Journal of **N**onlinear **A**nalysis and **O**ptimization

Vol. 14, Issue. 2 : 2023

ISSN : **1906-9685**

CONSTRUCTION SITE ACCIDENT ANALYSIS USING TEXT MINING AND NATURAL LANGUAGE PROCESSING TECHNIQUES

[1]P.Naga Lakshmi, [2]Akkireddy Adithi, [3]A. Jahnavi Reddy, [4]Amruth Deepthi, [5]B. Saanvi Reddy

[1]Assitsant Professor, [2,3,4,5] UG Students, Dept. Computer Science and Engineering-Data Science, Mallareddy Engineering college for Women, Hyderabad, India

ABSTRACT

Workplace safety is a major concern in many countries. Among various industries, the construction sector is identified as the most hazardous workplace. Construction accidents not only cause human sufferings but also result in huge financial loss. To prevent recurrence of similar accidents in the future and make scientific risk control plans, analysis of accidents is essential.

In the construction industry, fatality and catastrophe investigation summary reports are available for past accidents. In this study, text mining and natural language process (NLP) techniques are applied to analyze construction accident reports. To be more specific, five baseline models, support vector machine (SVM), linear regression (LR), K-nearest neighbor (KNN), decision tree (DT), Naive Bayes (NB) and an ensemble model are proposed to classify the causes of the accidents. Besides, Sequential Quadratic Programming (SQP) algorithm is used to perfect the weight of each classifier involved in the ensemble model.

EXperiment results show that the optimized ensemble model outperforms the rest models considered in this study in terms of average weighted F1 score. The result also shows that the proposed approach is more robust to cases of low support.

INTRODUCTION

The aim is to enhance safety measures and minimize the occurrence of accidents. By analyzing textual data such as accident reports, incident narratives, and safety records, valuable insights can be gained regarding the causes, patterns, and trends of accidents. This information can then be used to identify potential hazards, develop proactive safety strategies, and implement preventive measures to mitigate risks. Ultimately, the project seeks to create a safer work environment for construction workers and reduce the occurrence of accidents, leading to improved safety outcomes and enhanced overall construction site management.

Objective

By analyzing the textual data and applying NLP techniques, the project aims to identify the root causes of construction site accidents. This involves examining the accident reports, incident narratives, and safety records to uncover the contributing factors, such as human error, equipment malfunction, inadequate training, or

http://doi.org/10.36893/JNAO.2023.V14I2.0149-0160

insufficient safety protocols. By understanding the underlying causes, the project can provide valuable insights into areas that require improvement and help stakeholders implement effective preventive measures. The findings can be used to enhance safety training programs, revise safety guidelines, and implement risk mitigation strategies. Ultimately, the objective is to reduce the occurrence of construction site accidents by addressing the root causes and creating a safer work environment for construction workers.

LITERATURE SURVEY

It revealed the significance of analyzing accidents, the application of text mining and NLP in accident analysis, methodologies for accident categorization and severity assessment, root cause analysis, data collection and preprocessing techniques, relevant tools and technologies, and future research directions.

Existing System

Construction site accident analysis, traditional manual methods are commonly used, which involve the labor-intensive and time-consuming task of reviewing accident reports, incident narratives, and safety records manually. These documents are often in unstructured text format, making it challenging to extract valuable information and identify patterns or trends.

The lack of automation in the existing system poses limitations in terms of efficiency, accuracy, and scalability. It hinders the ability to handle large volumes of textual data and may lead to inconsistencies or oversights in accident analysis. Additionally, manual analysis may be subjective and prone to human errors, making it difficult to ensure consistent and reliable results.

Furthermore, the existing system may face challenges in identifying underlying causes and performing comprehensive root cause analysis due to the complexity of textual data. Lack of standardized categorization and severity assessment methods may also limit the system's ability to derive meaningful insights from the textual data.

Overall, the existing system relies heavily on manual effort, which can be inefficient, error-prone, and limited in terms of its ability to extract valuable insights from textual data. Therefore, there is a need for an automated and intelligent system that leverages text mining and natural language processing techniques to overcome these limitations and enhance the accuracy, efficiency, and effectiveness of construction site accident analysis.

Proposed System

This will provide safety to workers at construction site from accidents by analyzing past accident data by using machine learning algorithms and text mining technique such as TF-IDF (Term Frequency-Inverse Document Frequency) and natural language text processing to remove special symbols, stop words, stemming etc.

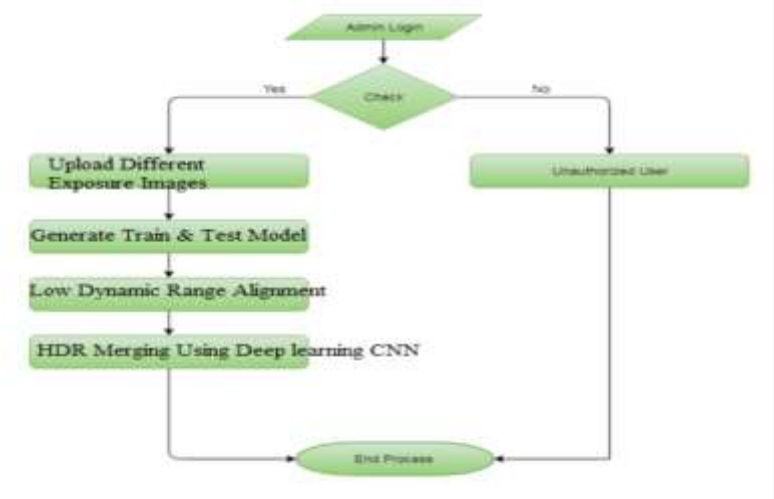
By automating data collection, preprocessing, and analysis, the system significantly reduces manual effort and time required for reviewing large volumes of accident reports. Through text mining and information extraction techniques, the system extracts relevant information such as accident types, locations, dates, severity, and causes from the reports.

Advantages of Proposed System

- Enhanced Safety Measures
- Efficient Data Processing
- Accurate Accident Categorization

Data Flow Diagram

http://doi.org/10.36893/JNAO.2023.V14I2.0149-0160



METHODOLOGY

1. **Data Collection:** Gather enough accident-related textual data, such as accident reports, incident narratives, safety records, or any other relevant documents. Ensure the data is representative of several types of accidents and covers a considerable time.
2. **Data Preprocessing:** Clean and preprocess the collected textual data to prepare it for analysis. This step may involve removing irrelevant information, standardizing formats, handling spelling errors, removing stop words, and performing other necessary text preprocessing techniques.
3. **Text Mining and NLP Analysis**: Apply text mining and NLP techniques to extract useful information from preprocessed text data. This can involve tasks like text classification, sentiment analysis, named entity recognition, topic modeling.
4. **Accident Categorization**: Categorize the accidents based on the extracted information and analysis results. Develop a classification scheme or use existing standards to classify accidents into various categories such as types of accidents, causes, severity levels, or other relevant factors.
5. **Root Cause Analysis**: Perform root cause analysis to identify the underlying causes and contributing factors of accidents. Analyze the patterns, correlations, and relationships among the categorized accidents to uncover common factors or trends.
6. **Insights and Visualization**: Generate insights and visualizations based on the analysis results. Present the findings in a clear and understandable manner using charts, graphs, or other visual representations.
7. **Preventive Measures**: Based on the identified root causes and insights from the analysis, suggest preventive measures and safety improvements to minimize the occurrence of accidents.
1. **Evaluation and Validation**: Evaluate the effectiveness of the methodology and the results obtained. Validate the findings with domain experts or compare them with existing accident data or expert knowledge to ensure the accuracy and reliability of the analysis.

IMPLEMENTATION

We have implemented ML Algorithms and NLP Techniques in this project. There are different types of ML algorithms, we used various Supervised Algorithms and used various NLP techniques.

Algorithms

Logistic Regression

Logistic regression is a statistical model used for binary classification problems, where the goal is to predict the probability of an instance belonging to a particular class. Despite its name, logistic regression is a classification algorithm rather than a regression algorithm.

Formulas:

http://doi.org/10.36893/JNAO.2023.V14I2.0149-0160

The logistic regression model uses the logistic function (also known as the sigmoid function) to model the relationship between the independent variables and the probability of the outcome. The formula for logistic regression is:

$p = 1 / (1 + e^{(-z)})$

Where:

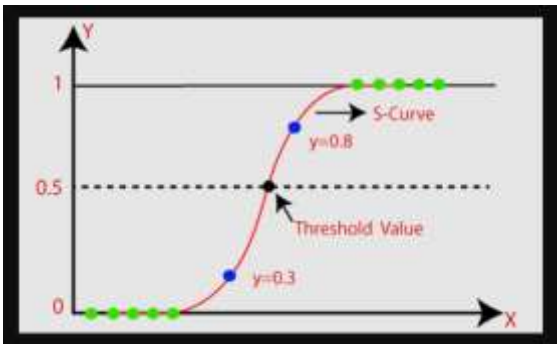
p represents the probability of the outcome.

z represents the linear combination of the independent variables and their coefficients, given by:

$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_p x_p$

$\beta_0, \beta_1, \beta_2, ..., \beta_p$ represent the coefficients (intercept and slopes) for the independent variables.

$x_1, x_2, ..., x_p$ represent the independent variables.



Naive Bayes Classifier

Naive Bayes is a probabilistic classification algorithm based on Bayes' theorem with the assumption of independence among the features. It is widely used for text classification and is known for its simplicity and efficiency.

Formulas:

Naive Bayes applies Bayes' theorem to calculate the posterior probability of a class given the observed features. The formula for Naive Bayes is:

$P(y|x_1, x_2, ..., x_n) = (P(x_1|y) * P(x_2|y) * ... * P(x_n|y) * P(y)) / P(x_1, x_2, ..., x_n)$

Where:

$P(y|x_1, x_2, ..., x_n)$ is the posterior probability of class y given the observed features.

$P(x_1|y), P(x_2|y), ..., P(x_n|y)$ are the conditional probabilities of the features given class y.

P(y) is the prior probability of class y.

$P(x_1, x_2, ..., x_n)$ is the evidence or marginal probability of the observed features.

Decision Tree

A decision tree is a versatile and widely used machine learning algorithm that can be used for both regression and classification tasks. It creates a tree-like model of decisions and their potential consequences based on the features of the data.

Information Gain, Entropy, and Gini Index are commonly used measures to evaluate the quality of splits in decision trees and determine the best attribute to split the data on.

Ensemble Learning Techniques

Bagging (Bootstrap Aggregating)

Bagging is an ensemble learning technique that aims to improve the stability and accuracy of predictions by combining multiple models trained on different subsets of the training data. The process involves the following steps:

http://doi.org/10.36893/JNAO.2023.V14I2.0149-0160

1. **Bootstrap Sampling:** Random subsets of the training data are created through bootstrap sampling, where instances are randomly selected with replacement. Each subset has the same size as the original training set.
2. **Model Training:** For each bootstrap sample, a separate model is trained using the same learning algorithm. Each model learns from a different subset of the data, introducing diversity among the models.
3. **Prediction Aggregation:** The final prediction is made by aggregating the predictions of all the individual models. In classification tasks, the majority vote among the models is taken as the final prediction. In regression tasks, the predictions are averaged.

Boosting

Boosting is an ensemble learning technique that focuses on reducing bias and improving the accuracy of predictions. The key steps in boosting are as follows:

1. **Weight Assignment:** Each instance in the training data is assigned an initial weight.
2. **Model Training and Weight Update:** Models are trained iteratively, giving more importance to instances that were misclassified in previous iterations. Each model aims to minimize the error made by the previous models.
3. **Prediction Combination:** The final prediction is made by combining the predictions of all the models, typically using a weighted voting scheme. The weights are determined based on the performance of each model during training.

Stacking (Stacked Generalization)

Stacking is an ensemble learning technique that combines predictions from multiple models using a meta-model. The process involves the following steps:

- **Base Model Training:** Several diverse base models are trained on the training data, each using a different learning algorithm or configuration.
- **Prediction Generation:** The base models make predictions on the training data, which are then collected as additional features.
- **Meta-Model Training:** A meta-model, also known as a blender or meta-learner, is trained on the predictions made by the base models. The meta-model learns to combine the predictions effectively.
- **Final Prediction:** The final prediction is made using the trained meta-model, which takes the predictions of the base models as input.

K Nearest Neighbors (KNN)

k-Nearest Neighbors (k-NN) is a simple and intuitive algorithm used for both regression and classification tasks in machine learning. It is a non-parametric and instance-based algorithm, meaning it does not make assumptions about the underlying data distribution and instead relies on the proximity of instances in the feature space.

k-Nearest Neighbors (k-NN) is often referred to as a "lazy" algorithm because it does not explicitly build a model during the training phase.

Formulas:

Distance Calculation: The distance between two instances, x and y, can be calculated using various distance metrics. The most common distance metrics used in k-NN are Euclidean distance and Manhattan distance.

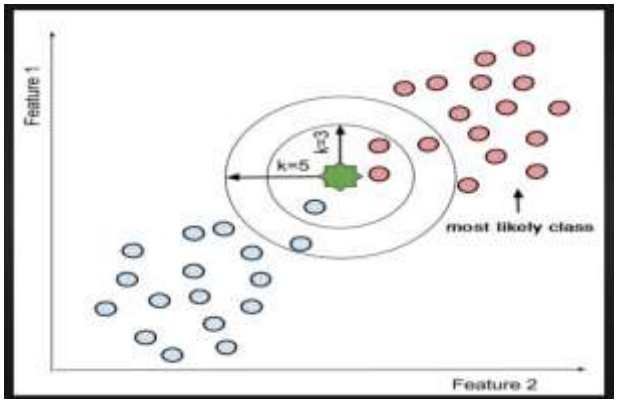
• Euclidean distance: d (x, y) = sqrt (sum ((xi – yi)2))

• Manhattan distance: d (x, y) = sum (|xi – yi|)

Prediction (Regression): For regression tasks, the predicted value for a new instance is calculated as the average of the target values of its k nearest neighbors.

Prediction (Classification): For classification tasks, the predicted class for a new instance is determined by the majority vote among its k nearest neighbors.

http://doi.org/10.36893/JNAO.2023.V14I2.0149-0160



K-nearest neighbors (KNN) is a simple and popular machine learning algorithm used for both classification and regression tasks.

Support Vector Machine (SVM)

Support Vector Machines (SVM) is a supervised machine learning algorithm that aims to find an optimal hyperplane in a high-dimensional feature space to separate instances of different classes. It accomplishes this by identifying a subset of training data points, called support vectors, which are crucial in defining the decision boundary.

Formulas:

Linear SVM: The decision function for linear SVM can be represented as: $f(x) = w^T x + b$

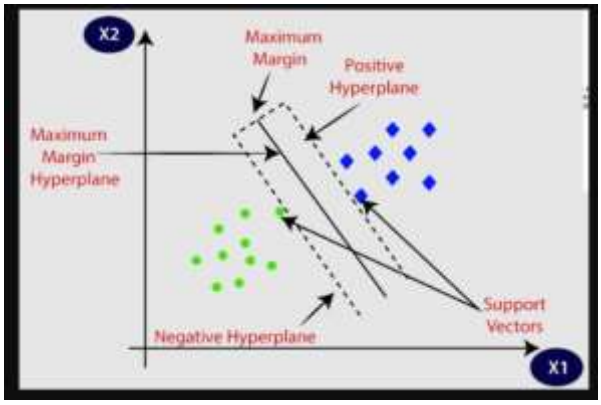
where:

• f(x) represents the decision function.
• x is the input vector.
• w is the weight vector.
• b is the bias term.

Non-linear SVM (Kernel trick): The decision function for non-linear SVM using the kernel trick can be represented as: $f(x) = \Sigma \alpha_i y_i K(x_i, x) + b$

where:

• f(x) represents the decision function.
• x is the input vector.
• x_i is a support vector.
• α_i is the corresponding Lagrange multiplier.
• y_i is the class label of the support vector.
• $K(x_i, x)$ is the kernel function that measures the similarity between x_i and x.
• b is the bias term.

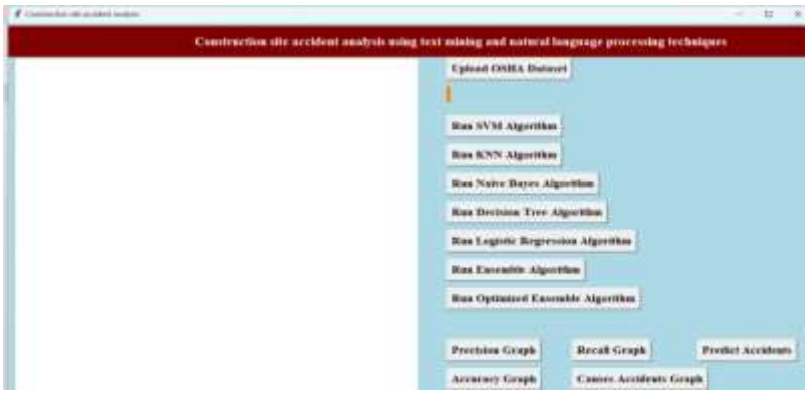


http://doi.org/10.36893/JNAO.2023.V14I2.0149-0160

TEST RESULTS

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product. There are several types of tests. Each test type addresses a specific testing requirement.

RESULTS



In above screen click on 'Upload OSHA Dataset' button and upload dataset

In above screen I am uploading 'OSHA.csv' dataset and after uploading dataset will get below screen

In the above screen we can see dataset contains total 599 records and all records contain total 3934 words or features for vector. Now click on 'Run SVM Algorithm' button to build SVM model on uploaded dataset and

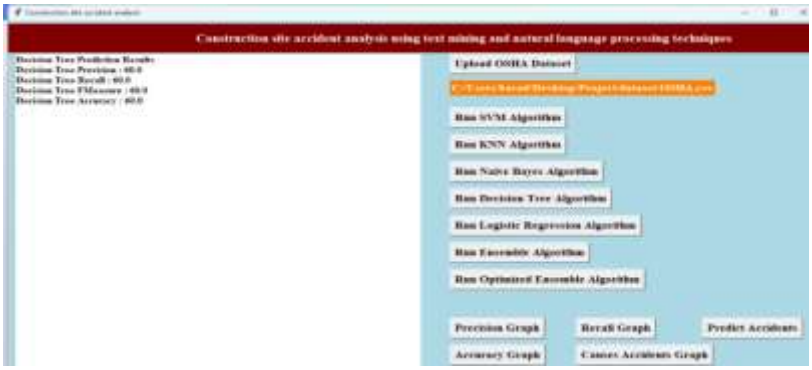
calculate its Prediction accuracy, precision etc.

In above screen we got SVM prediction score as 70% and now click on 'Run KNN Algorithm' button to get its prediction accuracy

In above screen for KNN we got 55% prediction accuracy and now click on 'Run Naïve Bayes Algorithm' button to get its accuracy

In above screen Naïve Bayes gave 50% accuracy and now click on 'Run Decision Tree Algorithm' button to get its accuracy

http://doi.org/10.36893/JNAO.2023.V14I2.0149-0160



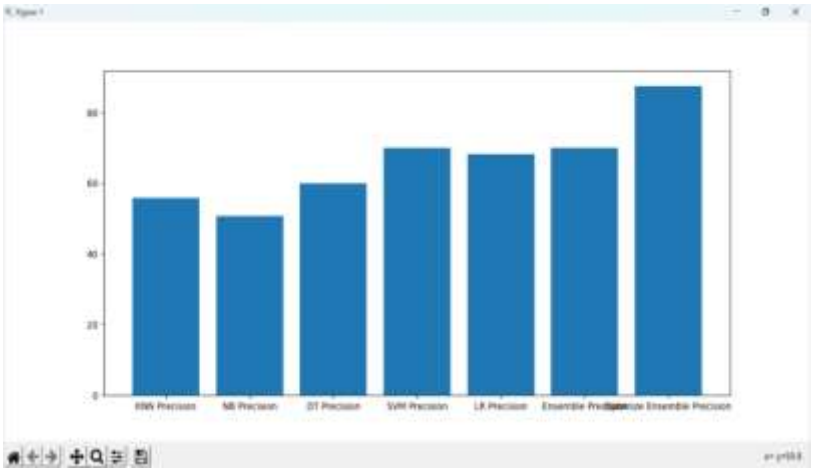
In above screen we got decision tree accuracy as 56% and now click on 'Run Logistic Regression Algorithm' button to get it accuracy

Similarly run ensemble algorithm

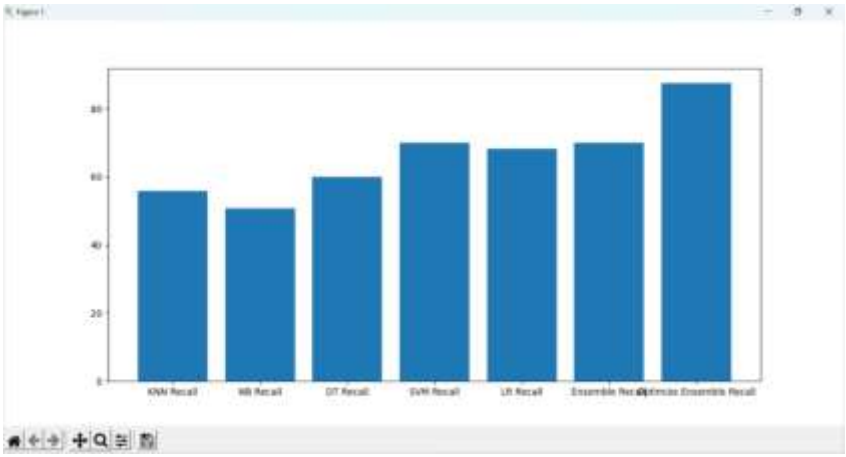
Similarly run 'Optimized Ensemble Algorithm' button to get accuracy of voting classifier. Optimize ensemble algorithm is also called as Voting Classifier

http://doi.org/10.36893/JNAO.2023.V14I2.0149-0160

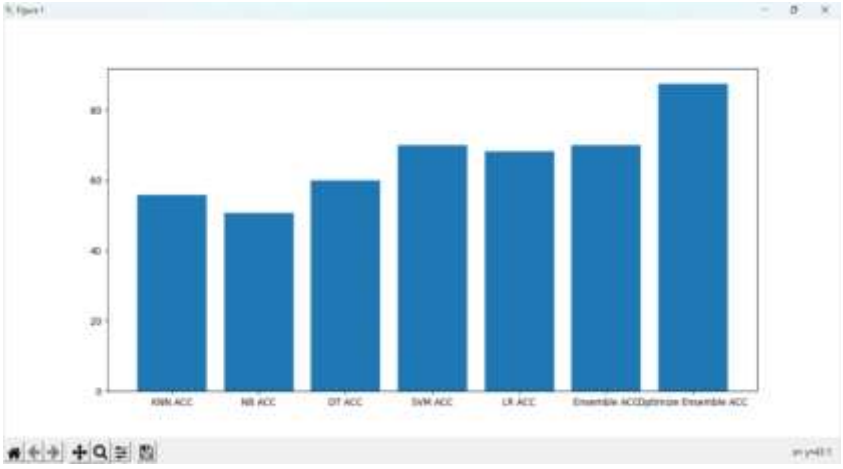
In above screen with Voting Classifier we got 86% accuracy. Now click on 'Precision Graph' button to view precision Comparison between all algorithms



In above graph x-axis represents algorithm names and y-axis represents precision of those algorithms. In above graph we can see Propose Optimize Ensemble (Voting Classifier) gave better performance. Now click on 'Recall Graph' button to get below recall comparison graph in all algorithms

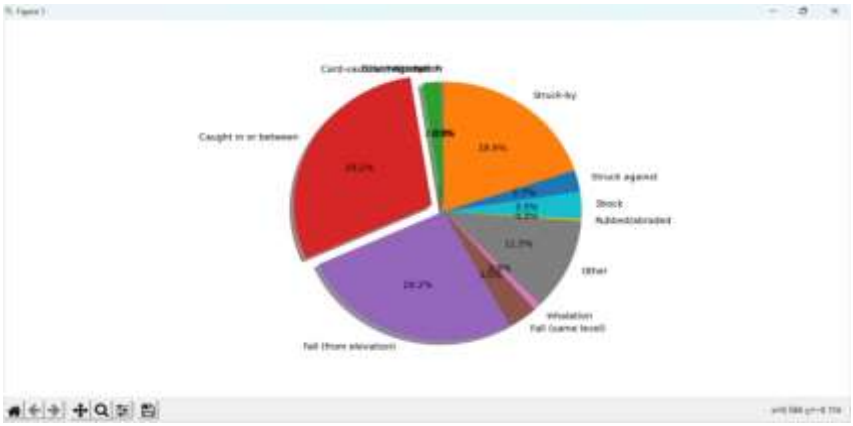


Accuracy graph comparison in all algorithms

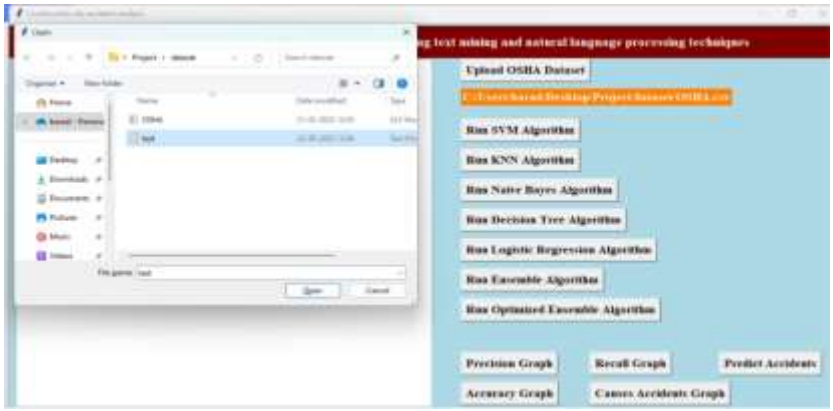


http://doi.org/10.36893/JNAO.2023.V14I2.0149-0160

In the above accuracy graph x-axis represents algorithm name and y-axis represents accuracy of those algorithms. Now click on 'Causes Accident Graphs' button to see several types of accidents causes in this dataset



Now click on 'Predict Accidents' button to upload new work as test data and predict future accidents can occur in this work



In above screen I am uploading 'test.txt' which contains new work and application predict accidents which may occur in this work

CONCLUSION

We proposed an approach to automatically extract valid accident precursors from a dataset of raw construction injury reports. Such information is highly valuable, as it can be used to better understand, predict, and prevent injury occurrence. For each of three supervised models (two of which being deep learning-based), we provided a methodology to identify (after training) the textual patterns that are, on average, the most predictive of each

safety outcome. We verified that the learned precursors are valid and made several suggestions to improve the results. The proposed methods can also be used by the user to visualize and understand the models' predictions. Incidentally, while predictive skill is high for all models, we make the interesting observation that the simple TF-IDF + SVM approach is on par with (or outperforms) deep learning most of the time**.**

FUTURE SCOPE

It holds great potential for further advancements and applications. By expanding the dataset, incorporating real-time monitoring and alerting systems, and delving into causal analysis and root cause identification, deeper insights into construction site accidents can be gained. Additionally, leveraging predictive analytics to forecast accident likelihood and implementing benchmarking and comparative analysis across different projects or regions can drive continuous improvement in safety measures. Integrating visualization techniques with the text mining and NLP results can enhance the understanding and presentation of the findings, making them more accessible to stakeholders and facilitating data-driven decision-making. As technology continues to evolve, combining text mining, NLP, and other emerging technologies such as machine learning, deep learning, or natural language understanding can unlock further opportunities in accident analysis, prevention, and safety management within the construction industry.

REFERENCES

[1] D. Reinsel, J. Gantz, J. Rydning, Data age 2025. The digitization of the world from edge to core, Tech. rep., Accessed 21st January 2020 (2018).

[2] S. Grimes, Unstructured data and the 80 percent rule, Accessed 21st January 2020 (2008). URLhttp://breakthroughanalysis.com/2008/08/01/ unstructured-data-and-the-80-percent-rule/

[3] D. D. Woods, E. S. Patterson, E. M. Roth, Can we ever escape from data overload? A cognitive systems diagnosis, Cognition, Technology & Work 4 (1) (2002) 22–36. doi:10.1007/ s101110200002.

[4] N. Henke, J. Bughin, M. Chui, J. Manyika, T. Saleh, B. Wiseman, G. Sethupathy, The age of analytics: competing in a data-driven world, Tech. rep., Accessed 21st January 2020 (2016).

[5] D. Lukic, A. Littlejohn, A. Margaryan, A framework for learning from incidents in the workplace, Safety Science 50 (4) (2012) 950–957. doi:10.1016/j.ssci.2011.12.032.

[6] J. M. Sanne, Incident reporting or storytelling? Competing schemes in a safety-critical and hazardous work setting, Safety Science 46 (8) (2008) 1205–1222. doi:10.1016/j.ssci. 2007.06.024.

[7] W. J. Wiatrowski, J. A. Janocha, Comparing fatal work injuries in the united states and the european union, Tech. rep., Accessed 21st January 2020 (June 2014). URL https://www.bls.gov/opub/mlr/2014/article/ comparing-fatal-work-injuries-us-eu.htm

[8] I. Goodfellow, Y. Bengio, A. Courville, Deep learning, MIT Press, 2016. URL http://www.deeplearningbook.org.