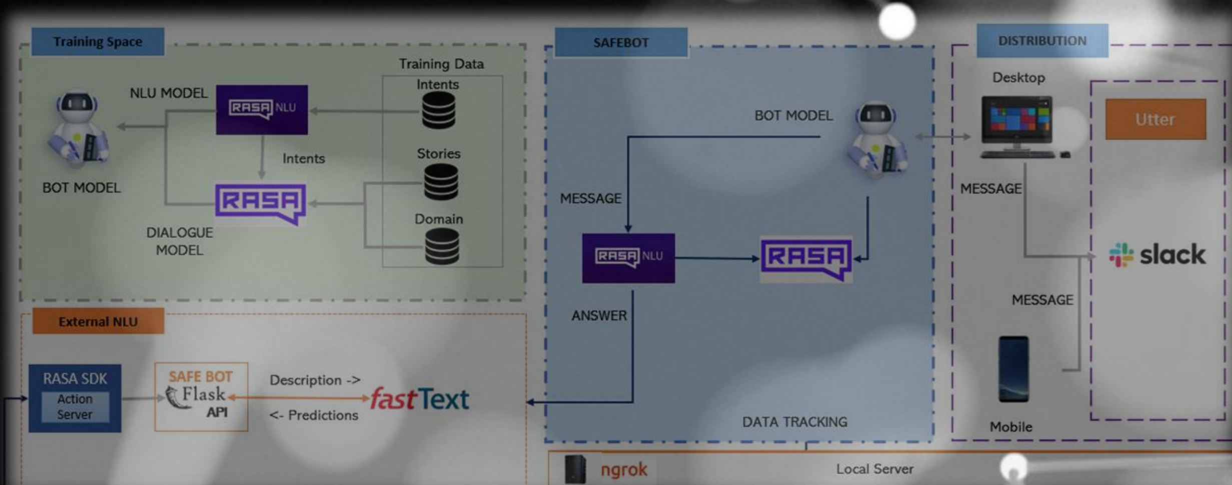
FINAL REPORT

SAFEBOT



CAPSTONE PROJECT | NLP 2 GROUP 2

Table of Contents

Mentor: Mr. Rohit Raj

Members:
Shankhadeep Chaudhuri
Mario Mathew
John Cherian
Kiran Bobba
Ajay Kumar

SUMMARY

There is an emerging need in the current industrial sector to find a solution for the increase in recent injuries/accidents in plants. There are cases of death which has also been registered.

The industry is in dire need of a solution for it’s employees to safeguard their interests in certain sectors.

SAFEBOT

Project Overview
Abstract

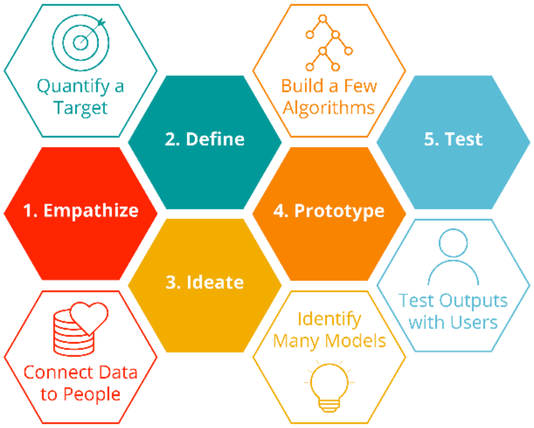
WELCOME

Business analytics are becoming commonplace in the government and private sectors where organizations are using historical performance data and predictive modeling to support a wide variety of operational and business needs.

Performance management related to safety is not new, but risk and safety management solutions which integrate key performance indicators, incident and near miss data, modeling results and subjective inputs from the workforce are propelling organizations into the next phase. Modern data science tools are capable of extracting, integrating and analyzing previously inaccessible and siloed data. Monitoring culture and ultimately predicting safety performance are no longer impossible tasks.

Industries around the world are in the dark when it comes to educating their stakeholders and taking precautionary measures in a time bound manner about industrial safety. The costs incurred due this is often high – injury and death of employees and contractors, causing pain and suffering to them and their families and resulting in hospitalization charges, litigation fees, loss of reputation and lost employee morale.

The purpose of the project is to make a chatbot that can be easily deployed on any platform with slack messenger installed, so that it can be used by any personnel in companies interested in improving safety and safety related regulations and practices.

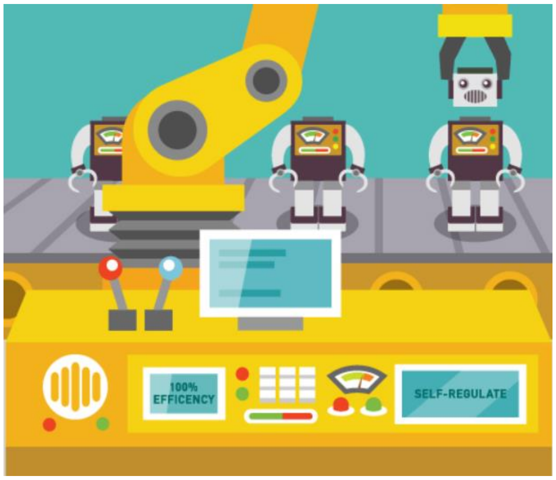


An AI based NLP chatbot for industrial safety deployed on the messaging platform – Slack.

We have designed a highly custom configurable architecture using the open source Rasa chatbot project as our base chatbot implementation where we plug in our custom deep learning models which can be upgraded as technology and algorithms improve over time.

We have toyed with various machine learning and deep learning models and for our prediction engine and have narrowed down to a few deep learning models due to performance factors. We are continuously striving to maximize on our metrics on the same.

The end result will be an end-to-end chatbot implementation for companies interested in increasing timely safety related information access that reduces injuries, deaths and related litigation expenses, emotional trauma and lost employee morale most of which once lost is impossible to retrieve thus making the offering a must have.

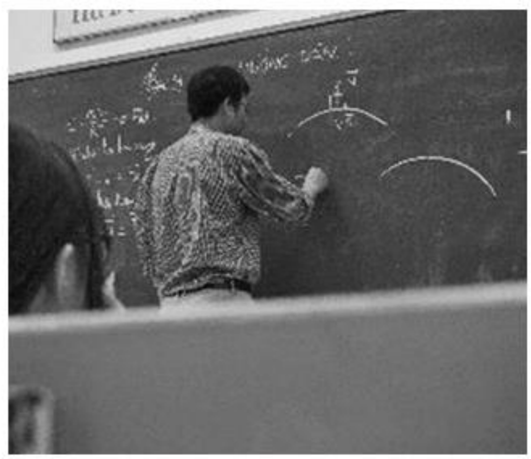


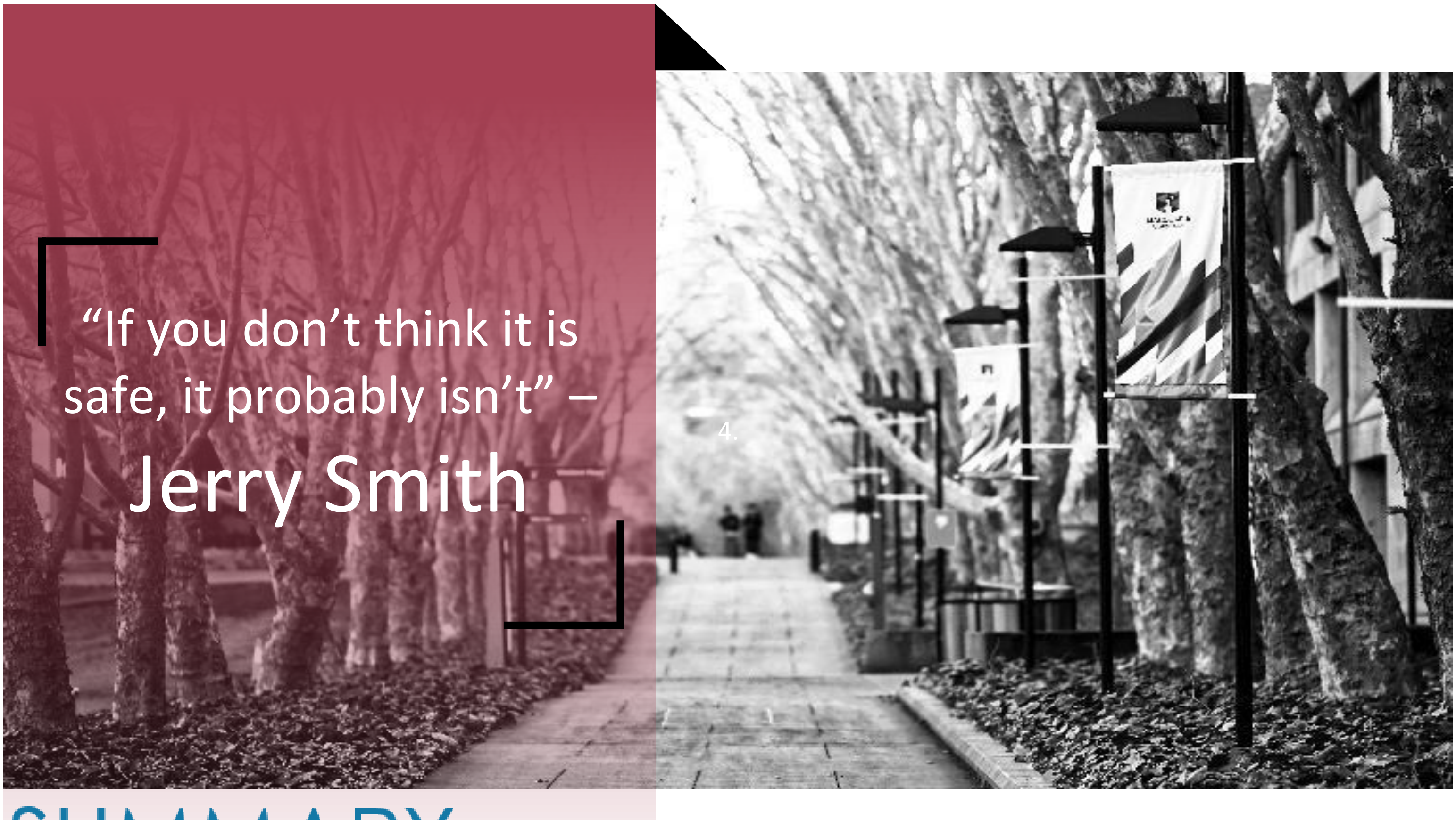
PROBLEM STATEMENT

For industries around the world, accidents in the work place are of a major concern, since it affects the lives and wellbeing of their employees, contractors and their families and the industry faces loses in terms of hospital charges, litigation fees, reputation and lost employee morale. Based on these facts it is intended to build a chatbot that can highlight the safety risk as per the incident description to the professionals including:

- ✓ Personnel from the safety and complaince team
- ✓ Senior management from the plant
- ✓ Personnel from other plants across the globe
- ✓ Government and industrial safety groups
- ✓ Anyone interested or doing research in industrial safety
- ✓ Emergency health and safety teams
- ✓ Fire safety and industrial hazard teams
- ✓ General management
- ✓ Other personnel requiring safety risk information

so that these professionals can:
Take preventive and proactive measures based on history React faster to employee satisfaction related to safety Help position the equipment and machinery in a safe place where risk of potential accidents can be minimized Gain insights about safety in industries safety is paramount Reduce insurance costs by better handling of personnel, equipment and other resources Take other safety related decisions and actions.





SUMMARY

INSIGHTS

On inspection of the dataset, it appears that:S

- The dataset is limited and consists of four hundred and twenty-five records only so training the models with high accuracy could be a challenge
- The dataset is imbalanced on certain variables like potential accident level and accident level, this means that we may not get consistent results unless the dataset is treated to reduce imbalance.
- Minor accidents are more common than major accidents, this looks like real world situations.
- There is data from three countries.
- There are twelve locals or cities from which the data is taken.
- There are two industry sectors - mining, metals and third all others grouped together as others.
- There are five accident levels.
- There are six potential accident levels.
- There are employees, third parties and remote third parties involved in the accidents.
- There are thirty-three different types of critical risk one of which has been assigned to an accident incident.
- The accident description is highly unclean and so it will require a considerable amount of effort to clean it to produce results.
- The dataset consists of data from January 2016 to July 2017.
- Males are more involved than females in accidents, this too looks similar to real world situations as there are considerably lower number of females working in industrial environments.

OBJECTIVE

GOAL

- To create an industrial conversational bot
  - The chatbot will be used as an guide for the customer, employee and management to access the potential risk which might be involved for a specific sphere of work.

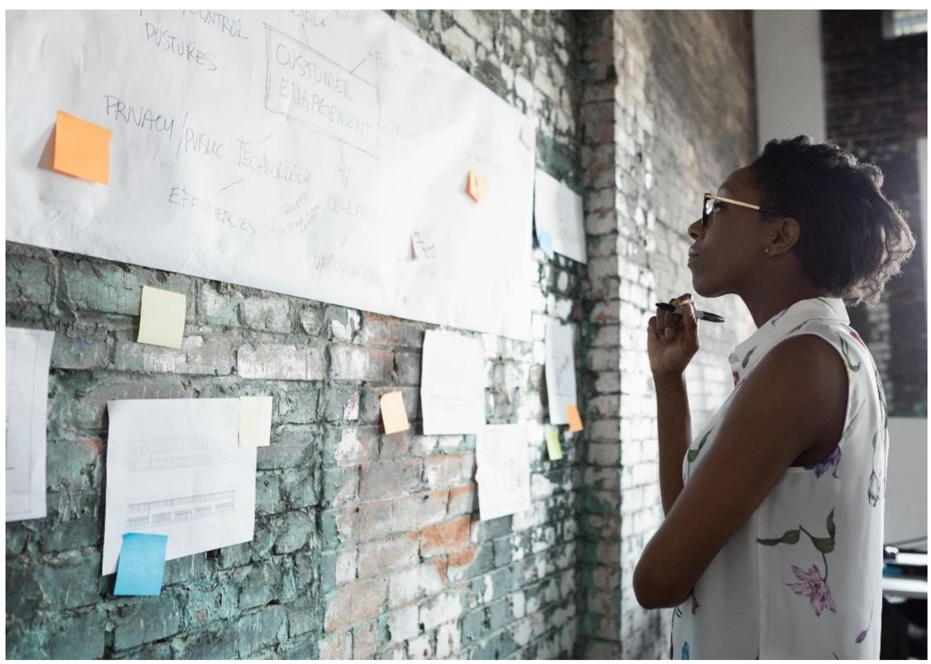
HIGHLEVEL FINDINGS

- Many body-related actions and accidents has been found.
- A lot of equipment related accident has been mentioned in the dataset.
- Poor features map found with lack of access to quality data found.

HIGHLEVEL IMPLICATIONS

- The main causes of the accidents are hand related operation and time base factor.
- More strict safety standard needs to be maintained to reduce accidents.
- Equipment based safety standards needs to be defined.

The broader version is evaluated further in the EDA.



.

EDA

GENERAL INDUSTRY MEASURES - SAFETY

- Proper Plant Layout
- Proper Fire Prevention System
- Health & Hygiene
- Proper Safety Training
- Proper Alarms And warning systems
- Appropriate sensors and safety gears for employees
- Sufficient lighting in the work area as well as the pathways
- Use proper tool for the job.
- Always wear proper safety gear for the work.
- Use proper tool for the job.
- Make sure chemicals are properly labeled and stored.

EMPLOYEE BENEFITS

Industrial safety knowledge.

- Risk avoidance.
- Safety features.

APPROACH

The approach was initially to remove the stop words, use lemmatization.
We started using N Gram, Univariate and Bivariate and time series analysis to decide on the type, trend and pattern of the accident causes.

- Use Pre-processing technique
  - ➢ Time-related feature extraction
  - ➢ NLP pre-processing (special characters removal, removing stop words)
- Practice EDA technique
- Practice visualizing technique.
- Practice feature engineering technique
- Practice modeling technique
- Causal analysis skill

PRE-PROCESSING

- We noticed that except a 'date' column all other columns are categorical columns.
- We observed that there are records of accidents from 1st Jan 2016 to 9th July 2017 in every month. So there are no outliers in the 'Date' column.
- There are 12 Local cities where manufacturing plant is located and it's types are in sequence so there are no outliers in 'Local' column.
- There are only three Industry Sector types which are in sequence so there are no outliers in 'Industry Sector' column.
- There are only five Accident Level types which are in sequence so there are no outliers in 'Accident Level' column.
- There are only six Potential Accident Level types which are in sequence so there are no outliers in 'Potential Accident Level' column.
- There are only two Gender types in the provided data so there are no outliers in 'Gender' column.
- There are only three Employee types in the provided data so there are no outliers in 'Gender' column.
- There are quite a lot of Critical risk descriptions, and we don't see any outliers but with the help of SME we can decide whether this column has outliers or not.

INSIGHTS

- ➢ Though the staffs of the manufacturing plants are mostly males, EDA shows that males are likely involved in accidents (95%).
- ➢ And males tend to get involved in accidents with higher risk levels than females.
- ➢ Comparing employee's accidents count with third parties' accide0.nts count, EDA shows that third parties are likely involved in accidents (58%)

WHAT'S NEXT

Deciding Models and Model Building

Objective:

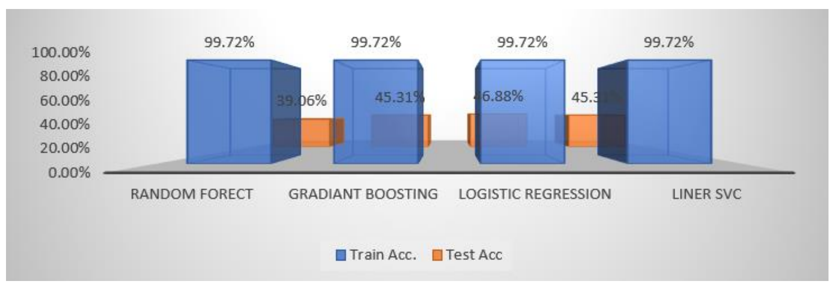
- Presumption of cause of accidents
- Surveying a factor that increases severity of accidents

Building the model which classify the severity of accidents, we can understand the factor related to the causality of accidents.
So, two models were built based on those cases below.

- Accident Level.
- Potential accident level.

The model that has been used are below:

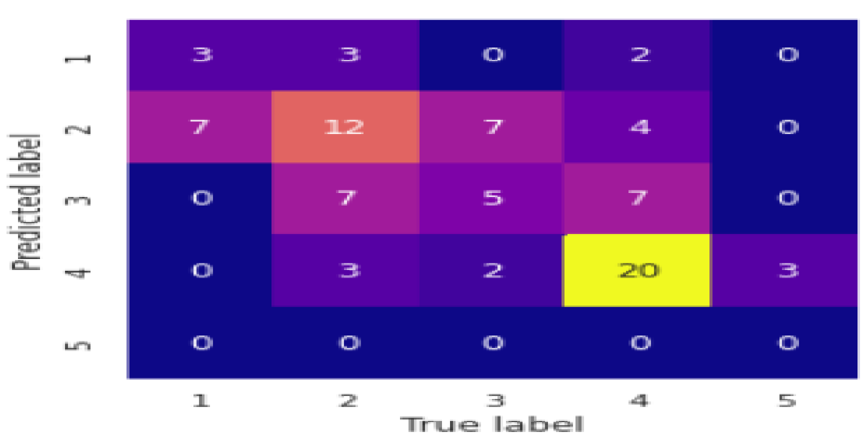
- Random Forest
- Gradient Boosting
- Logistic Regression
- Linear SVC model



While using Neural Network Classifiers the steps involved were:

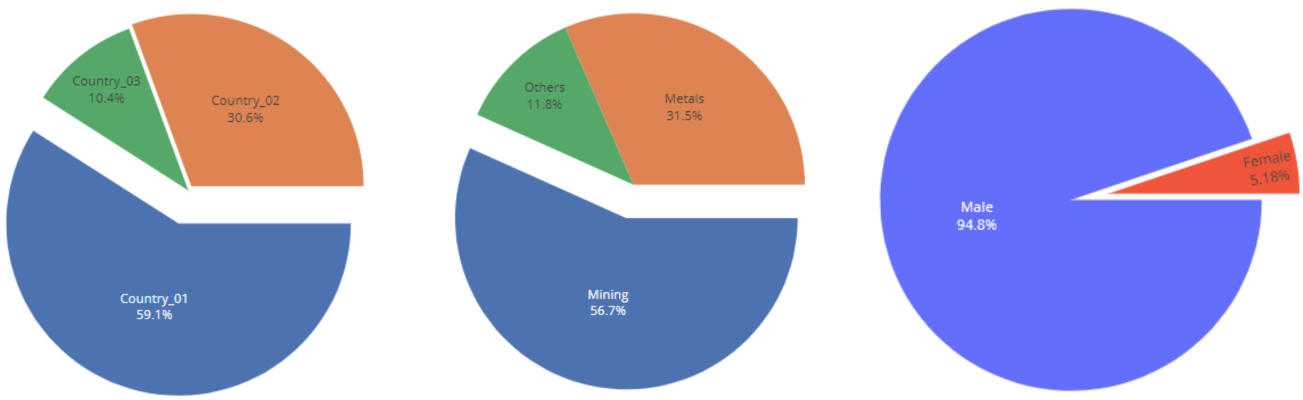
- Padding sequences.
- Use of Glove embedding
- RNN Classifier
- LSTM Classifier

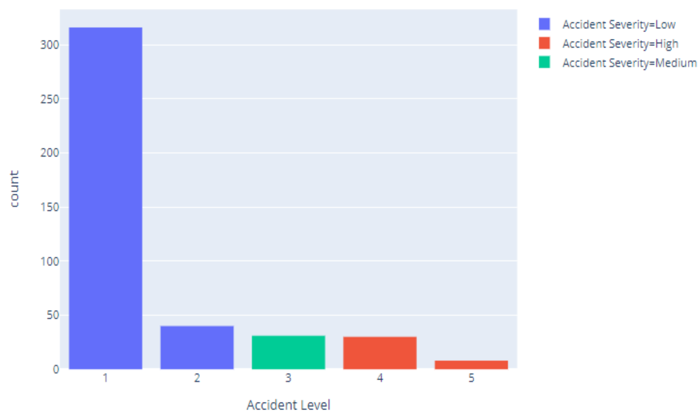
CONFUSION MATRIX



VISUAL ANALYSIS

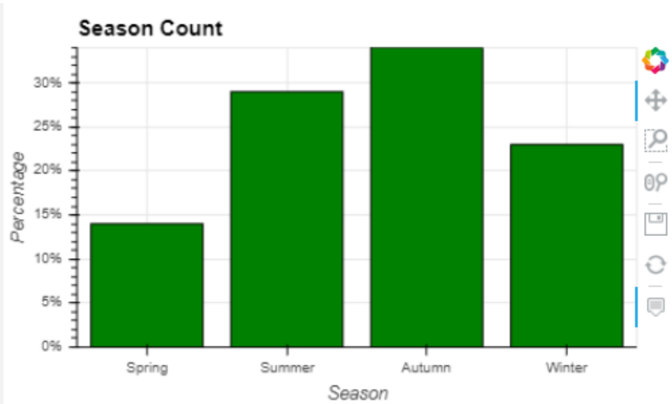
	Q1	Q2	Q3
COUNTRY	Country01: 59.1$	Country02: 38.6%	Country03: 10.4%
SECTORS	Metals: 31.5%	Mining: 56.7%	Others:11.8%
GENDER	Male: 94.8%	Female: 5.18%	



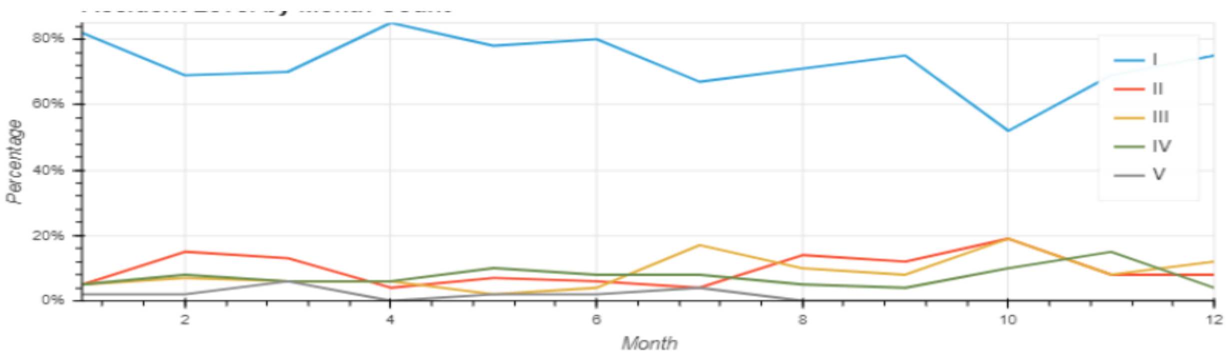


Categorization has been done for accident severity which also follows the most obvious trends/

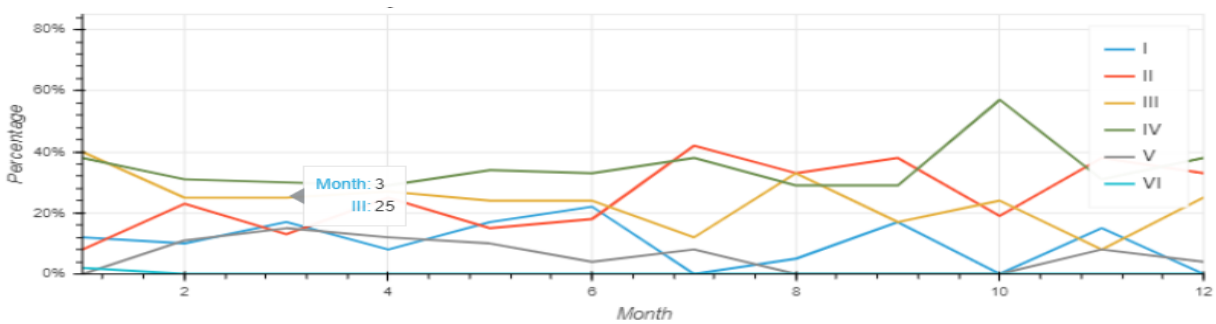
Most possible cases of accident has been witness during Autumn.

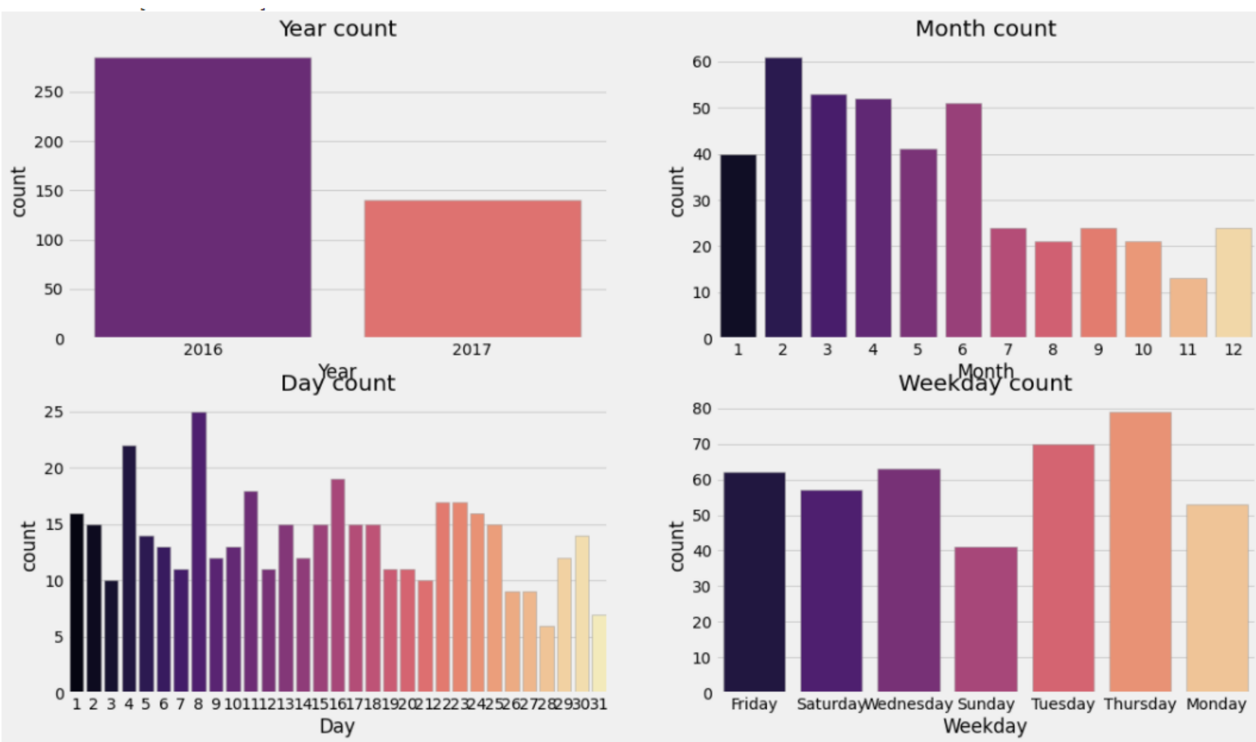


Accident Level monthly occurrences:



Potential Level monthly occurrences:

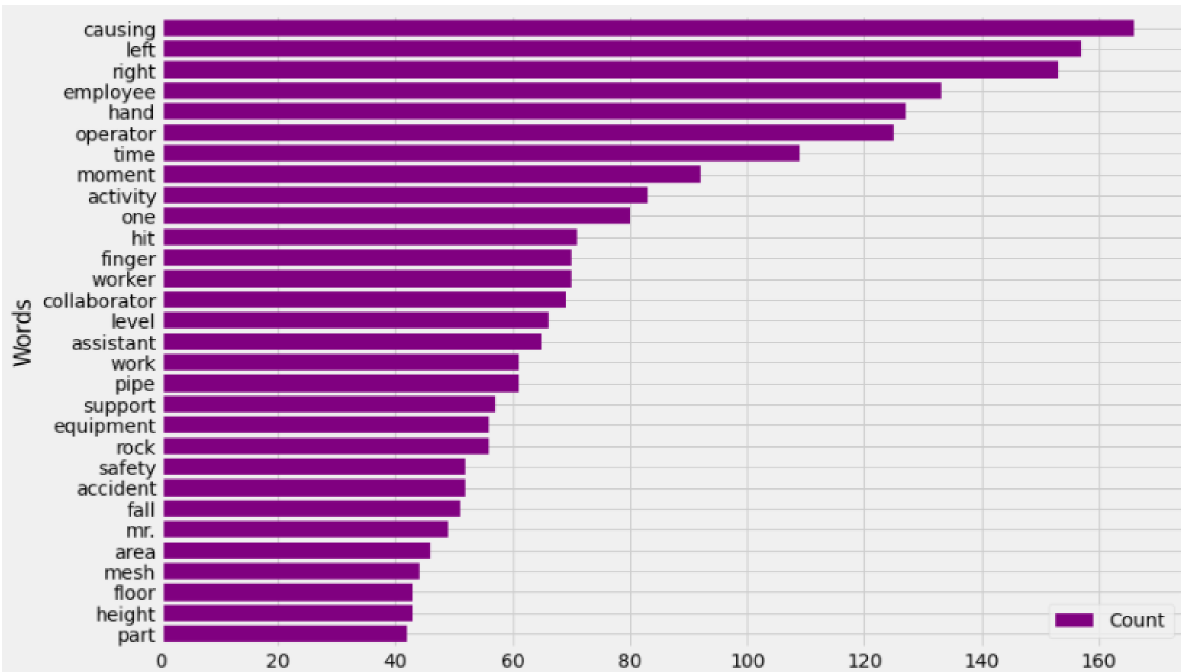




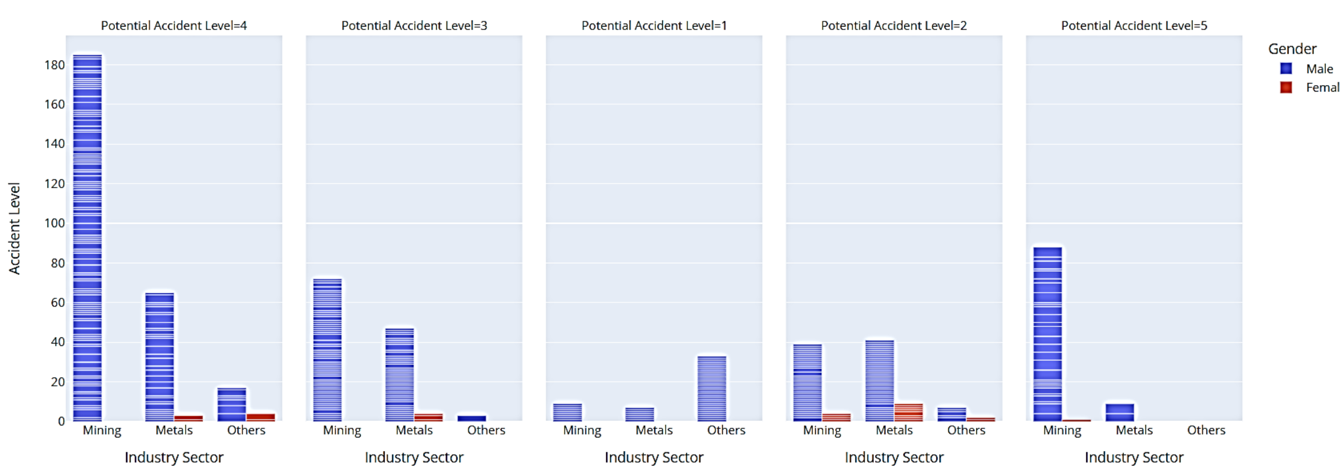
It is observed that number of accidents occurred in 2016 is more compared to 2017, in year 2016 we have all 12 months of data whereas year 2017 has only 7 months of data. It seems that the number of accidents decreased in latter of the year / month.

CAUSAL ANALYSIS

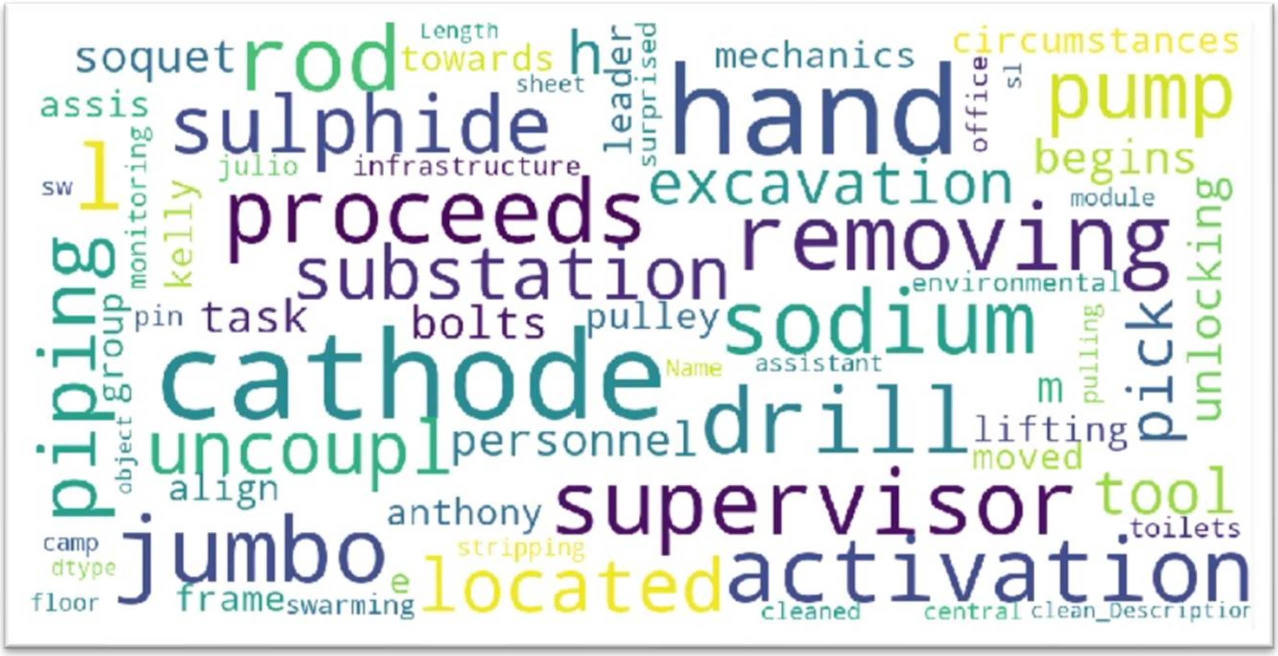
- ✓ The most frequent word is causing (a verb).
- ✓ There are several nouns like pipe, collaborator, time etc.
- ✓ Most accidents involved the hands of the persons involved from which we can draw a statement that operation procedure well define can definitely impose accident-avoidance conditions.
- ✓ There are other words which depict some sort of action (verbs). For example, hit, remove, fall move...etc.



From the below diagram we can draw that male are the most effected gender with Potential Accident Level 4 and 5 which is from Mining sector.



WORD CLOUD REPRESENTATION

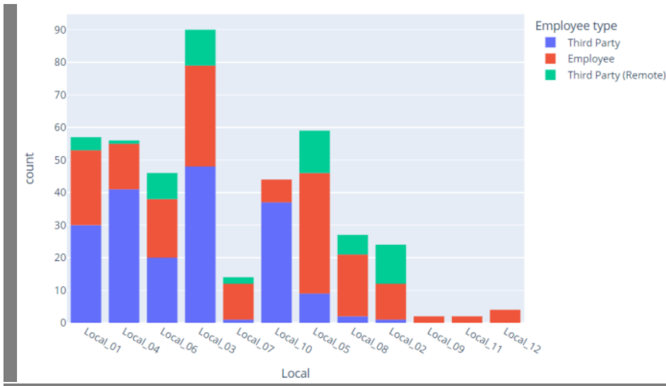


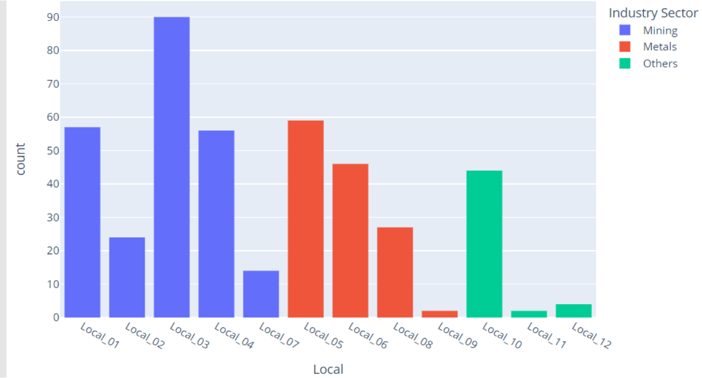
Observations

There are many body-related, employee related, movement-related, equipment-related and accident-related words.

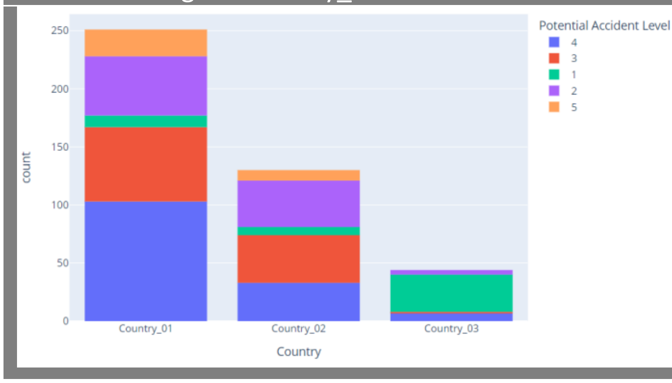
- Body-related: left.
- Employee-related: employee, supervisor.
- Movement-related: drill, lift, align, remove
- Equipment-related: cathode, pump, rod, sodium.
- Accident-related: accident, activity.

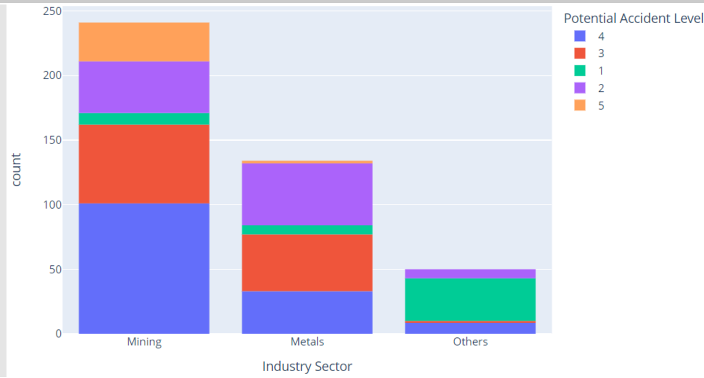
MULTI VARIATE ANAYSIS	
✓ **Local_3 is the most effected city and most effected class of Employee type are Third Party and Employee.**	✓ Local 3 has highest number of Mining industry sector accident. ✓ Local 5 has highest number of Metals industry sector accident. ✓ All the Mining industry sector accidents happened in Local 1,2,3,4,7. ✓ All the Metals industry sector accidents happened in Local 5,6,8,9. ✓ All the Others industry sector accidents happened in Local 10,11,12.



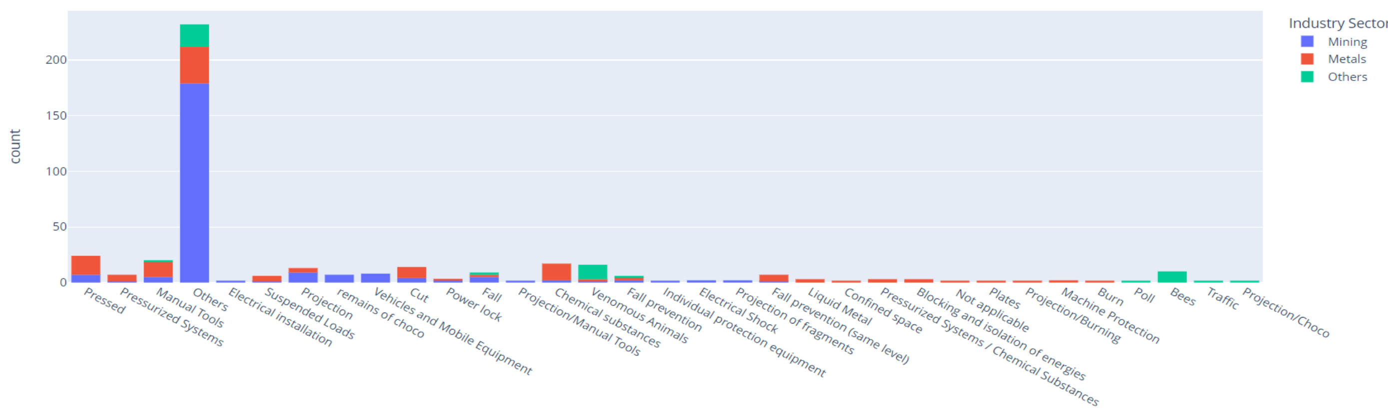


- ✓ Country_01 is the most effected country and most of the classes of Potential Accident Level belongs to country_01
- ✓ Mining sector is the most effected and severity level of Accidents also belongs to the same sector

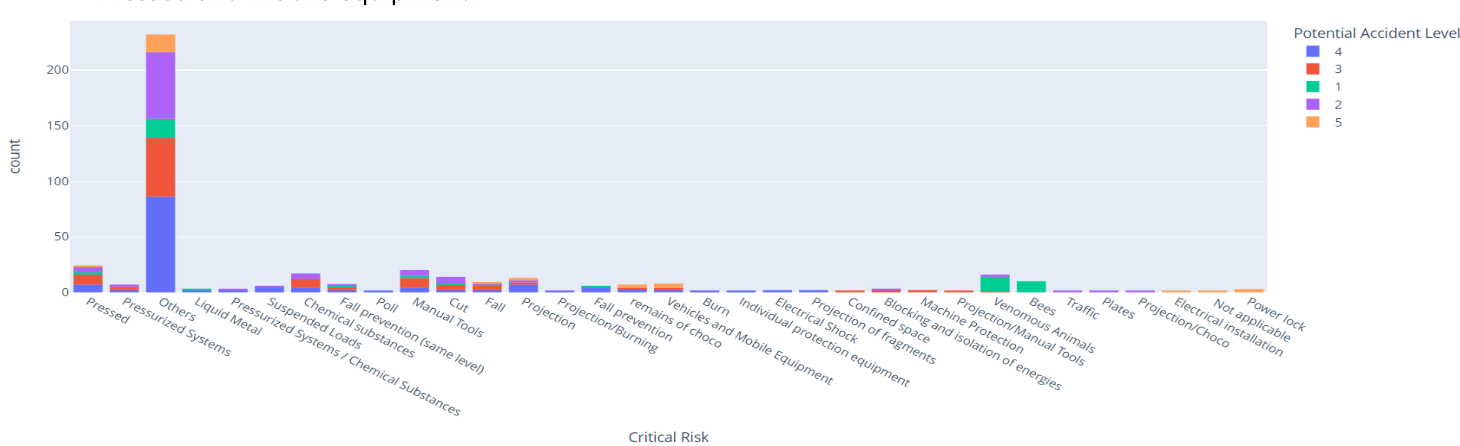




- ✓ Mining sector is the most effected sector and most of the classes of Critical Risk comes from this sector.



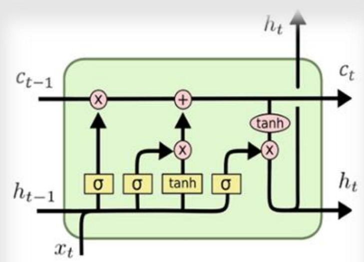
- ✓ Most of the classes of Potential Accident Level are from other class of Critical Risk which is 232 in No.
- ✓ The severity of the Potential Accident Level are from the class Fall, Electrical installation, Vehicles, Projection, Pressed and Mobile equipment.



MODEL TUNNING

fastText

- FastText is an open-source, free, lightweight library that allows users to learn text representations and text classifiers
- It works on standard, generic hardware. Models can later be reduced in size to even fit on mobile devices.



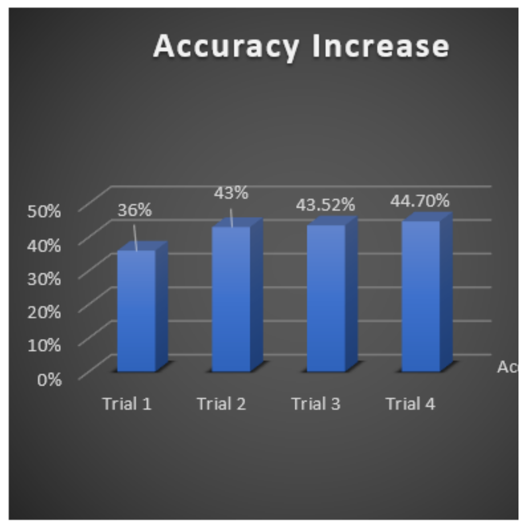
LSTM
(Long-Short Term Memory)

- An artificial recurrent neural network (RNN) architecture used in the field of deep learning.
- It is well-suited to classifying, processing and making predictions based on time series data, since there can be lags of unknown duration between important events in a time series.

FASTTEXT - APPROACH

- With Hyper parameters tuning with below code such as EPOC, Learning Rate, Wordgrams, hierarchical softmax and Multi label(just tried). Training with EPOC 300 for both Accident and Potential accident levels.
- With WordNGrams : adding more than '1' Wordgram decreasing the accuracy so No effect or improvement adding wordgram hyperparameter
  So far FastText giving the accuracy of
  Potential Accident level - 43%

Run Seq.	Parameters	Accuracy
1	lr=0.7	36%
	epoch=300	
	bucket=200000	
	dim=50	
	loss='hs'	
2	lr=0.7	43%
	epoch=300	
	bucket=200000	
	dim=50	
	loss='ova'	
3	lr=0.7	43.52%
	epoch=300	
4	lr=0.7	44.70%
	epoch=300	
	wordNgrams=1	

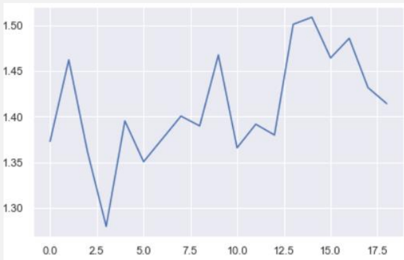


LSTM - APPROACH

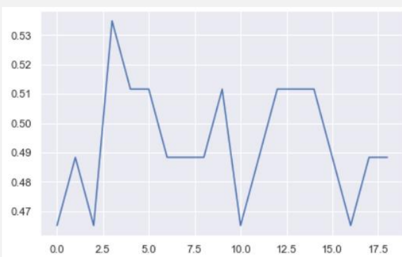
A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate. The cell remembers values over arbitrary time intervals and the three gates regulate the flow of information into and out of the cell.

LSTM networks are well-suited to classifying, processing and making predictions based on time series data, since there can be lags of unknown duration between important events in a time series. LSTMs were developed to deal with the vanishing gradient problem that can be encountered when training traditional RNNs. Relative insensitivity to gap length is an advantage of LSTM over RNNs, hidden Markov models and other sequence learning methods in numerous applications.

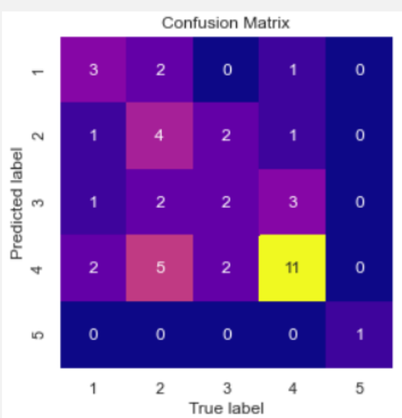
Model Loss function



Model accuracy



Confusion Matrix



Summary:

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding (Embedding)        (None, 85, 200)           437000
_________________________________________________________________
bidirectional (Bidirectional (None, 85, 100)           100400
_________________________________________________________________
global_max_pooling1d (Global (None, 100)               0
_________________________________________________________________
dropout (Dropout)            (None, 100)               0
_________________________________________________________________
dense (Dense)                (None, 5)                 505
=================================================================
Total params: 537,905
Trainable params: 100,905
Non-trainable params: 437,000
_________________________________________________________________
```

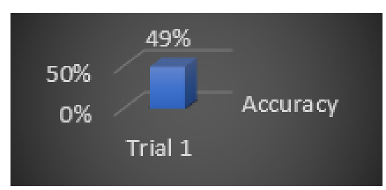
<u>Approach</u>

- Steps used :
  - Adding the Bidirectional LSTM layer with 128 units
  - Adding global pooling to make it 1D
  - Adding dropout to avoid overfitting
  - Dense(6, activation='softmax')

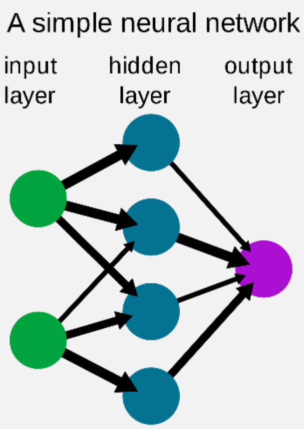
Classification Report:

```
              precision    recall  f1-score   support

           1       0.50      0.50      0.50        10
           2       0.80      0.32      0.46        25
           3       0.34      0.86      0.49        14
           4       0.57      0.48      0.52        33
           5       0.50      0.33      0.40         3

    accuracy                           0.49        85
   macro avg       0.54      0.50      0.47        85
weighted avg       0.59      0.49      0.49        85
```

Run Seq.	Parameters	Accuracy
1	batch_size=32 epochs=25 callbacks= [callback_list]	51%



Bi-Directional LSTM is working with best accuracy of 49%. Model needs more data cleaning.



N U E R A L N E T W O R K C L A S S I F I E R

Approach :

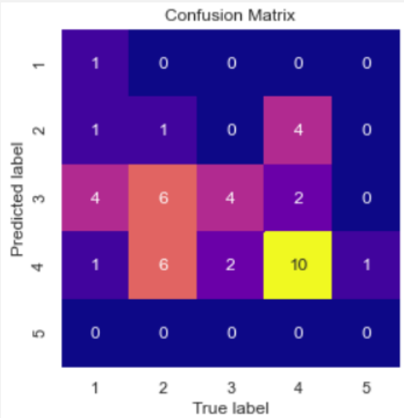
- The model is split into 382 group for train and 43 groups for test.
- Max features used is 10000
- Steps used :
  - Pad sequences
  - Creating weight matrix and Glove embeddings.
  - Glove used : glove.6B.200d
- Layer structure used :
  - Dense(100, activation='relu',input_shape=())
  - Dropout(0.4))
  - BatchNormalization()
  - Dense(50, activation='relu')
  - Dropout(0.4)
  - BatchNormalization()
  - Dense(25, activation='relu')
  - Dropout(0.4)

Model Summary

Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 85, 200)	437000
flatten (Flatten)	(None, 17000)	0
dense (Dense)	(None, 100)	1700100
dropout (Dropout)	(None, 100)	0
batch_normalization (BatchNo	(None, 100)	400
dense_1 (Dense)	(None, 50)	5050
dropout_1 (Dropout)	(None, 50)	0
batch_normalization_1 (Batch	(None, 50)	200
dense_2 (Dense)	(None, 25)	1275
dropout_2 (Dropout)	(None, 25)	0
dense_3 (Dense)	(None, 5)	130

```
Total params: 2,144,155
Trainable params: 1,706,855
Non-trainable params: 437,300
```

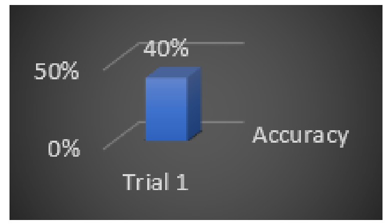
Confusion Matrix



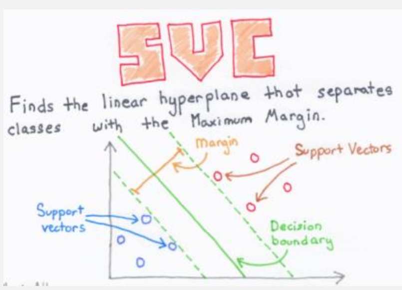
Classification Report:

```
              precision    recall  f1-score   support

           1       1.00      0.14      0.25         7
           2       0.17      0.08      0.11        13
           3       0.25      0.67      0.36         6
           4       0.50      0.62      0.56        16
           5       0.00      0.00      0.00         1

    accuracy                           0.37        43
   macro avg       0.38      0.30      0.25        43
weighted avg       0.43      0.37      0.33        43
```

Run Seq.	Parameters	Accuracy
1	batch_size=32	40%
	epochs=100	
	max_features = 10000	
	callbacks= [callback_list]	



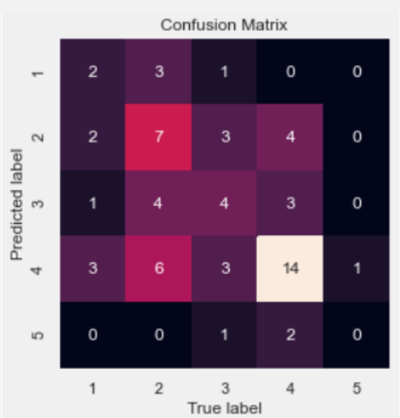
What we could ascertain is the Neural Network model is not learning well. Accuracy is 40%.



LINEAR SVC MODEL

The Linear Support Vector Classifier (SVC) method applies a linear kernel function to perform classification and it performs well with many samples. If we compare it with the SVC model, the Linear SVC has additional parameters such as penalty normalization which applies 'L1' or 'L2' and loss function.

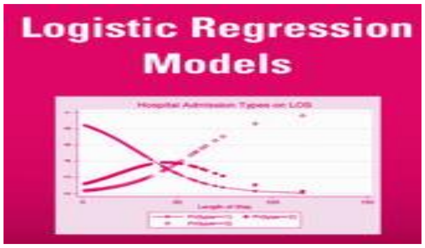
Confusion matrix



Classification Report:

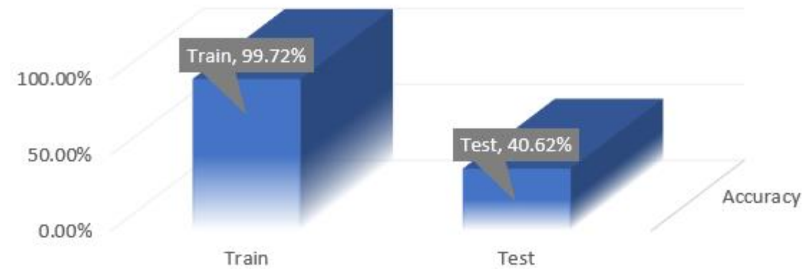
```
              precision    recall  f1-score   support

           1       0.33      0.25      0.29         8
           2       0.44      0.35      0.39        20
           3       0.33      0.33      0.33        12
           4       0.52      0.61      0.56        23
           5       0.00      0.00      0.00         1

    accuracy                           0.42        64
   macro avg       0.32      0.31      0.31        64
weighted avg       0.43      0.42      0.42        64
```

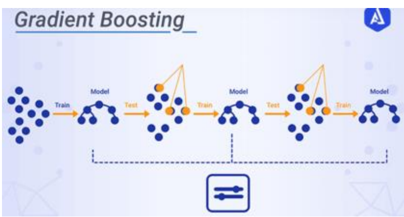
What we could ascertain is the SVC model is not learning well. Accuracy is 42%.



LOGISTIC REGRESSION

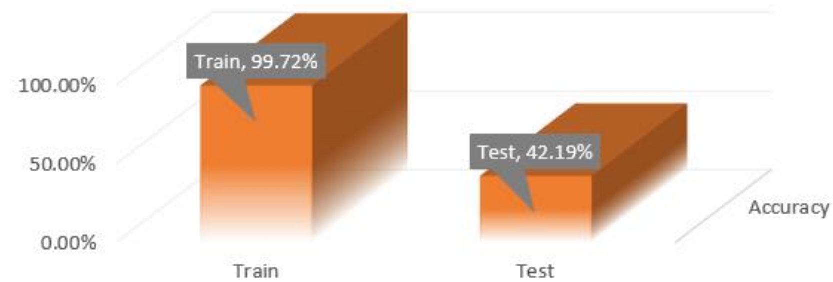
Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist. In regression analysis, logistic regression (or logit regression) is estimating the parameters of a logistic model (a form of binary regression.

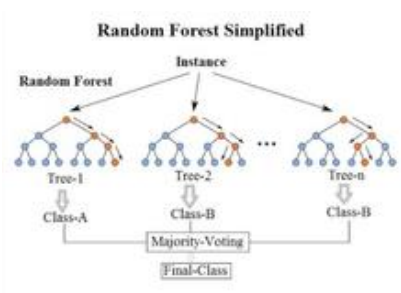




GRADIENT BOOSTING

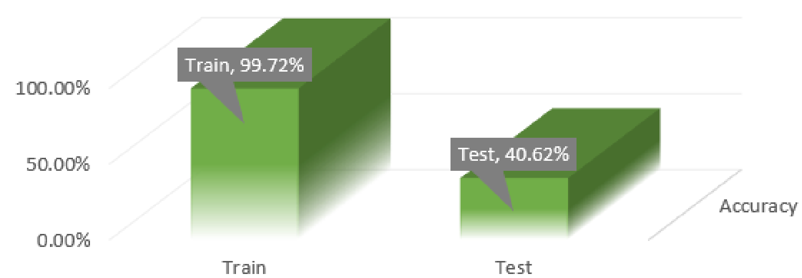
Gradient boosting is a machine learning technique for regression, classification and other tasks, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees.



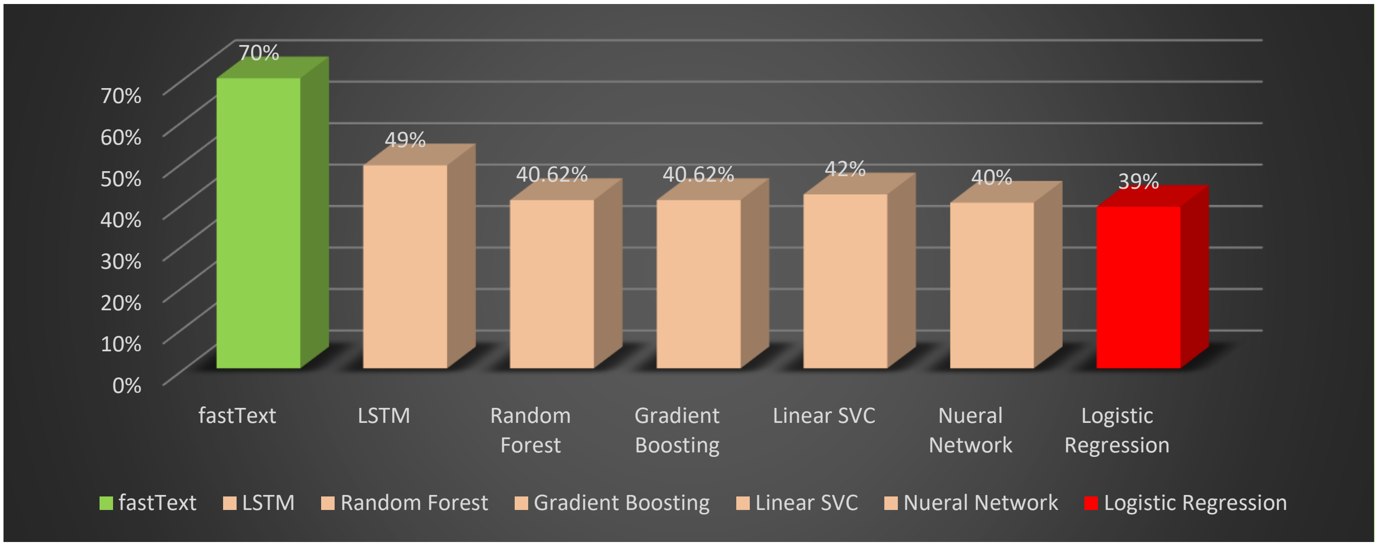


RANDOM FOREST

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time. For classification tasks, the output of the random forest is the class selected by most trees. For regression tasks, the mean or average prediction of the individual trees is returned.



GLOBAL MODEL COMPARISON



Key Highlights:

- The best performing model is LSTM.
- The poorest performer is Logistic regression.

Model Descriptions :

- ✓ A **Bidirectional LSTM** is a sequence processing model that consists of two LSTMs: one taking the input in a forward direction, and the other in a backwards direction. We have used Bidirectional LSTM with various hyperparameters such as 50 LSTM cells and Dropout to avoid overfitting. We have also used callback functions to stop the running if val_loss did not improve with patient level of 10. The Model is facing some overfitting issue due to less no. of observation and lots of Naming and unwanted words in the corpus. We are taking manual steps to clean the corpus as we have 425 observations.

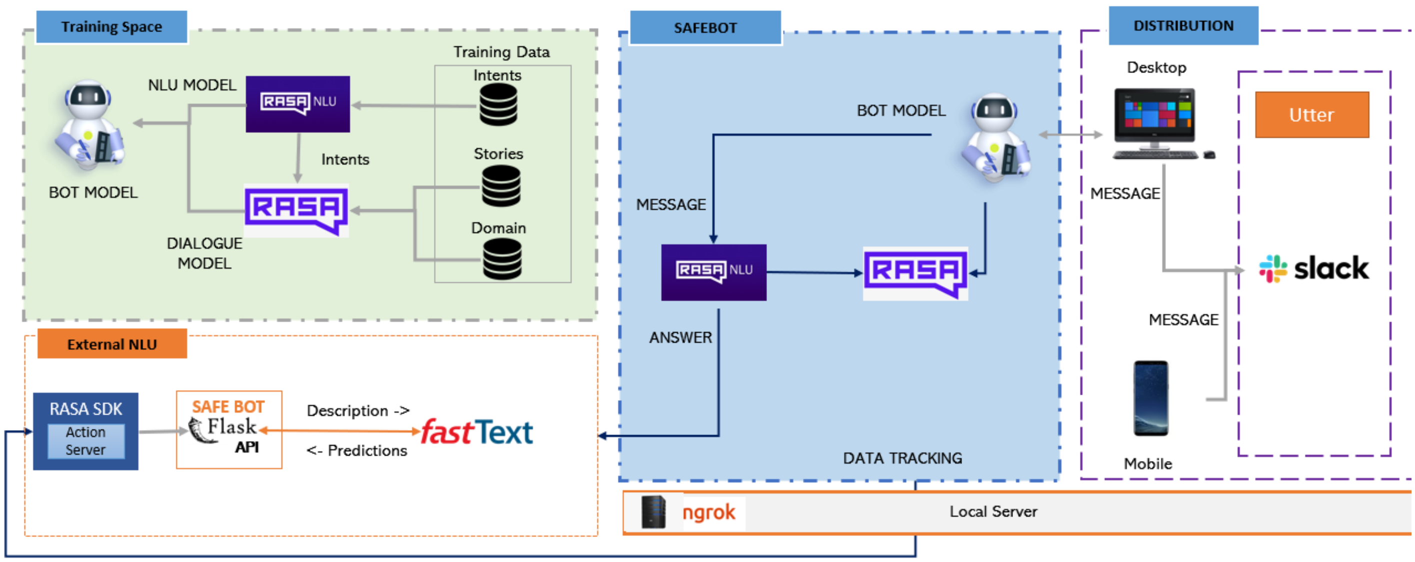
- ✓ **Logistic model** is used to model the probability of a certain class or event existing such as pass/fail, win/lose and alive/dead. Model is also facing overfitting. We are trying more hyperparameter tuning to improve the test accuracy

- ✓ The Objective of a **Linear SVC (Support Vector Classifier)** is to fit to the data you provide, returning a "best fit" hyperplane that divides, or categorizes, your data. From there, after getting the hyperplane, you can then feed some features to your classifier to see what the "predicted" class is. As the model is facing overfitting issue. We are working on more hyperparameter tuning and Data cleaning part.

- ✓ **GBM** is a sequential model. It relies on the intuition that the best possible next model, when combined with previous models, minimizes the overall prediction error. The key idea is to set the target outcomes for this next model to minimize the error. Model build is also facing overfitting. We are also trying more hyperparameter tuning to improve the test accuracy.

- ✓ **Random forests** or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time. For classification tasks, the output of the random forest is the class selected by most trees. Models build on Random Forest is also facing overfitting. We are also trying more hyperparameter tuning to improve the test accuracy.

THE ARCHITECTURE



In Detail :

- We have used a Hybrid NLU based architecture that will learn your daily interactions and enhance it's stories accordingly. In conjunction to that it will also use the custom model we have built in basic python landscape.

EXPLORATION

- The architectural landscape is split in 5 parts :
    - The Training space.
    - The SAFEBOT itself.
    - The distribution.
    - External NLU
    - The local server.

THE TRAINING SPACE

A dialogue modeler has been created with RASA having it's normal architecture on Intents, Stories and Domain. The BOT is configured to response friendly chats and accordingly the intents has been feed to the system. This part is done by training via RASA NLU.

The SAFEBOT

The final version of the BOT design out of RASA has got numerous connection interfaces :

- It copies the model trained by RASA NLU to respond friendly chats.
- It is connected directly to the distribution channel via slack API. It is intercepting the messages via webhooks configured in the SLACK API.
- It is also connected to the interface via the local client.

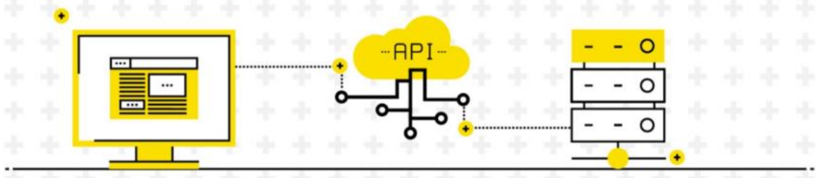
The Distribution

The distribution to the end customer is made via SLACK Channel. We could have used multiple platforms like WhatsApp, messenger on top of RASA but we choose this one being a clustered platform which has its own easy to use configuration along with it's own multi-channel interface which is also platform independent.

External NLU

We have created an API which is exposed to the RASA server and the configuration is handled in actions.py which is a pre-built architecture used inside RASA.

When to Create an API



In general, consider an API if:

- Your data set is large, making download via FTP unwieldy or resource-intensive.
- Your users will need to access your data in real time, such as for display on another website or as part of an application.
- Your data changes or is updated frequently.
- Your users only need access to a part of the data at any one time.
- Your users will need to perform actions other than retrieve data, such as contributing, updating, or deleting data.

In our case we have data dump in the form of a downloadable CSV.

Flask Python is a microframework, which will allow you to have a web application running with very few steps and nearly no setup. This framework's simplicity and strong control over what you want in your application have made it one of the most used web development frameworks.

We used the saved model and exposed it via flask API to the RASA thereby used the trained model for interfacing the questions received by SLACK channel.

The local server

We choose NGROK to be used as a local client server.
ngrok provides a real-time web UI where you can introspect all HTTP traffic running over your tunnels. Replay any request against your tunnel with one click which also helped us to see the connections and also to debug for problem to short out.

THE SNAPSHOT OF THE BOT

As we have used the SLACK platform it helped to be on the same page and make decisions faster by bringing all of your work communication into one place. The Slack features :

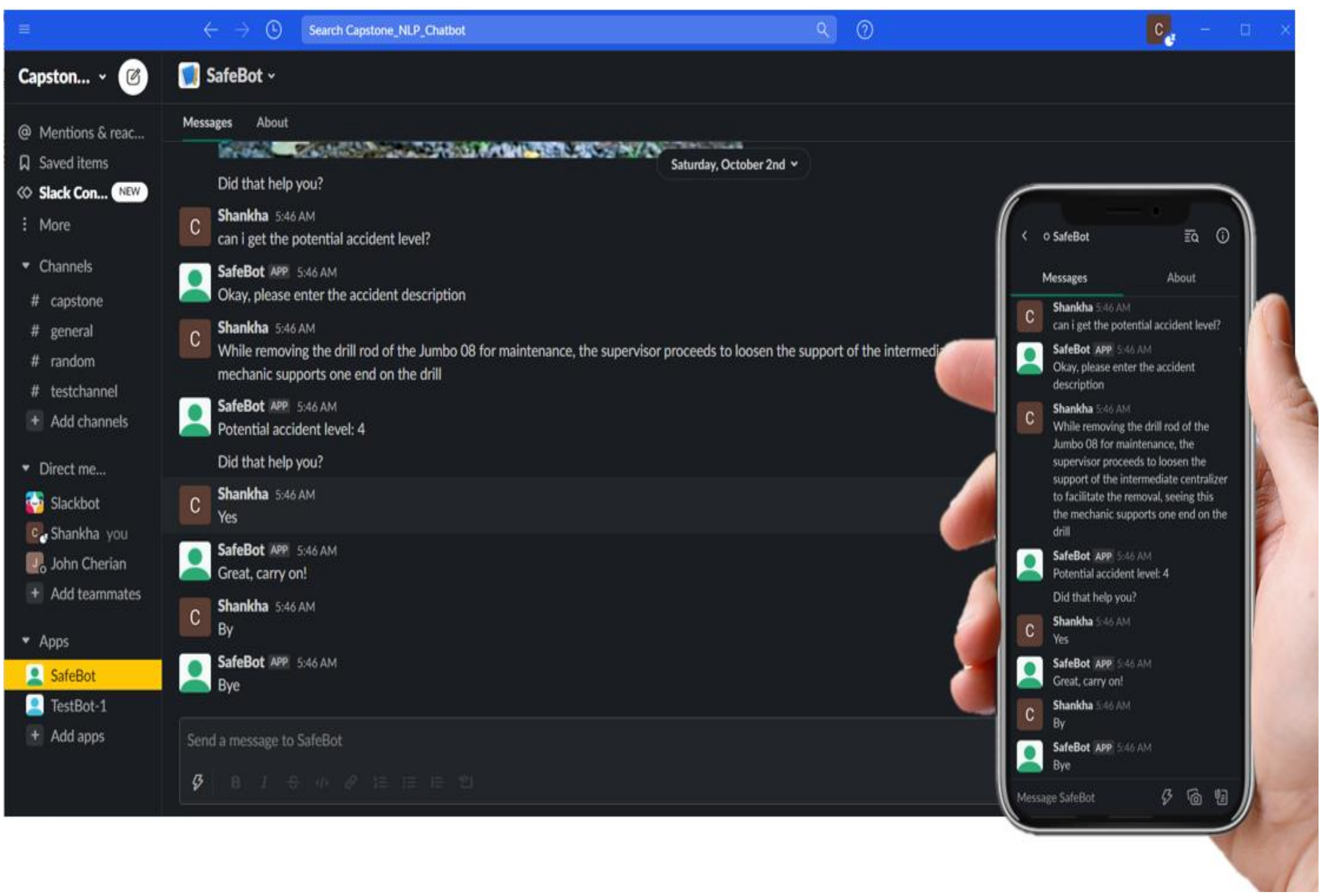
Channels
Stay organized and focused with central spaces for conversations, files, tools and people.

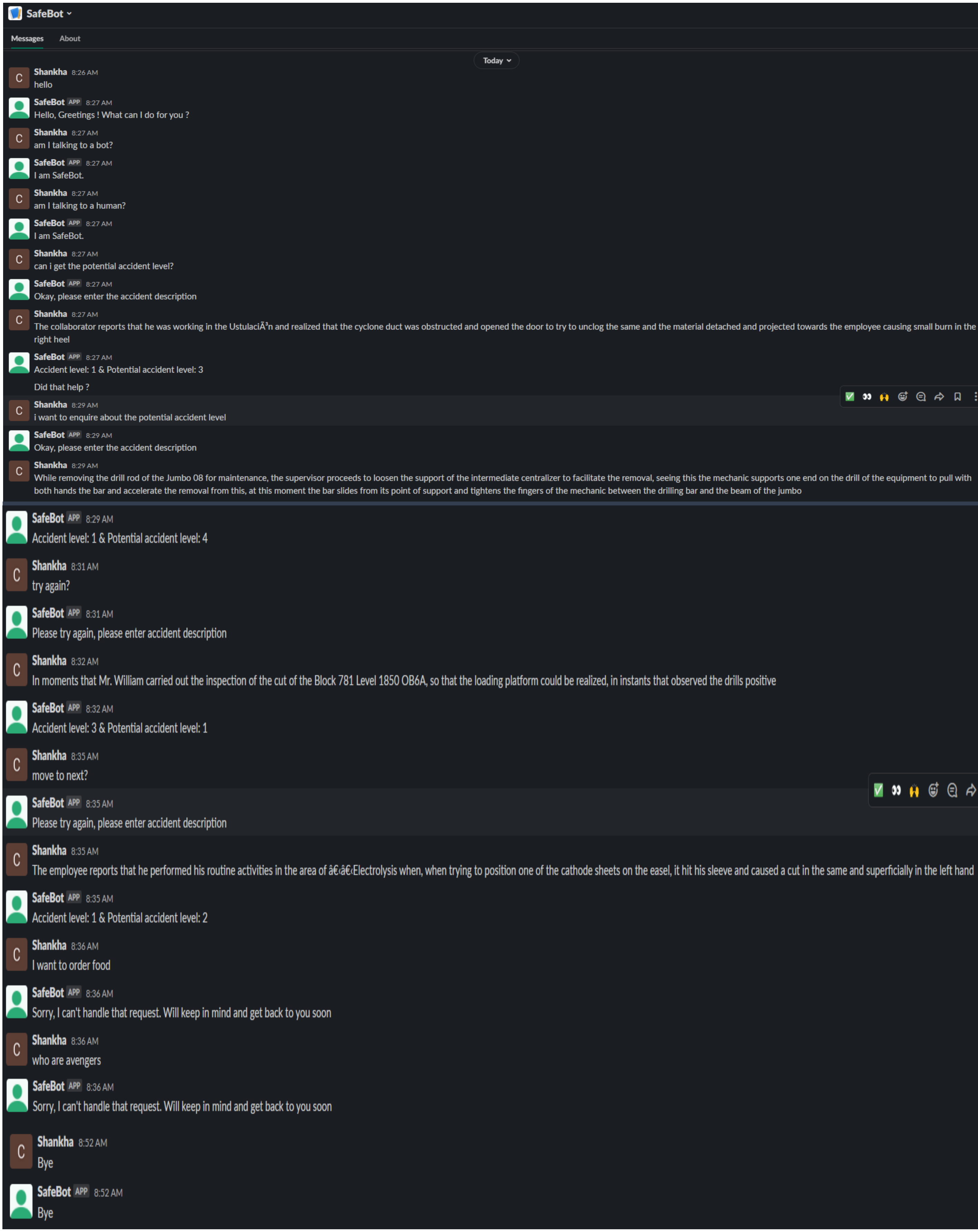
Slack Connect
Collaborate with teams at other companies the same way you do with teams at your own.

Communicate better with your entire team

When you can easily reach people—individually or the whole team—you can work more effectively. Choose the right words to say, and send your message immediately, or schedule it for later.

THE SAFEBOT in action (cross platform) :

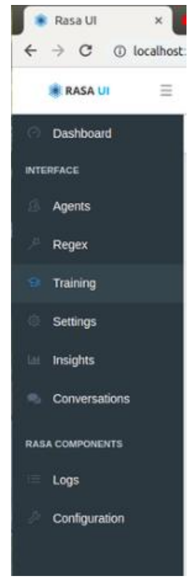



SafeBot
Messages
About
Today
Shankha 8:26 AM
hello
SafeBot APP 8:27 AM
Hello, Greetings ! What can I do for you ?
Shankha 8:27 AM
am I talking to a bot?
SafeBot APP 8:27 AM
I am SafeBot.
Shankha 8:27 AM
am I talking to a human?
SafeBot APP 8:27 AM
I am SafeBot.
Shankha 8:27 AM
can i get the potential accident level?
SafeBot APP 8:27 AM
Okay, please enter the accident description
Shankha 8:27 AM
The collaborator reports that he was working in the UstulaciÃ³n and realized that the cyclone duct was obstructed and opened the door to try to unclog the same and the material detached and projected towards the employee causing small burn in the right heel
SafeBot APP 8:27 AM
Accident level: 1 & Potential accident level: 3
Did that help ?
Shankha 8:29 AM
i want to enquire about the potential accident level
SafeBot APP 8:29 AM
Okay, please enter the accident description
Shankha 8:29 AM
While removing the drill rod of the Jumbo 08 for maintenance, the supervisor proceeds to loosen the support of the intermediate centralizer to facilitate the removal, seeing this the mechanic supports one end on the drill of the equipment to pull with both hands the bar and accelerate the removal from this, at this moment the bar slides from its point of support and tightens the fingers of the mechanic between the drilling bar and the beam of the jumbo
SafeBot APP 8:29 AM
Accident level: 1 & Potential accident level: 4
Shankha 8:31 AM
try again?
SafeBot APP 8:31 AM
Please try again, please enter accident description
Shankha 8:32 AM
In moments that Mr. William carried out the inspection of the cut of the Block 781 Level 1850 OB6A, so that the loading platform could be realized, in instants that observed the drills positive
SafeBot APP 8:32 AM
Accident level: 3 & Potential accident level: 1
Shankha 8:35 AM
move to next?
SafeBot APP 8:35 AM
Please try again, please enter accident description
Shankha 8:35 AM
The employee reports that he performed his routine activities in the area of â€‹â€‹Electrolysis when, when trying to position one of the cathode sheets on the easel, it hit his sleeve and caused a cut in the same and superficially in the left hand
SafeBot APP 8:35 AM
Accident level: 1 & Potential accident level: 2
Shankha 8:36 AM
I want to order food
SafeBot APP 8:36 AM
Sorry, I can't handle that request. Will keep in mind and get back to you soon
Shankha 8:36 AM
who are avengers
SafeBot APP 8:36 AM
Sorry, I can't handle that request. Will keep in mind and get back to you soon
Shankha 8:52 AM
Bye
SafeBot APP 8:52 AM
Bye

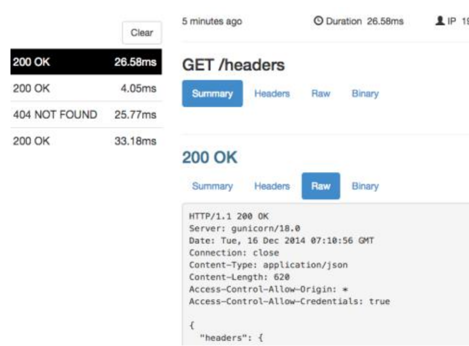
DEPLOYMENT STRATEGY

The complete architecture has been executed from a local test environment. Hence some of the below mentioned action must be performed to be able to make the flow running:

- Setting up local RASA Workspace:
    - RASA has very good docs to help you set the framework. The process is well explained in the below link. https://rasa.com/docs/rasa/installation/
    - Once installed RASA X can be launch to setup you first level of intents.
    - RASA X can be installed in local mode, server mode, by docker compose or by helm chart.
      The guide is there in the below link: https://rasa.com/docs/rasa-x/installation-and-setup/installation-guide
    - RASA has got its own UI once setup is done to be able to handle all the intent, setting an configuration at finger tips.
    - The manual is also there can be done with the help of RASA Docs.
    - Once done the RASA server can be started with command:
      RASA RUN
    - The training needs to be done before and can be done with command
      RASA TRAIN
    - Now this step needs to be done once all the model training is complete and the RASA action and Flash API is ready to be started.



- Setting up the client server (NGOK)
    - NGROK can be found in the below link which can be downloaded and started. https://ngrok.com/download
    - The NGROK runs in CMD environment.
    - Command:
      NGROK HTTP 5005
      Where 5005 is the port on which the RASA server will be running.
    - NGROK has got its own debug window by which you can see the incoming requested received.
    - These will help you to easily debug and find the root cause of the issue if you face a problem understanding the connection.



- Setting up Rasa Action server
    - There is provision in RASA to use external hooks to receive information e.g., plugin external APIs or to perform special actions which is not possible via RASA NLU.
    - We have use external model: FastText to reach our goal along with some intervention.

- Setting up Flash API

  Install FastText

    - This is required as step 5 loads a saved FastText model for prediction

  Install Flash

    - This is required as step 5 opens a port on http://127.0.0.1:5000/ to listen on incoming requests using Flash
    - Set working directory in SafeBot_API_Pred_Multi.py or SafeBot_API_Pred_Multi.ipynb
    - Copy safebot_multi.bin to the working dirctory set in 3
    - Run SafeBot_API_Pred_Multi.py or SafeBot_API_Pred_Multi.ipynb
    - This enables Rasa to pass the accident description by calling http://127.0.0.1:5000/get_p_acc_lvl and this API performs the prediction and returns the Accident level and Potential Accident Level to Rasa through actions.py

- Setting up SLACK

  Slack can be setup using the instructions on the link below https://rasa.com/docs/rasa/connectors/slack/

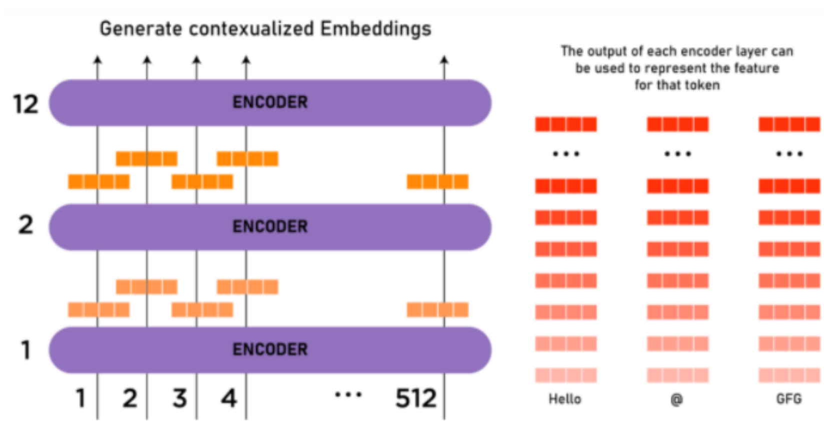
  If you are using the default credentials that was used at demo time, then please make sure you are only updating the 'Receiving Messages' part in the link above, that is the web hook using NGROK.

HOW TO IMPROVE YOUR MODEL PERFORMANCE?

THE CHALLENGE AT HAND IS THAT WE DO NOT HAVE A LARGE DATASET, OUR DATASET HAS ONLY 425 RECORD.

- One of the main reasons for not achieving very high accuracy could be the lack of large labelled text datasets. Most of the labelled text datasets are not big enough to train deep neural networks because these networks have a huge number of parameters and training such networks on small datasets will cause overfitting.

- We are also aware that the NLP models are typically shallower and thus require different fine-tuning methods. BERT (Bidirectional Encoder Representations from Transformers) is a big neural network architecture with Millions of parameters. So, training a BERT model from scratch on a small dataset would result in overfitting.



So, we propose to use a pre-trained BERT model that was trained on a huge dataset, as a starting point and then we can further train the model on our relatively smaller dataset.

- We will be exploring different Fine-Tuning Techniques mentioned below in the weeks to come

- Train the entire architecture – We can further train the entire pre-trained model on our dataset and feed the output to a SoftMax layer. In this case, the error is back-propagated through the entire architecture and the pre-trained weights of the model are updated based on the new dataset.

- Train some layers while freezing others – Another way to use a pre-trained model is to train it partially. What we can do is keep the weights of initial layers of the model frozen while we retrain only the higher layers. We can try and test as to how many layers to be frozen and how many to be trained.

- Freeze the entire architecture – We can even freeze all the layers of the model and attach a few neural network layers of our own and train this new model. Note that the weights of only the attached layers will be updated during model training.

- We will probably use this last approach. We will freeze all the layers of BERT during fine-tuning and append a dense layer and a SoftMax layer to the architecture.

ADJUSTING HYPER PARAMETERS-

- We want to build a model that performs robustly and to this effect, we use the same set of hyper parameters across tasks and validation set. We shall also explore AWD-LSTM language model (Merity et al., 2017a) with an embedding size of 400, 3 layers,1150 hidden activations per layer, and a BPTT batch size of 70. We apply dropout of 0.4 to layers, 0.3 to RNN layers, 0.4 to input embed-ding layers, 0.05 to embedding layers, and weight dropout of 0.5 to the RNN hidden-to-hidden matrix. The classifier has a hidden layer of size 50.

- We use Adam with $\beta_1 = 0.7$ instead of the de-fault $\beta_1 = 0.9$ and $\beta_2 = 0.99$, We use a batch size of 64, a base learning rate of 0.004 and 0.01 for fine tuning the language models and the classifier respectively.

We are hopeful that by adopting these fine-tuning techniques we will be able to achieve high accuracy for the final model that we are going to deploy.

FUTURE WORK AND ENHANCEMENTS

We are also aware that the NLP models are typically shallower and thus require different fine-tuning methods but again its the lack of sufficient labelled text dataset

We have attempted to implement BIRT (Bidirectional Encoder Representations from Transformers),SMOT and Auto tuning with FastText to improve on the model performance.

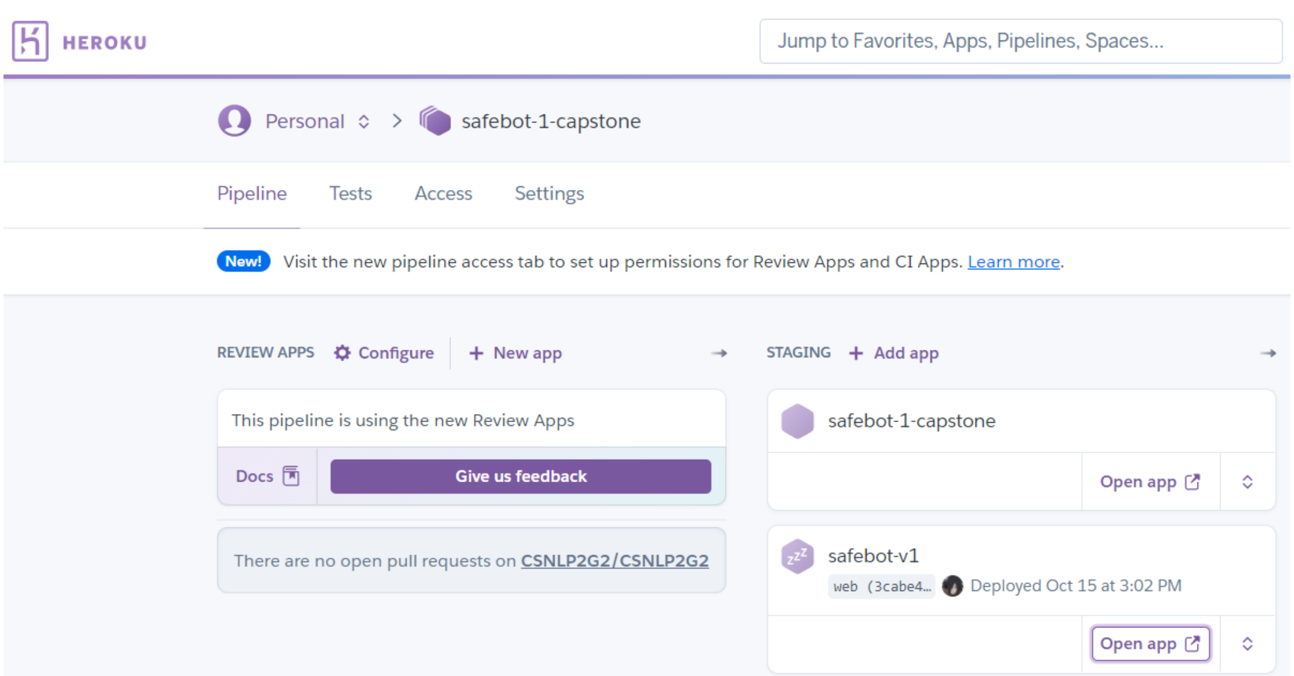
However, we have encountered few road blocks and realised we needed much more time digging deep into the issues we had encountered with different version conflicts and compatibility issues as BIRT. So once these configuration management issues are resolved then we can

freeze the entire layers of BERT during fine tuning and append a dense layer and a Softmax layer to the architecture as its a big neural network architecture with Millions of parameters and training a BERT model from scratch on a small dataset would result in overfitting

We had applied SMOTE but it suffered with over fitting though the accuracies have improved considerably.Further exploration needed applying SMOTE on training data with a 2 step approach i.e over sample the minority class and under sample the majority class by 10% to overcome the problem of over fitting

We had attempted to implement Fast Text Auto Tuning but ended up witnessing infinite process running and crashes, upon reading the community support for the work around we have learnt that some of the wrapper files written in C++ need some changes. Due to lack to sufficient time we couldn't make much headway in this area and this needs further exploration so that Fast Text's hyper parameters auto tuning can be perfectly implemented.

We are in progress of using the cloud platform of HEROKU for our project. This is help us avoid the time limilation and dependency on local server.



LIMITATIONS

As we know sizeable number of observations are critical for any data exploration, analysis and for model building to draw inference and to build intelligence on it regardless of the techniques being deployed.

The challenge at hand was that we do not have a large dataset, We exactly stumbled upon this aspect as we had mere 425 records of dataset with class imbalance and with poor data quality and after all the EDA and balancing techniques being applied we believe its the limited size of data that’s not giving expected better results(>80%) with our models

One big relationship the word2vec model does not allow us to capture is implicit negative signals. These can be especially relevant for modelling a user’s journey. We have created a separate RASA NLU to perform this activity.

CLOSING REFLECTIONS

The Final end-to end model based on fastText architecture gave an aggregated accuracy of 70.00%, and may be suitable for deployment.

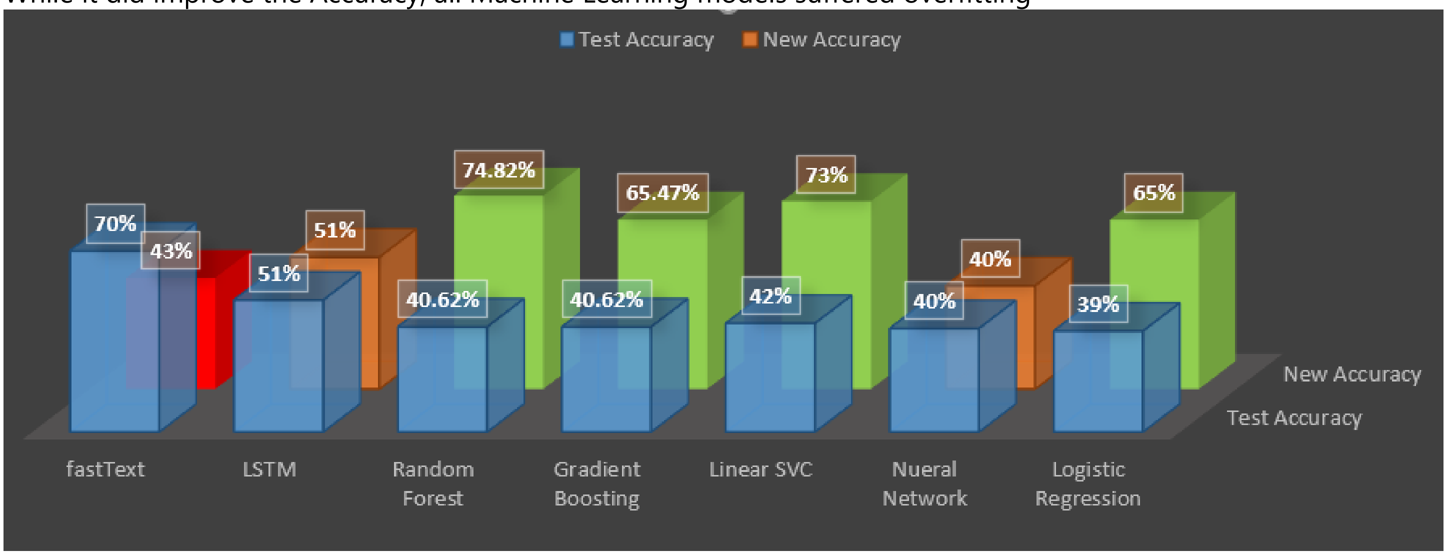
For the design NLP based potential accident level prediction, exploration of classical machine learning models (SVM, Light GBM, LSTM and others), may be used to develop the model that would be effective for problems of both regression and classification.

The challenge at hand is that we do not have a large dataset, Our dataset has only 425 records

One of the main reasons for not achieving very high accuracy could be the lack of large labeled text datasets. Most of the labeled

- text datasets are not big enough to train deep neural networks because these networks have a huge number of parameters
- training such networks on small datasets will cause overfitting.
-

However we did impliment SMOT(synthetic minority oversampling technique) and below were the results.
While it did improve the Accuracy, all Machine Learning models suffered overfitting



Classification report-

```
      precision recall f1-score   support
1      0.97        0.74        0.84        38
2      0.40        0.57        0.47        21
3      0.52        0.59        0.55        29
4      0.48        0.44        0.46        27
5      1.00        0.92        0.96        24
accuracy 0.65 139
macro avg 0.67 0.65 0.65 139
weighted avg 0.70 0.65 0.67 139
```

We are also aware that the NLP models are typically shallower and thus require different fine-tuning methods
We have also examined BERT but couldn't make much headway as we have encountered issues such as packages incompatibility.

Version conflicts and few methods of the source files still not stabilized and in view of the above and in the interest of time we have parked BERT for future research.

We have evaluated with FastText's auto Hyper parameters tuning and realized it not only needs higher compute capability as our

- 16 GB ram systems were crashing and on Collab ran eternally and upon reaching community support as have realized that some
- of the wrappers written in C++ were still not stabilized such as 'autotune.cc' wrapper file. Owing to these challenges we have
- parked above approached for future research.

REFERENCES

Sl no.	Topic	Link
1	**RASA Installation**	https://rasa.com/docs/rasa-x/installation-and-setup/installation-guide/
2	**RASA Slack INtegration**	https://rasa.com/docs/rasa/connectors/slack/
3	**HEROKU deplyment**	https://blog.heroku.com/how-to-deploy-your-slack-bots-to-heroku
4	**Local client server setup**	https://ngrok.com/docs
5		https://fasttext.cc/docs/en/support.html
6	**Creating WEB API**	https://programminghistorian.org/en/lessons/creating-apis-with-python-and-flask
		https://towardsdatascience.com/creating-restful-apis-using-flask-and-python-655bad51b24
7	**Training RASA NLU**	https://rasa.com/docs/rasa/tuning-your-model/
8	**Analytics inference**	https://www.kaggle.com/ihmstefanini/industrial-safety-and-health-analytics-database
9	**Business inference for situational analysis**	http://eolss.net/Sample-Chapters/C16/E1-58-07.pdf
		https://en.wikipedia.org/wiki/Industry_in_Brazil
10	**BI Solutions Study**	https://rasa.com/blog/building-bots-with-rasa-shiba-slack-hosted-interface-for-business-analytics/

GITHUB Repo	**https://github.com/CSNLP2G2/CSNLP2G2.git**
Drive Contents (Work Elements)	**https://drive.google.com/drive/folders/1g9Xy9Z0WyjuHOU6n8QtdWxhS2nPRo8cH?usp=sharing**
SLACK Creds	**User : chatbot.cs.nlp2g2@gmail.com Pass : ChatBotCSNLP2G2**