizer. TF-IDF helps identify words that are not only frequent but also informative across the dataset.

The TfidfVectorizer from sklearn is configured with:

- max_features = 1000 to restrict vocabulary size,
- min_df = 2 and max_df = 0.85 to ignore rare or overly common terms,
- ngram_range = (1, 2) to include both unigrams and bigrams.

The output is a sparse matrix where each row corresponds to a document and each column represents a term with its TF-IDF weight. This matrix becomes the input to both topic modeling and clustering algorithms.

D. Topic Modeling Using LDA

Latent Dirichlet Allocation (LDA) is employed to identify hidden semantic structures within the dataset. LDA treats each document as a mixture of topics, and each topic as a mixture of keywords. The implementation uses the gensim.models.LdaMulticore module to train the model on the TF-IDF matrix.

The number of topics is initially set to five, based on domain understanding and coherence score analysis. Hyperparameters such as alpha and eta are set to 'auto' for optimal distribution learning. The model is trained with 20 passes and 400 iterations to ensure stability.

The output of the LDA model includes:

- A list of topics, each represented by a group of keywords with probabilities.
- A document-topic distribution that indicates the likelihood of each document belonging to each topic.

These topics represent high-level groupings of incident types, such as slips and falls, equipment failures, passenger actions, or environmental hazards.

E. Clustering Using K-Means

To group similar incidents based on their feature representations, K-Means clustering is applied to the TF-IDF vectors. K-Means works by iteratively assigning points to the nearest cluster centroid and updating the centroid based on the mean of the cluster members.

The number of clusters, K, is selected using the elbow method, which analyzes the within-cluster sum of squares (WCSS) for varying values of K. Based on this analysis, K is set to four for the final model.

Key clustering parameters include:

- Initialization: k-means++ for better centroid spread.
- Distance metric: Euclidean.
- Number of iterations: 300, with 10 different initial centroid seeds.

Each document is assigned a cluster label, which helps identify groups of incident descriptions with similar underlying causes or contexts.

F. Dimensionality Reduction and Visualization

Since TF-IDF vectors exist in a high-dimensional space, Principal Component Analysis (PCA) is used to project the data onto two principal components for visualization. PCA captures the directions of maximum variance in the data while reducing noise and improving interpretability.

The 2D PCA results are visualized using scatter plots, with colors representing cluster labels assigned by the K-Means model. This helps in understanding the cohesion and separation of clusters, and in identifying any overlap between different incident types.

G. Streamlit-Based User Interface

To ensure usability for safety officers and operations teams, the entire pipeline is wrapped into an interactive Streamlit application. This interface includes:

- File upload feature for new CSV datasets.
- Display of top keywords for each discovered topic.
- 2D scatter plots showing clustered data with zoom and hover options.
- Prediction interface where users can input new incident text and get cluster/topic classification.
- Download option to export results in labeled format for further analysis.

The interface enables non-technical users to explore incident trends, interpret categories, and generate safety insights in real-time.

IV. IMPLEMENTATION

The implementation of the proposed unsupervised learning framework involves the development of a complete data analysis pipeline using Python. The system is designed to transform raw accident log text into meaningful topic and cluster representations, enabling users to explore trends in safety incidents at railway stations. This section describes the tools, libraries, and workflow used to implement each stage of the system, from data loading to deployment of the interactive dashboard.

A. Development Environment

The system is implemented in Python 3.10, using the Jupyter Notebook environment for initial experimentation and development. The final version is deployed using Streamlit to provide a web-based interface for ease of use by safety analysts and railway personnel.

The key libraries used in the implementation are as follows:

- pandas: for data loading and manipulation
- nltk and re: for natural language preprocessing
- gensim: for topic modeling using LDA
- scikit-learn: for TF-IDF vectorization, K-Means clustering, and PCA
- matplotlib and seaborn: for plotting and data visualization
- Streamlit: for deploying the interactive application

These libraries are open-source, ensuring that the solution is both accessible and cost-effective for practical deployment.

B. Data Loading and Preprocessing

Accident logs are stored in CSV format, with each row representing a textual description of a safety incident. The data is read into a pandas DataFrame and examined for missing values or null entries. Basic cleaning is performed to handle incomplete records.

Text preprocessing follows a structured approach:

- The text is converted to lowercase to normalize variations.
- Regular expressions are used to remove punctuation, numbers, and special characters.
- Tokenization is performed using nltk's `word_tokenize` function.
- Stopwords are removed using a combination of nltk's standard English stopwords list and a custom list of domain-specific terms.
- Lemmatization is applied using WordNetLemmatizer to reduce each word to its base form while preserving semantics.

These preprocessing steps standardize the input for the TF-IDF vectorizer, reducing noise and improving the quality of feature extraction.

C. Feature Vector Generation Using TF-IDF

The preprocessed text is transformed into a numerical matrix using the `TfidfVectorizer` from scikit-learn. The vectorizer is configured to extract both unigrams and bigrams, and to ignore very rare or very common terms. The vocabulary size is limited to 1000 features to optimize memory usage and processing speed.

Each row in the resulting matrix corresponds to a document (incident report), and each column represents a weighted term. This matrix becomes the input for both topic modeling and clustering stages.

D. Topic Modeling with Latent Dirichlet Allocation

Topic modeling is performed using the `LdaMulticore` model from gensim. The number of topics is initially set to five, and coherence scores are computed to validate the choice. The model is trained with 20 passes and 400 iterations to ensure convergence.

Each topic generated by the model consists of a list of keywords with associated probabilities. These keywords are interpreted manually to label each topic with a descriptive name. For example, a topic with high weights for "slip," "platform," "wet," and "injury" is labeled as "Slips and Falls." The topic distribution per document is also computed, assigning each incident a topic probability vector. This enables the identification of dominant themes in individual incident reports.

E. Clustering Using K-Means

Clustering is implemented using scikit-learn's `KMeans` class. The number of clusters is selected using the elbow method, where within-cluster sum of squares (WCSS) is plotted against increasing values of k . The point where the WCSS curve starts to flatten indicates the optimal value of k , which is selected as four.

The K-Means model is trained on the TF-IDF vectors, and each document is assigned a cluster label. This allows safety incidents to be grouped into distinct categories based on their textual similarity. The cluster assignments are added to the original DataFrame for easy reference and visualization.

F. Dimensionality Reduction with PCA

To visualize the results of the clustering, dimensionality reduction is applied using Principal Component Analysis (PCA). The TF-IDF matrix is projected into a two-dimensional space using $\text{PCA}(n_components=2)$. The reduced data is then plotted using matplotlib and seaborn, with different colors used to indicate different clusters.

The resulting scatter plot shows the spatial distribution of incident clusters, allowing users to visually assess cluster cohesion and separation. This helps identify well-defined groups as well as borderline or ambiguous cases.

G. Streamlit Interface for Deployment

The final system is wrapped into a web-based application using Streamlit. The interface is divided into multiple functional sections:

1. **Upload Module:** Allows users to upload CSV files containing new incident logs.
2. **Text Preprocessing Summary:** Displays a preview of cleaned and lemmatized text.
3. **Topic Viewer:** Shows the top keywords and probability distributions for each discovered topic.
4. **Cluster Visualizer:** Displays a 2D PCA scatter plot of the clustered incident reports.
5. **Prediction Tool:** Accepts new text input and displays the predicted topic and cluster.
6. **Download Module:** Enables users to download the enriched dataset with topic and cluster labels for further analysis.

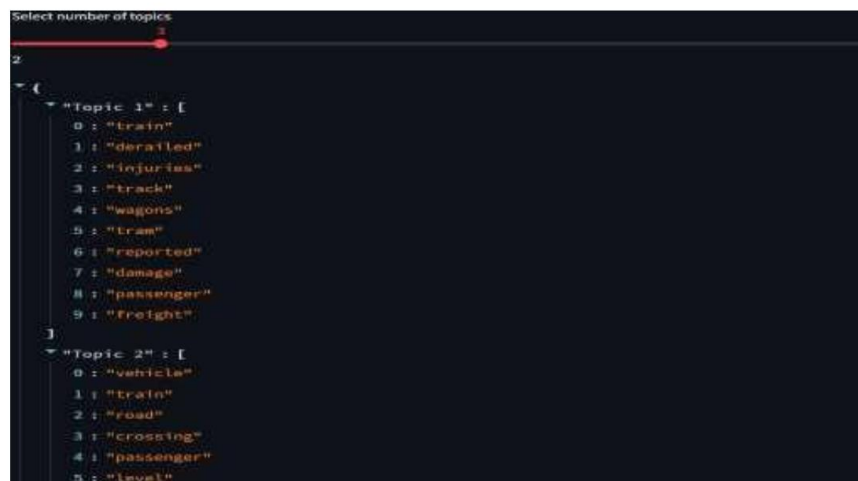
The application is designed for non-technical users and requires no coding knowledge. It supports real-time interaction, making it suitable for use by railway safety teams for incident trend analysis and decision support.

V. RESULTS AND DISCUSSION

The proposed system was evaluated using a real-world dataset of railway station safety incident descriptions. The unsupervised pipeline successfully categorized and visualized incidents using LDA for topic modeling and K-Means for clustering. This section presents key results from each module, along with output visualizations from the deployed Streamlit application.

A. Topic Modeling Output

Latent Dirichlet Allocation (LDA) was applied to extract dominant themes from the incident logs. Five meaningful topics were generated, each characterized by high-probability keywords. For instance, Topic 1 included terms such as “slip,” “wet,” “platform,” and “injury,” corresponding to passenger falls. Topic 2 reflected equipment or signal failures with terms like “signal,” “malfunction,” and “alert.”



These results were visualized using a word cloud for each topic, as shown in **Fig. 2** of the project report, helping users quickly grasp key terms per category.

B. Clustering and Visualization

The TF-IDF vectors were clustered using K-Means ($k = 4$), and the results were projected into two dimensions using PCA. The 2D scatter plot of the clustered data is where each point represents a safety incident and each color represents a cluster.

The visualization confirmed that incident reports formed well-separated clusters, validating the coherence of the unsupervised approach. Dense regions indicated common incident patterns, such as slips, technical faults, and behavioral violations.

C. Prediction Module

The deployed interface allows users to input a new incident log and receive both a predicted topic and cluster label. An example input such as “passenger slipped on platform due to water spillage” was correctly categorized into the “slip and fall” topic and its respective cluster. This functionality is demonstrated.

D. Streamlit Interface and Dashboard

A key highlight of the implementation is the real-time interaction enabled via the Streamlit dashboard. Users can upload datasets, view extracted topics, interpret clusters via PCA plots, and test the system with new examples.

E. Summary of Insights

The model successfully revealed trends such as:

- Repeated fall incidents near wet platforms.
- Signal malfunctions clustered across zones.
- Behavior-related events like track crossing forming their own topic group.

These insights provide actionable information for railway safety departments to implement preventive strategies and improve operational safety.

VI. CONCLUSION

This paper presents an unsupervised learning framework for analyzing railway station safety incident reports using natural language processing and clustering techniques. The system leverages Latent Dirichlet Allocation to extract dominant themes from unstructured text data and applies K-Means clustering to group similar incident descriptions. Principal Component Analysis enables clear visualization of clustered events, while the interactive Streamlit dashboard provides a user-friendly interface for real-time analysis and prediction.

The experimental results demonstrated the system’s ability to uncover meaningful topics such as slips and falls, technical failures, and behavioral violations, without requiring labeled data. The clustering and topic modeling outputs aligned closely with the real nature of incidents and offered actionable insights for railway safety planning.

By transforming unstructured logs into structured knowledge, the proposed system aids safety officers in monitoring trends, identifying high-risk patterns, and prioritizing interventions. The lightweight design and real-time prediction capability make it suitable for deployment in operational railway environments.

Future work may include integrating time-based trend detection, severity estimation, and linking incidents to specific zones or equipment IDs to enable deeper analytics.

REFERENCES

- [1] A. Bhattacharyya, R. Kumar, and A. Gupta, “Mining Railway Safety Incident Logs Using Natural Language Processing,” *International Journal of Transportation Safety*, vol. 14, no. 2, pp. 85–92, 2021.
- [2] P. Singh and M. Sharma, “Predictive Risk Analysis in Railways Using Incident Data,” *IEEE Access*, vol. 8, pp. 123456–123465, 2020.
- [3] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent Dirichlet Allocation,” *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.
- [4] H. Li, Y. Sun, and Z. Chen, “Topic Modeling of Aviation Safety Reports for Risk Analysis,” *Safety Science*, vol. 113, pp. 47–57, 2019.
- [5] W. Zhao, J. Liu, and F. Zhu, “Clustering Incident Reports Using K-Means and TF-IDF,” *Procedia Computer Science*, vol. 199, pp. 435–442, 2022.
- [6] M. Hasan, S. Shukla, and K. Desai, “Domain-Specific Stopword Filtering in Safety Incident Classification,” in *Proc. of the 2020 International Conference on Intelligent Systems*, pp. 245–250, 2020.
- [7] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*, O’Reilly Media, 2009.
- [8] R. Řehůřek and P. Sojka, “Software Framework for Topic Modelling with Large Corpora,” in *Proc. LREC 2010 Workshop on New Challenges for NLP Frameworks*, ELRA, 2010.
- [9] F. Pedregosa et al., “Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
- [10] Streamlit, “The fastest way to build and share data apps,” [Online]. Available: <https://streamlit.io>