Accident Analysis on Construction Sites Using Data Mining and Natural Language Processing

G.Vihari[1], V.Swathi[2] , Vurla Veeraju[3] , Ajaykumar Dharmireddy[4]

[1]Department of information technology, Sir C.R.Reddy College of Engineering, Eluru, Andhra Pradesh, India

[2-4]Department of ECE, Sir C.R.Reddy College of Engineering, Eluru, Andhra Pradesh, India

Abstract:- The issue of worker protection is of paramount importance in many nations. The construction industry is singled out as the most dangerous to work in. In addition to the human toll, construction accidents may have a devastating economic impact. Analysis of accidents is crucial for preventing such mishaps in the future and developing sound risk management strategies. Summary reports of fatality and disaster investigations are available for previous incidents in the construction sector. The information on construction industry accidents examines here using text mining and natural language processing (NLP) methods. To categories the origins of the incidents, the support vector machine (SVM), linear regression (LR), K-nearest neighbour (KNN), decision tree (DT), Naive Bayes (NB), and an ensemble model. In addition, the weight of each classifier in the ensemble model optimizes using the sequential-quadratic programming (SQP) technique. The optimized ensemble model achieved a higher weighted F1 score on average than the other models tested in this research. The outcome demonstrates that the suggested method is more resilient in low-support scenarios. In addition, we present an unsupervised chunking method to extract accident-causing everyday items using grammatical rules found in the reports. Foreign objects often cause construction accidents, so locating and removing them is crucial. We address the suggested approaches' shortcomings and provide recommendations for moving further.

Keywords*: Natural language processing (NLP), construction accidents, linear regression (LR), Decision tree (DT), K-nearest neighbour (KNN) , Sequential-quadratic programming (SQP).*

1. Introduction

The problem involves utilizing past accident data to develop machine learning algorithms to effectively analyze the data and identify the root causes of accidents [1]. We can prevent future accidents by providing new work test data to predict potential causes of accidents. Machine learning algorithms have the potential to aid in the identification and extraction of hazardous objects, including but not limited to misused tools, sharp objects nearby, and damaged equipment [2]. Developing a precise visual system capable of real-time classification and analysis of fruits is crucial for optimizing harvesting robots in terms of cost-effectiveness and efficiency. Nevertheless, the pragmatic achievements in this domain remain restricted. As far as our comprehension extends, no scholarly inquiry exists concerning the implementation of machine vision for date fruits within an orchard milieu. This study presents a proficient framework for machine vision that robots can use for harvesting date fruits. The framework comprises three classification models for real-time classification of date fruit images based on their type, maturity, and the harvesting decision [3]. The classification models employ deep convolutional neural networks incorporating transfer learning and fine-tuning techniques on pre-existing models [4]. They generated a comprehensive dataset of images depicting date fruit clusters in an orchard [5]. This dataset comprises over 8000 photos of five different date varieties at varying stages of pre-maturity and maturity

2. Types of Testing

Unit testing: The goal of unit testing is to ensure that a programmeme behaves as intended and that the inputs produce the desired outputs by generating test cases that exercise the programmeme's individual components [6]. It is crucial to verify the internal code flow and all decision branches. The functionality of an application is tested in increments. It is carried out after the completion of individual parts but before they are combined. In order to do such invasive tests, detailed information on the building's construction is required [7]. Unit tests are used to verify the functionality of a specific business procedure, software, or hardware configuration [8]. Each alternative path through a business process may be evaluated separately to ensure it satisfies standards and yields the desired results.

Integration testing: Integration tests are run to make sure that all of an app's components mesh well with one another. For each test case, we only care about the simplest possible outcome for the given screen or field [9]. Even if each component was OK on its own, as shown by passing unit tests, integration tests verify that the entire system is right and consistent. To find any problems that may arise when parts are put together, integration testing is essential.

Functional test: Functional testing is a methodical way to prove that the system works as intended, in accordance with the business and technical specifications, the system documentation, and the user guides.

Systems/Procedures: Invoking a system or method that serves as an interface is required [10]. Requirements, important functionalities, and unique test cases are the focal points of functional testing planning and execution. Additionally, testing must take into account systematic coverage of key business process flows, data fields, specified procedures, and subsequent activities [11]. The effectiveness of existing tests and the identification of new tests are done before functional testing is considered complete.

System Test: System testing verifies that the integrated software system as a whole fits the criteria. It validates a setup to provide consistent and predictable outcomes [12]. The configuration-oriented system integration test is a form of system testing. System testing relies on process descriptions and flows, with an emphasis on pre-driven process connections and integration points.

White Box Testing: When a software tester understands the programme's design, structure, and language, they are said to be doing "white box sting." Indeed, this is the meaning of life. It's used for scenarios that can't be tested in a "black box.".

Black Box Testing: During black box testing, testers pretend to know nothing about the code's internals, architecture, or language. Black box testing, like other methods of testing, requires a written document defining the system to be tested, such as a specification or requirements document [13]. Testing in which the goal plan is completely disregarded. You can't see what's within. The test simply provides inputs and responds to outputs without taking into consideration how the application really works.

3. Methodology

Information systems in businesses can benefit from graphical representations of data flows known as dataflow diagrams. DFD describes the processes that are involved in a system to transport data from the input to the files to generate reports. It is possible to classify data flow diagrams as logical or physical [14]. A logical data flow diagram is a visual representation of the flow of data needed to carry out a certain business function. The diagram of the physical data flow is a representation of the logical data flow's actual execution [15]. In DFD, the processes that take in, transform, store, and disperse information between a system, its surroundings, and its constituent parts are graphically represented. The visual aspect makes it a useful medium for conveying information between the user and the system designer.

. Proposed design

The author of this paper presents a concept aimed at enhancing the safety of workers at construction sites by utilizing machine learning algorithms and text mining techniques, such as TF-IDF (Term Frequency-Inverse Document Frequency) and natural language text processing [16]. The approach involves analyzing past accident data and removing special symbols, stop words, and stemming, among other techniques.

One of the main advantages of the proposed system is the utilization of a data set, which facilitates easy harvesting of data.

System Architecture Module description: This study delves into the analysis of construction site accidents by applying text mining and natural language processing techniques [17]. The methodology involves eliminating stop words, punctuations, and special symbols and using stemming techniques to cleanse historical accident data. Following the data cleaning process, the subsequent step involves converting all textual data into a numeric vector through the utilisation of the TF-IDF technique [18]. The TF-IDF approach consists in assigning a weight to each word in a vector based on its frequency and subsequently utilising this vector to construct a machine-learning training model [19]. Upon the provision of new test data, the data is subjected to the TF-IDF conversion process, which is applied to the trained model to facilitate the search for comparable data and subsequently generate predictions based on the identified similarities. The following example outlines the process of converting text into a TF-IDF vector. Suppose I have 3sentences.According to popular belief, consuming an apple daily prevent the need for medical attention. Apples are beneficial for one's health. Initially, the elimination of stop words, such as "an," "a," "of," and "in," is executed in sentences. Subsequently, the remaining words are utilise to construct vector columns [20]. Once the columns establish, assign each word count as the value of its respective vector. Please refer to the vector columns below.

System Specification software requirements: The functional requirements for a secure cloud storage service are straightforward and uncomplicated.

The service's functionality should include the capability to retain and manage the user's data.

The data should be readily available via any internet-connected device.

The service must be able to synchronies the user's data across devices such as notebooks, smart phones, and other devices.The services must record all past modifications, commonly called versioning.

Unified Modeling Language, or UML, is an abbreviation. UML, or the Unified Modelling Language, is a relatively new method of describing and modeling software[21]. One of the most commonly used ways to model business processes.It relies on graphical representations of programmes to aid development. The adage "a picture is worth a thousand words" is certainly true in this case. Problems in code or company procedures may be more easily seen through the use of diagrams. The disarray around software development and documentation inspired the creation of UML [22]. Tare systems were represented and documented in a variety of ways throughout the 1990s. In response to the necessity for a standard notation to depict such systems, three software engineers at Rational Software created the Unified Modelling Language (UML) between 1994 and 1996 [23]. It eventually became the standard in 1997 and has stayed so with just minor revisions since then.

4. Results and Discussions

Finding defects is the objective of the testing process. Testing is done to locate and fix any and all issues that may be associated with a product. It helps check the functionality of individual components, whole assemblies, and even finished products. It is the process of exercising software to verify that the software system satisfies its requirements and user expectations and does not fail in an undesirable way. There are a few distinct kinds of tests to choose from. Because there are many different types of testing demands, there are many different kinds of tests.

Unit Testing: While it is normal practice to conduct coding and unit testing as two separate stages of the software development lifecycle, unit testing is often performed as part of a combined code and unit test phase.

Test strategy and approach: Functional tests will be prepared in great depth, and manual testing in the field will be undertaken.

The goals of the test are to ensure that all fields accept valid input. The specified link must be used to access the desired page. The login page, messages, and answers must all be instant

Indicators to be evaluated: Check to verify that the entries have the appropriate formatting. It is not reasonable to accept several submissions of the same thing. There shouldn't be any broken connections.

The practise of methodically analysing two or more interdependent software components on a single platform for the existence of interface issues is referred to as software integration testing, and it is also known by its alternate name, incremental integration testing. An integration test is performed with the intention of ensuring that two or more software components (such as those that make up a software system or, more broadly, software applications at the corporate level) operate together without experiencing any glitches.

Every single one of the test cases described above was finished off with flying colours. There were not even the slightest of difficulties.

This is the user acceptance testing phase. Given the importance of testing to the completion of any given project, it is imperative that users have a significant role in its execution. It also guarantees that the system satisfies the functional requirements. Every single one of the test cases described above was finished off with flying colours. We could not locate any errors. This is seen in Figure 1(a).Simply choose the dataset you want to upload by selecting the "Upload OSHA Dataset" button. Example 1.(a) Displays the process of uploading the 'OSHA.csv' dataset, as well as the results of uploading the dataset.

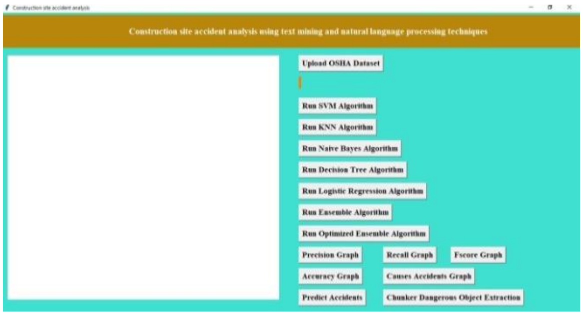

Construction site accident analysis
Construction site accident analysis using text mining and natural language processing techniques
Upload OSHA Dataset
Run SVM Algorithm
Run KNN Algorithm
Run Naive Bayes Algorithm
Run Decision Tree Algorithm
Run Logistic Regression Algorithm
Run Ensemble Algorithm
Run Optimized Ensemble Algorithm
Precision Graph
Recall Graph
Fscore Graph
Accuracy Graph
Causes Accidents Graph
Predict Accidents
Chunker Dangerous Object Extraction

Fig. 1(a). click on 'Upload OSHA Dataset' button and upload dataset.

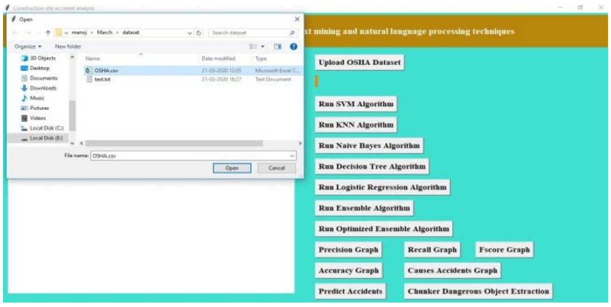
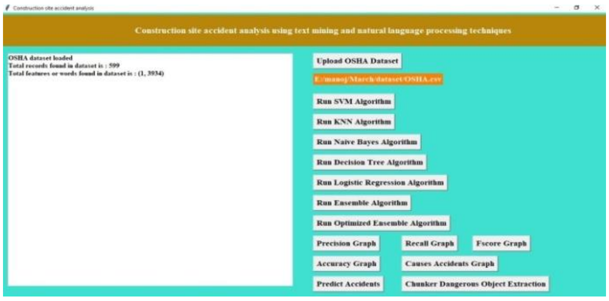


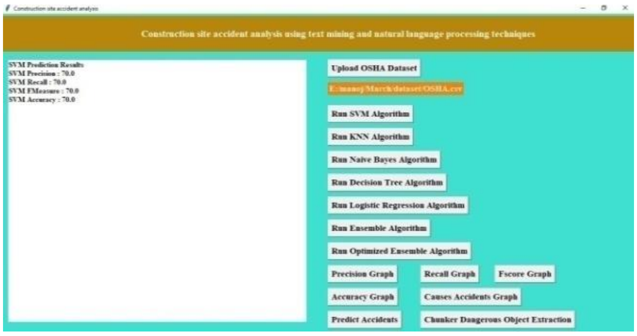
Fig 1(b). Uploading 'OSHA.csv' dataset and after uploading dataset will get below screen

Figure.2(a) depicts we can see dataset contains total 599 records and all records contains total 3934 word for features for vector. Now click on 'Run SVM Algorithm' button to build SVM model on uuploaded dataset and calculate its prediction accuracy, precision etc


```
OSHA dataset loaded
Total records found in dataset is : 599
Total features or words found in dataset is : (1, 3934)
```

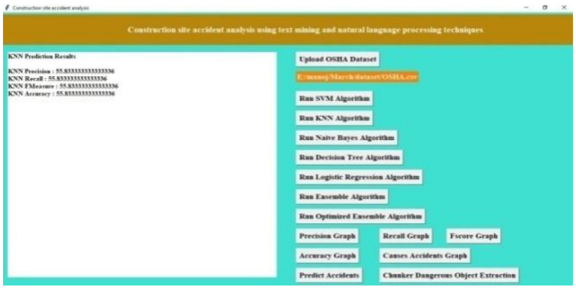
Construction site accident analysis
Construction site accident analysis using text mining and natural language processing techniques
Upload OSHA Dataset
E:/manoj/March/dataset/OSHA.csv
Run SVM Algorithm
Run KNN Algorithm
Run Naive Bayes Algorithm
Run Decision Tree Algorithm
Run Logistic Regression Algorithm
Run Ensemble Algorithm
Run Optimized Ensemble Algorithm
Precision Graph
Recall Graph
Fscore Graph
Accuracy Graph
Causes Accidents Graph
Predict Accidents
Chunker Dangerous Object Extraction

Fig. 2.(a) 'Run SVM Algorithm' button to build SVM model on uuploaded dataset and calculate its prediction accuracy, precision .


Construction site accident analysis
Construction site accident analysis using text mining and natural language processing techniques
Upload OSHA Dataset
E:/manoj/March/dataset/OSHA.csv
Run SVM Algorithm
Run KNN Algorithm
Run Naive Bayes Algorithm
Run Decision Tree Algorithm
Run Logistic Regression Algorithm
Run Ensemble Algorithm
Run Optimized Ensemble Algorithm
Precision Graph
Recall Graph
Fscore Graph
Accuracy Graph
Causes Accidents Graph
Predict Accidents
Chunker Dangerous Object Extraction

```
SVM Prediction Results
SVM Precision : 70.0
SVM Recall : 70.0
SVM FMeasure : 70.0
SVM Accuracy : 70.0
```

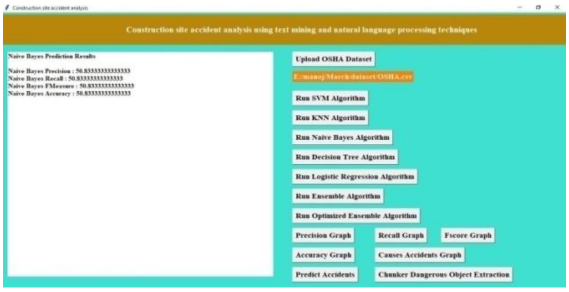
Fig.2(b). SVM prediction score as 70%


Construction site accident analysis
Construction site accident analysis using text mining and natural language processing techniques
Upload OSHA Dataset
E:/manoj/March/dataset/OSHA.csv
Run SVM Algorithm
Run KNN Algorithm
Run Naive Bayes Algorithm
Run Decision Tree Algorithm
Run Logistic Regression Algorithm
Run Ensemble Algorithm
Run Optimized Ensemble Algorithm
Precision Graph
Recall Graph
Fscore Graph
Accuracy Graph
Causes Accidents Graph
Predict Accidents
Chunker Dangerous Object Extraction

```
KNN Prediction Results

KNN Precision : 55.833333333333336
KNN Recall : 55.833333333333336
KNN FMeasure : 55.833333333333336
KNN Accuracy : 55.833333333333336
```

Fig.3. Run KLL Algorithm gets prediction score as 55%

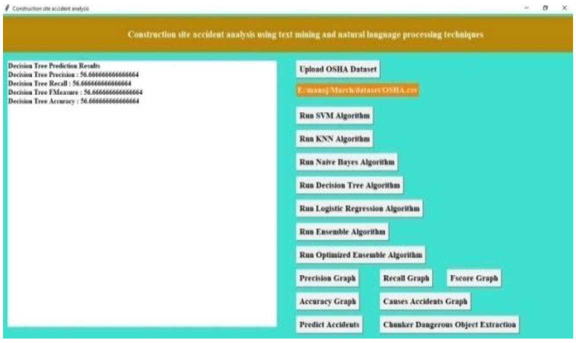
Fig.3.RunKNN Algorithm' button to get its prediction accuracy uploaded dataset and calculate its prediction accuracy, precision. we got 55% prediction accuracy.


Construction site accident analysis
Construction site accident analysis using text mining and natural language processing techniques
Upload OSHA Dataset
E:/manoj/March/dataset/OSHA.csv
Run SVM Algorithm
Run KNN Algorithm
Run Naive Bayes Algorithm
Run Decision Tree Algorithm
Run Logistic Regression Algorithm
Run Ensemble Algorithm
Run Optimized Ensemble Algorithm
Precision Graph
Recall Graph
Fscore Graph
Accuracy Graph
Causes Accidents Graph
Predict Accidents
Chunker Dangerous Object Extraction

```
Naive Bayes Prediction Results

Naive Bayes Precision : 50.83333333333333
Naive Bayes Recall : 50.83333333333333
Naive Bayes FMeasure : 50.83333333333333
Naive Bayes Accuracy : 50.83333333333333
```

Fig.4. 'Run Naïve Bayes Algorithm' get its accuracy

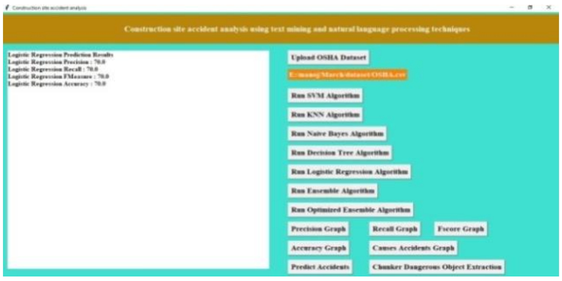
Figure 4 depicts 'Run Naïve Bayes Algorithm' button to get 50% accuracy


Construction site accident analysis
Construction site accident analysis using text mining and natural language processing techniques
Upload OSHA Dataset
E:/manoj/March/dataset/OSHA.csv
Run SVM Algorithm
Run KNN Algorithm
Run Naive Bayes Algorithm
Run Decision Tree Algorithm
Run Logistic Regression Algorithm
Run Ensemble Algorithm
Run Optimized Ensemble Algorithm
Precision Graph
Recall Graph
Fscore Graph
Accuracy Graph
Causes Accidents Graph
Predict Accidents
Chunker Dangerous Object Extraction

```
Decision Tree Prediction Results
Decision Tree Precision : 56.666666666666664
Decision Tree Recall : 56.666666666666664
Decision Tree FMeasure : 56.666666666666664
Decision Tree Accuracy : 56.666666666666664
```

Fig.5. 'Run Decision Tree Algorithm' button to gets 56% aaccuracy

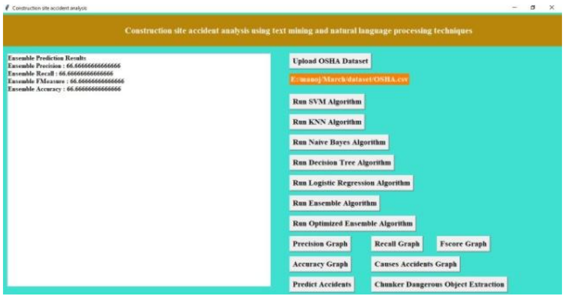
This figure 5 depicts click on 'Run Decision Tree Algorithm' button to get 56% aaccuracy.


Construction site accident analysis
Construction site accident analysis using text mining and natural language processing techniques
Upload OSHA Dataset
E:/manoj/March/dataset/OSHA.csv
Run SVM Algorithm
Run KNN Algorithm
Run Naive Bayes Algorithm
Run Decision Tree Algorithm
Run Logistic Regression Algorithm
Run Ensemble Algorithm
Run Optimized Ensemble Algorithm
Precision Graph
Recall Graph
Fscore Graph
Accuracy Graph
Causes Accidents Graph
Predict Accidents
Chunker Dangerous Object Extraction

```
Logistic Regression Prediction Results
Logistic Regression Precision : 70.0
Logistic Regression Recall : 70.0
Logistic Regression FMeasure : 70.0
Logistic Regression Accuracy : 70.0
```

Fig.6. ‘Run Logistic Regression Algorithm’ button to get 70% accuracy

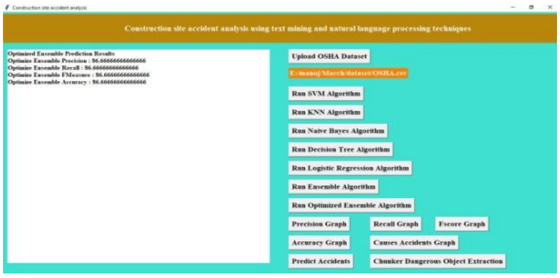
In this figure 6 click on ‘Run Logistic Regression Algorithm’ button to get 70% accuracy.


Construction site accident analysis
Construction site accident analysis using text mining and natural language processing techniques
Upload OSHA Dataset
E:/manoj/March/dataset/OSHA.csv
Run SVM Algorithm
Run KNN Algorithm
Run Naive Bayes Algorithm
Run Decision Tree Algorithm
Run Logistic Regression Algorithm
Run Ensemble Algorithm
Run Optimized Ensemble Algorithm
Precision Graph
Recall Graph
Fscore Graph
Accuracy Graph
Causes Accidents Graph
Predict Accidents
Chunker Dangerous Object Extraction

```
Ensemble Prediction Results
Ensemble Precision : 66.66666666666666
Ensemble Recall : 66.66666666666666
Ensemble FMeasure : 66.66666666666666
Ensemble Accuracy : 66.66666666666666
```

Fig.7. Run ‘Ensemble Algorithm’ button to get 66% accuracy.

Fig.7.Similarly run ‘Ensemble Algorithm’ button to get 66% accuracy.


Construction site accident analysis
Construction site accident analysis using text mining and natural language processing techniques
Upload OSHA Dataset
E:/manoj/March/dataset/OSHA.csv
Run SVM Algorithm
Run KNN Algorithm
Run Naive Bayes Algorithm
Run Decision Tree Algorithm
Run Logistic Regression Algorithm
Run Ensemble Algorithm
Run Optimized Ensemble Algorithm
Precision Graph
Recall Graph
Fscore Graph
Accuracy Graph
Causes Accidents Graph
Predict Accidents
Chunker Dangerous Object Extraction

```
Optimized Ensemble Prediction Results
Optimize Ensemble Precision : 86.66666666666666
Optimize Ensemble Recall : 86.66666666666666
Optimize Ensemble FMeasure : 86.66666666666666
Optimize Ensemble Accuracy : 86.66666666666666
```

Fig.8. Run ‘Optimized Ensemble Algorithm’ button to get 86% accuracy.

Similarly run ‘Optimized Ensemble Algorithm’button to get accuracy of voting classifier. Opti mize ensemble algorithm is also called as Voting Classifier . In this figure 9 depicts with Voting Classifier we got 86% accuracy

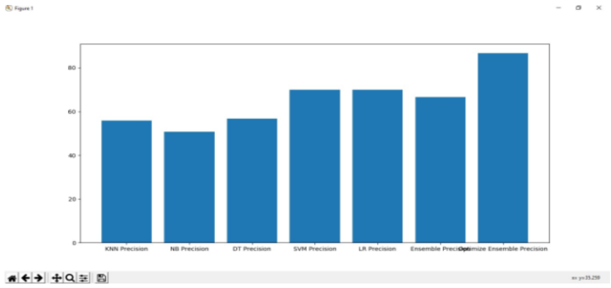


Fig.9: Precision Comparison between all algorithms

In this figure 9 depicts with Voting Classifier we got 86%accuracy. Now click on ‘Precision Graph’ button to view precision Comparison between all algorithms. In this 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.

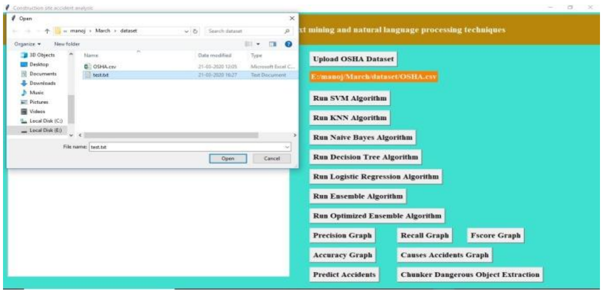
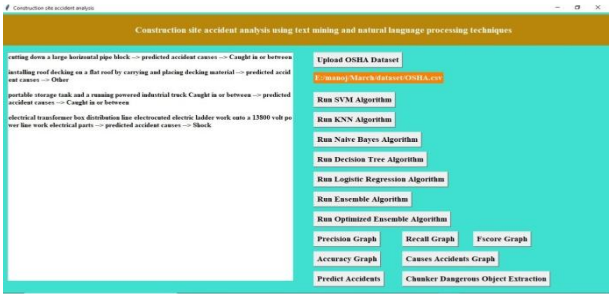


Fig. 10. Uploading ‘test.txt’ application predict accidents

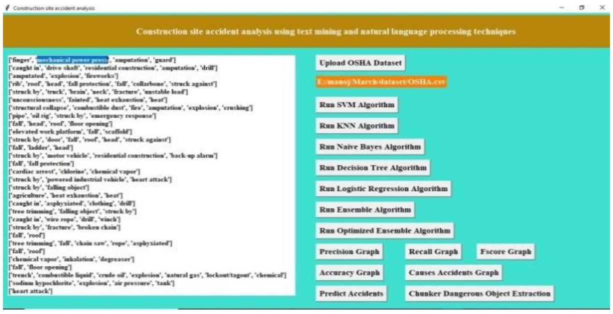
In this figure 10 depicts Iam uploading ‘test.txt’ which contains new work and application predict accidents which mayoccur in this work.


Construction site accident analysis
Construction site accident analysis using text mining and natural language processing techniques
Upload OSHA Dataset
E:/manoj/March/dataset/OSHA.csv
Run SVM Algorithm
Run KNN Algorithm
Run Naive Bayes Algorithm
Run Decision Tree Algorithm
Run Logistic Regression Algorithm
Run Ensemble Algorithm
Run Optimized Ensemble Algorithm
Precision Graph
Recall Graph
Fscore Graph
Accuracy Graph
Causes Accidents Graph
Predict Accidents
Chunker Dangerous Object Extraction

```
cutting down a large horizontal pipe block --> predicted accident causes --> Caught in or between

installing roof decking on a flat roof by carrying and placing decking material --> predicted accid
ent causes --> Other

portable storage tank and a running powered industrial truck Caught in or between --> predicted
accident causes --> Caught in or between

electrical transformer box distribution line electrocuted electric ladder work onto a 13800 volt po
wer line work electrical parts --> predicted accident causes --> Shock
```

Fig. 11. click on ‘Chunker Dangerous Object Extraction’

In this figure 11 depicts before first arrow mark is the description of work and after second arrow mark is the predicted future accident that may occur while doing this work. Now click on ‘Chunker Dangerous Object Extraction’ button to extract all dangerous object which has to remove while doing work and this objects can be sharp blades ormis used tools etc.


Construction site accident analysis
Construction site accident analysis using text mining and natural language processing techniques
Upload OSHA Dataset
E:/manoj/March/dataset/OSHA.csv
Run SVM Algorithm
Run KNN Algorithm
Run Naive Bayes Algorithm
Run Decision Tree Algorithm
Run Logistic Regression Algorithm
Run Ensemble Algorithm
Run Optimized Ensemble Algorithm
Precision Graph
Recall Graph
Fscore Graph
Accuracy Graph
Causes Accidents Graph
Predict Accidents
Chunker Dangerous Object Extraction

```
['finger', 'mechanical power press', 'amputation', 'guard']
['caught in', 'drive shaft', 'residential construction', 'amputation', 'drill']
['amputated', 'explosion', 'fireworks']
['rib', 'roof', 'head', 'fall protection', 'fall', 'collarbone', 'struck against']
['struck by', 'truck', 'brain', 'neck', 'fracture', 'unstable load']
['unconsciousness', 'fainted', 'heat exhaustion', 'heat']
['structural collapse', 'combustible dust', 'fire', 'amputation', 'explosion', 'crushing']
['pipe', 'oil rig', 'struck by', 'emergency response']
['fall', 'head', 'roof', 'floor opening']
['elevated work platform', 'fall', 'scaffold']
['struck by', 'door', 'fall', 'roof', 'head', 'struck against']
['fall', 'ladder', 'head']
['struck by', 'motor vehicle', 'residential construction', 'back-up alarm']
['fall', 'fall protection']
['cardiac arrest', 'chlorine', 'chemical vapor']
['struck by', 'powered industrial vehicle', 'heart attack']
['struck by', 'falling object']
['agriculture', 'heat exhaustion', 'heat']
['caught in', 'asphyxiated', 'clothing', 'drill']
['tree trimming', 'falling object', 'struck by']
['caught in', 'wire rope', 'drill', 'winch']
['struck by', 'fracture', 'broken chain']
['fall', 'roof']
['tree trimming', 'fall', 'chain saw', 'rope', 'asphyxiated']
['fall', 'roof']
['chemical vapor', 'inhalation', 'degreaser']
['fall', 'floor opening']
['trench', 'combustible liquid', 'crude oil', 'explosion', 'natural gas', 'lockout/tagout', 'chemical']
['sodium hypochlorite', 'explosion', 'air pressure', 'tank']
['heart attack']
```

Fig. 12. Tool extracted using noun phrases

In this figure121shows all tools names as noun phrases. Selected text in above screen is one of the example of tool extracted using noun phrases. Scroll down text area to get all tool names

5. Conclusion

We proposed an approach to automatically extract valid accident precursors from adataset of raw construction injury reports. Such information is highly valuable, as it can beused 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 ishigh for all models, we make the interesting observation that the simple TF-IDF + SVMapproach is on par with(or outperforms) deep learning most of the time.

References

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

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

[3] D.Govardhan Reddy, Ajaykumar Dharmireddy "Design of High Throughput AXI Compliant DDR3 Controller" International Journal of Advance Electrical and Electronics Engineering (IJAEEE), Volume-4 Issue-2, 2015, pp. 31-36

[4] 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.

[5] 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).30.

[6] J.M.Puithvi, Ajaykumar Dharmireddy " Control Module Design for ADC Based on FPGA" International Journal of electronics & communication technology, Vol. 5 ,sp.issue.3, 2014. Doi : 10.47893/IJESS.2014.1208.

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

[8] P. Ravali Teja , Ajaykumar Dharmireddy "Design And Analysis of Parallel BCD Adder With Power Gated Circuit for Future Technologies" International Conference on Recent Trends In Science And Technology, Vol. 5 ,sp.issue.3, 2014. Doi : 10.47893/IJESS.2014.1208.

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

[10] V. Lakshma Reddy , H. Sudhakar , Ajaykumar Dharmireddy "Realization of Redundant Binary Multiplier with Modified Partial Product Generator Using Verilog" International Journal of Scientific Research in Computer Science, Engineering and Information Technology, Vol.2,Issue 6 ,Dec 2017, pp. 924-927.

[11] W. J. Wiatrowski, J. A. Janocha, "Comparing fatal work injuries in the united states and the euro peanunion" ,Tech.rep., Accessed 21st January. 2020 (June2014).URL https:// www.bls.gov/ opub/mlr/ 2014/ article/ comparing- fatal- work –injuries -us-eu.htm

[12] I. Good fellow, Y.Bengio, A. Courville, " Deep learning", MIT Press, 2016. URL http://www .deep learning book.org

[13] J.Schmid huber, Deeplearning in neural networks: An overview, Neural Networks 61 (2015) 85– 117.

doi:10.1016 /j.neunet. 2014.09.003.

[14] Y.LeCun, L.Bottou,Y.Bengio, P.Haffner, Gradie nt-based learning applied to document recognition, Proceedings of the IEEE 86(11)(1998) 2278–2324. doi:10.1109/5. 726791.

[15] Ajaykumar Dharmireddy. Surya Manohar, G.T.Sri Hari, G. Gayatri, A. Venkateswarlu, "Detection of COVID-19 from X-RAY Images using Artificial Intelligence (AI)" 2022 International Conference on Intelligent Technologies (CONIT), PP.1-5, 2022. DOI**:** 10.1109/ CONIT 55038. 2022. 9847741

[16] Ajaykumar Dharmireddy, M. Greeshma, S. Chalasani, S. T. Sriya, S. B. Ratnam and S. Sana, "Azolla Crop Growing Through IOT by Using ARM CORTEX-M0," 2023 3rd International conference on Artificial Intelligence and Signal Processing (AISP), VIJAYAWADA, India, 2023, pp. 1-5.

[17] T.Mikolov, K.Chen, G.Corrado, J.Dean, "Efficient estimation of word representations in vector space, arXiv preprint, arXiv:1301.3781.

[18] Y. Kim, Convolutional neuralnetworks for sentence classification, arXiv preprint,arXiv:1408.5882

[19] Ajaykumar Dharmireddy, I.S.R, P.H.S.T. Murthy Performance analysis of tri-gate soi finfet Structure with various fin heights using TCAD Simulation" JARDCS Vol. 11, Issue spl - 2, Jan - March 2019.

[20] Ajaykumar Dharmireddy, Dr Sreenivasa Rao Ijjada,I.Hemalatha "Performance analysis of various Fin patterns of hybrid Tunnel FET" International Journal of Electrical and Electronics Research , Vol.10 issue no.4,pp. 806–810, 2022.

[21] Ajaykumar Dharmireddy and Sreenivasarao Ijjada (2023), Performance Analysis of Variable Threshold Voltage (ΔVth) Model of Junction less FinTFET. IJEER 11(2), 323-327. 2023. DOI: 10.37391/IJEER.110211.

[22] Ajaykumar Dharmireddy, Sreenivasarao Ijjada, H Sudhkar, Hemalatha. I "High Switching Speed and Low Power Applications for Hetro Junction DGTFET" Telematique, Vol.22 issue no.1,pp. 165–172, 2023.

[23] Ajaykumar Dharmireddy, Sreenivasa Rao Ijjada , "Steeper Slope Characteristics of DM FINTFET" European Chemical Bulletin, Vol.12 issue no.1,pp. 1312–1321, 2023.