Tab 1

PGP AIML Feb 24 A - Group 7

**CAPSTONE PROJECT - Interim Report**

NATURAL LANGUAGE PROCESSING

INDUSTRIAL SAFETY - NLP BASED CHATBOT

horizontal line

# Placeholder image

# 

# TEAM DETAILS

| **Mentor**: ROHIT GUPTA | |
| --- | --- |
| NAVEEN JOTHI  MOHIT JAIN RAHUL PATERIA  AKASH KUMAR  SOHIT SINGH  PRATIK PATIL | [naveenjothi040@gmail.com](mailto:naveenjothi040@gmail.com)  [mohit\_jain@yahoo.com](mailto:mohit_jain@yahoo.com)  [rahulpateriya121@gmail.com](mailto:rahulpateriya121@gmail.com)  [akki.vik3@gmail.com](mailto:akki.vik3@gmail.com)  [Singh.sohit46@gmail.com](mailto:Singh.sohit46@gmail.com)  [pppratikpatil723@gmail.com](mailto:pppratikpatil723@gmail.com) |

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

[**TEAM DETAILS 2**](#_gjdgxs)

[**PROBLEM STATEMENT 6**](#_arolcxe0i15c)

[DATA DESCRIPTION 6](#_la5jp5tnimjw)

[**PROJECT OBJECTIVE 7**](#_4sfysv2zp72)

[**EXPLORATORY DATA ANALYSIS 7**](#_dt16u8sqi2rl)

[INITIAL REVIEW 7](#_suygn8zzl2p)

[HANDLING DUPLICATES AND OTHER DATA CLEANING STEPS 9](#_ma0abj7on5s5)

[ANALYSING PATTERNS 12](#_f8v1xtd5ja6q)

[OBSERVATIONS 13](#_a6ya9b49m645)

[VISUALIZATION 14](#_t3sldqejev78)

[UNIVARIATE ANALYSIS 14](#_ic05hbetxeqt)

[KEY INSIGHTS - Univariate Analysis 17](#_9fs7evvfko04)

[1. Geographical Distribution 17](#_hpclbhc5b2d6)

[2. Industry Sectors 18](#_l1vksfu8osro)

[3. Accident Severity 18](#_erqk9rhayz67)

[4. Demographics 18](#_4pohq7uud0g)

[5. Critical Risks 18](#_vll4eku3fj57)

[6. Temporal Trends 19](#_tgn5qh4fa7uy)

[RECOMMENDATIONS 19](#_vxxjkst55j1t)

[BIVARIATE ANALYSIS 19](#_rjw0eopizwhx)

[Correlation between ‘Accident Level’ and ‘Potential Accident Level’ 19](#_23fabmwv834z)

[Correlation interpretation and implications 20](#_5l912kg2twbb)

[Trend of accidents by year-month by ‘Potential Accident Level’ 24](#_ikha6h2qd6my)

[KEY INSIGHTS - Bivariate Analysis 26](#_39vgytmdj8a8)

[Potential Accident Level by Industry Sector 26](#_hog0zfn2m5iq)

[Potential Accident Level by Year 26](#_uh4urh951xau)

[Potential Accident Level by Critical Risk 26](#_98d81kb6d8y9)

[General Observations 26](#_s5krr392umz)

[MULTIVARIATE ANALYSIS 27](#_xgr2r4z9oxcj)

[KEY INSIGHTS - Multivariate Analysis 30](#_e0ldtyu2fx0s)

[Accident Level vs. Potential Accident Level by Industry Sector 30](#_olbgn3kcgt1h)

[Accident Level by Local and Country 30](#_ubdj1ehtta33)

[Critical Risks vs. Accident Level by Gender 30](#_kh74fd76fk85)

[Year-Month Trends of Accident Levels by Industry Sector 30](#_jymvx5m0gajk)

[General Observations 31](#_rtyk482al12l)

[RECOMMENDATIONS 31](#_6wlsrg5vebhy)

[**NLP - PREPROCESSING 32**](#_fyma60c4kb2p)

[TRANSLATION 32](#_d7bp6bylu4qi)

[CLEANUP STEPS 34](#_j1llwp5jlboc)

[OBSERVATIONS 36](#_5t01nb599veu)

[**VISUALIZING THE DATA (NLP) 37**](#_jznqubk3v8js)

[WORD FREQUENCY DISTRIBUTION 37](#_ohyo8ug9322m)

[OBSERVATIONS 37](#_pps5b8mfa3v5)

[DISTRIBUTION OF TEXT LENGTH 39](#_pbw32w4ekb28)

[OBSERVATIONS 39](#_g9kc1s8my78k)

[RECOMMENDATIONS 40](#_7lrabftprexk)

[TOP N MOST COMMON WORDS 40](#_dicepm4wogxh)

[OBSERVATIONS 41](#_ad85t5w43ztk)

[N-GRAMS 42](#_pxhckk19a9rn)

[UNI-GRAMS 42](#_cpfiazmd3ag5)

[BI-GRAMS 43](#_n3tr1xp2nt9j)

[TRI-GRAMS 44](#_uwdsjka06i2h)

[OBSERVATIONS FROM N-GRAMS: 44](#_qfnw40r399wz)

[N-GRAMS BY INDUSTRY SECTORS 45](#_ikpzuxjsege9)

[OBSERVATIONS AND INSIGHTS: 46](#_9b6c4oc3j5ye)

[POS Tagging 46](#_m9k4g3hdsbpj)

[OBSERVATIONS AND INSIGHTS: 47](#_mqm6lp5bultf)

[**EXPORTED TO EXCEL FORMAT 48**](#_xafbwis2st77)

[**DECIDING MODELS AND MODEL BUILDING 48**](#_8dlsdxuoj69k)

[CHECK FOR IMBALANCES 48](#_chv4o1puvnaq)

[OBSERVATIONS AND INSIGHTS 49](#_u54dqmspx6k)

[MERGING POTENTIAL ACCIDENT LEVEL VI WITH V 49](#_nmjpaj4gc0e)

[OBSERVATIONS FROM THE PLOT AND DISTRIBUTION 50](#_td1mq4jpp7rd)

[ADDRESSING IMBALANCE IN THE TARGET VARIABLE "POTENTIAL ACCIDENT LEVEL" 51](#_oknpsmp1lj4h)

[DATA PREPARATION AND FEATURE ENGINEERING 51](#_f73ywb1ozf70)

[COMBINING FEATURES 53](#_q6u1t3vo82p9)

[RESAMPLING TECHNIQUES TO HANDLE IMBALANCE 55](#_34osjj95wbh8)

[MODEL BUILDING 58](#_ik0l5zjnsgvv)

[MODEL BUILDING AND EVALUATION 58](#_std29d5tzn54)

[12. With GloVe Embeddings: 59](#_vokwsjm04ugj)

[Insights from Model Performance with GloVe Embeddings: 60](#_gxcdo36ejjue)

[13. With Word2Vec Embeddings: 61](#_ooz1tw1jdbyk)

[Insights from Model Performance with Word2Vec Embeddings: 62](#_x0lt192xor4g)

[14. With TF-IDF Embeddings: 63](#_x3y3a2qid25n)

[Insights from Model Performance with TF-IDF Embeddings: 64](#_qn0h7b74oleo)

[HYPERPARAMETER TUNING 65](#_h6ily5g7cj75)

[MODELS SELECTED FOR HYPERPARAMETER TUNING 65](#_rle2bxekxvcr)

[FITTING THE MODEL WITH GRIDSEARCH WITH BEST PARAMETERS FOR THE SELECTED MODELS. 67](#_412ma89vyx16)

[SELECTION OF THE BEST EMBEDDING TECHNIQUE 68](#_t5p25b3ijnla)

[SYNONYM REPLACEMENT FOR DATA AUGMENTATION 71](#_wy0x0i75vjkr)

[RETRAINING THE DATA 73](#_ejjmu3jjax25)

[**KEY INSIGHTS AFTER DATA AUGMENTATION 76**](#_v5kfbr798gk4)

[**DESIGN AND TRAIN NEURAL NETWORK CLASSIFIERS 77**](#_yxb8a1dlmeq5)

[ANN CLASSIFICATION NETWORK 77](#_9mrj7ydlbizp)

[MODEL WITH AUGMENTED AND GLOVE DATAFRAME 78](#_szxnaotbnppb)

[INSIGHTS FROM NEURAL NETWORK MODEL WITH AUGMENTED GLOVE DATAFRAME: 80](#_6r31wytdvj4y)

[MULTI CLASS CLASSIFICATION WITH AUGMENTED AND COMBINED GLOVE FEATURED DATAFRAME 81](#_hxmxycowmp5s)

[INSIGHTS FROM MULTI-CLASS CLASSIFICATION WITH AUGMENTED GLOVE FEATURES: 84](#_55wjit2kivc3)

[MULTI CLASS CLASSIFICATION WITH GLOVE FEATURES FROM ACCIDENT DESCRIPTION 85](#_u08kp2scbcxg)

[INSIGHTS FROM MULTI-CLASS CLASSIFICATION WITH GLOVE FEATURES: 89](#_ahy198ti6d5g)

[**DESIGN, TRAIN AND TEST LSTM OR RNN CLASSIFIER 89**](#_77w8bajnxfuq)

[LSTM MODEL WITH TEXT INPUTS ONLY 90](#_86a0i3ae1fyh)

[INSIGHTS FROM TEXT INPUT ONLY - LSTM 95](#_msoh8xfy17rj)

[LSTM WITH COMBINED CATEGORICAL AND GLOVE FEATURES DATAFRAME 97](#_3rs8uggcdl73)

[INSIGHTS FROM HYBRID LSTM MODEL 102](#_q9oepdchluj3)

# 

# PROBLEM STATEMENT

The database comes from one of the biggest industries in Brazil and the world. Industries and companies around the globe urgently need to understand why employees still suffer injuries and accidents in plants. Sometimes, they also die in such an environment.

## DATA DESCRIPTION

This database is basically records of accidents from 12 different plants in 03 different countries where every line in the data is an occurrence of an accident.

**Columns description:**

* Data: timestamp or time/date information
* Countries: which country the accident occurred (anonymised)
* Local: the city where the manufacturing plant is located (anonymised)
* Industry sector: which sector the plant belongs to
* Accident level: from I to VI, it registers how severe was the accident (I means not severe but VI means very severe)
* Potential Accident Level: Depending on the Accident Level, the database also registers how severe the accident could have been (due to other factors
* involved in the accident)
* Genre: if the person is male of female
* Employee or Third Party: if the injured person is an employee or a third party
* Critical Risk: some description of the risk involved in the accident
* Description: Detailed description of how the accident happened.

# PROJECT OBJECTIVE

Design a ML/DL based chatbot utility which can help the professionals to highlight the safety risk as per the incident description.

# EXPLORATORY DATA ANALYSIS

## INITIAL REVIEW

After importing the necessary libraries and loading the data, we analyzed the data's structure and significance, taking appropriate actions based on our observations.

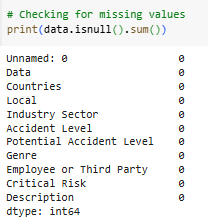


**Insights gained on dataset:**

The dataset is relatively small (425 rows x 11 columns) but contains relevant information.

It is important to note that some components of the dataset were anonymized to conceal the names and locations of the facilities.

**ANALYSING AND HANDLING MISSING/NULL VALUES**



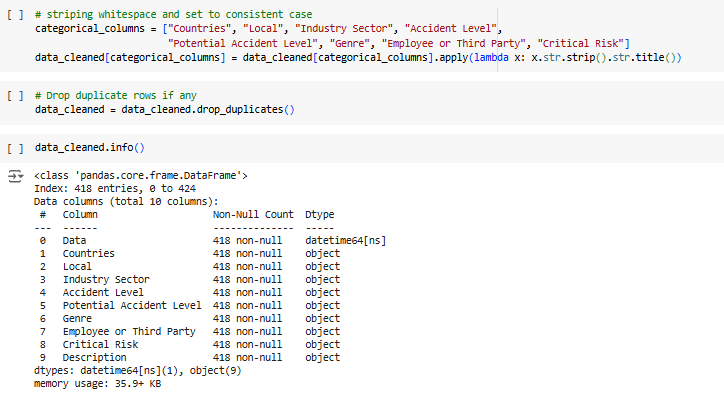


There are no missing values in the dataset. An unnecessary column named "Unnamed" was removed due to the lack of related metadata, as it added no value to the analysis.

## HANDLING DUPLICATES AND OTHER DATA CLEANING STEPS

We observe duplicate records in the given dataset, hence we drop the duplicates.

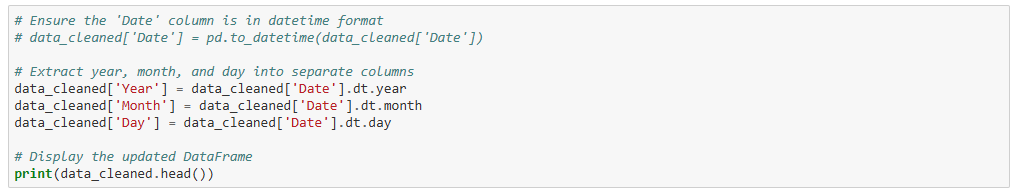
Initial dataset had 425 rows as observed in the previous step, after dropping 7 duplicate rows, we are left with 418 rows.



We noticed that the dataset contains two column names, "Data" and "Genre," in Portuguese. To ensure consistency in the language across the dataset, we are converting these columns to English.

## 

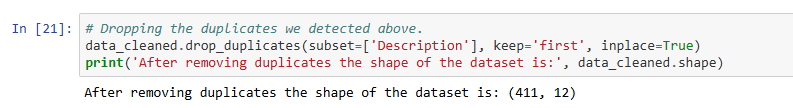
We are extracting the day, month, and year from the date column in order to conduct exploratory data analysis (EDA) in the subsequent steps. This will help us understand the distribution and identify patterns related to the days and months when incidents occurred.



Since we have already extracted the date features into individual columns, the original "Date" feature is no longer necessary and will be dropped.

## 

Additionally, we observe that the data frame contains 7 rows where only one or two column values differ, while the "Description" column remains the same across these rows. This inconsistency is logically incorrect, so we have decided to drop these rows as well.

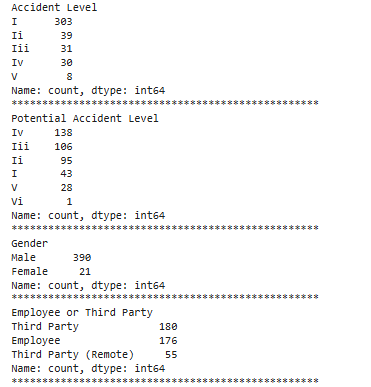


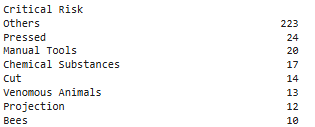
## 

## ANALYSING PATTERNS

Analyzing the occurrence patterns across categorical data within the dataset to identify key trends and insights.









### OBSERVATIONS

* Most of the incidents (over 50%) occurred in Country\_01
* Most of the incidents from Country\_01 has been reported from Local\_03 city
* Most incidents reported are least severe (Level I)
* Most incidents occurred are to male which signifies it's a male dominated industry
* "Others" is the largest category (223 incidents), suggesting the need for clearer risk definitions.

## 

## 

## 

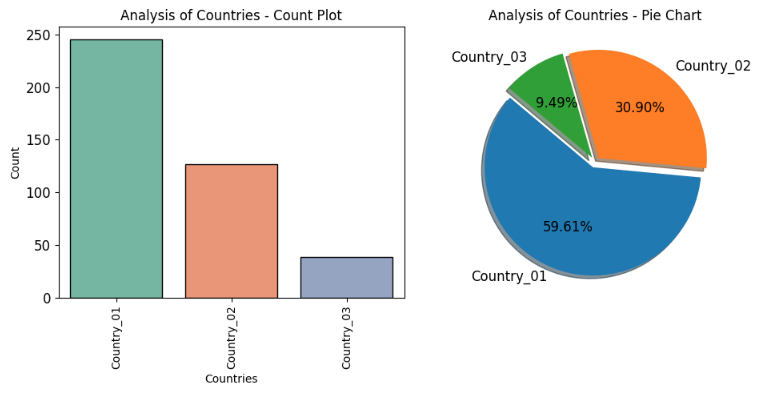
## 

## 

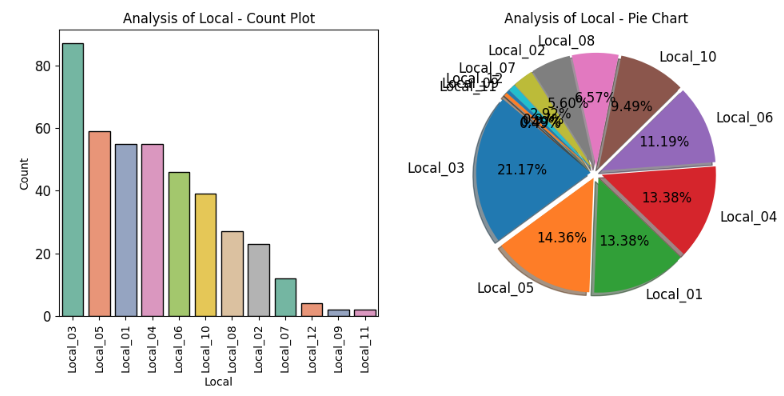
## VISUALIZATION

### UNIVARIATE ANALYSIS

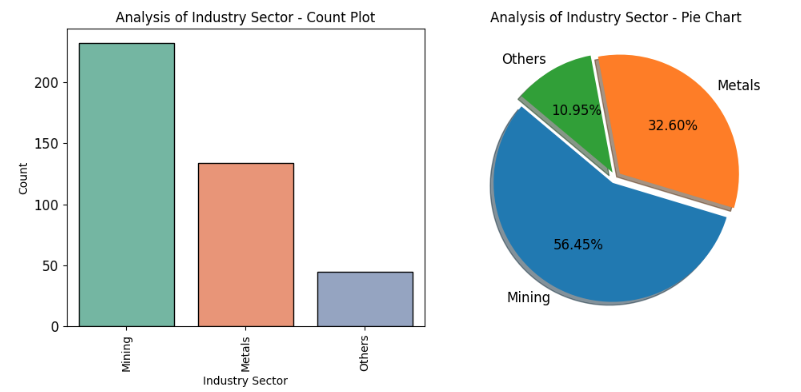
Incident occurrences by countries



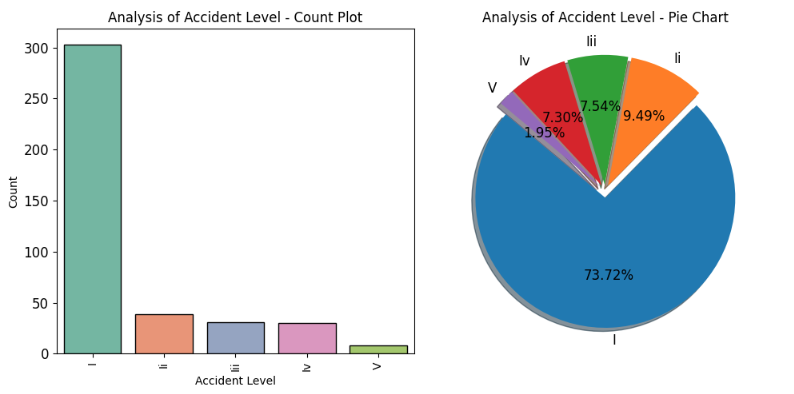
Incident occurrences by local (City)



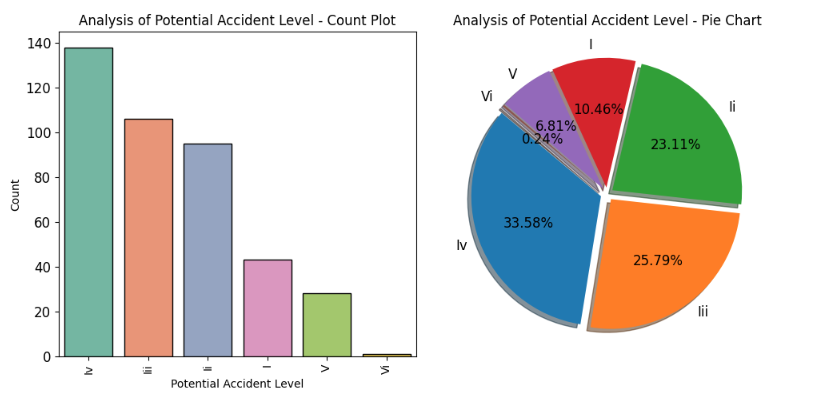
Incident occurrences by industry sector



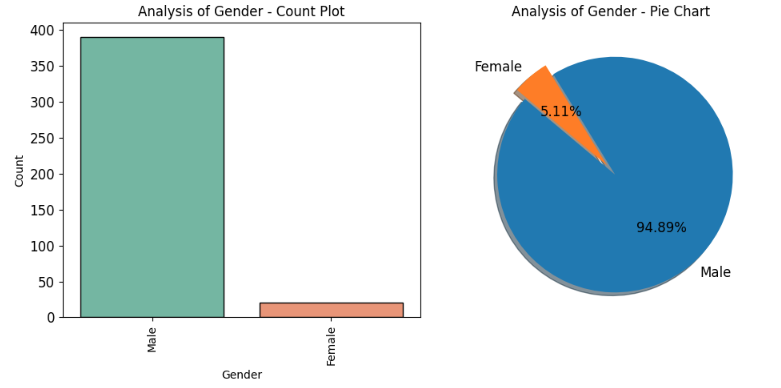
Incident occurrences by accident level



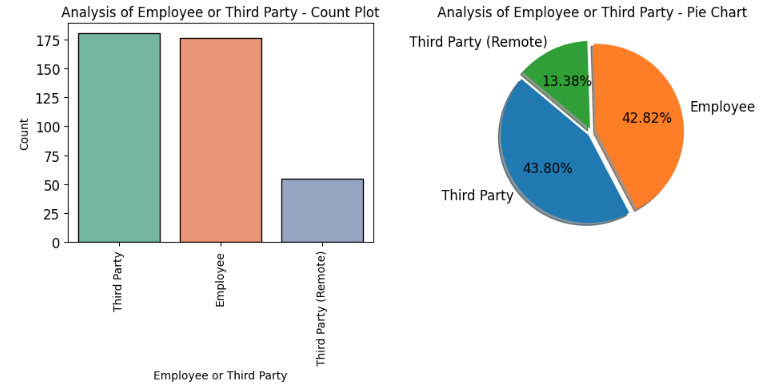
Incident occurrences by potential accident level



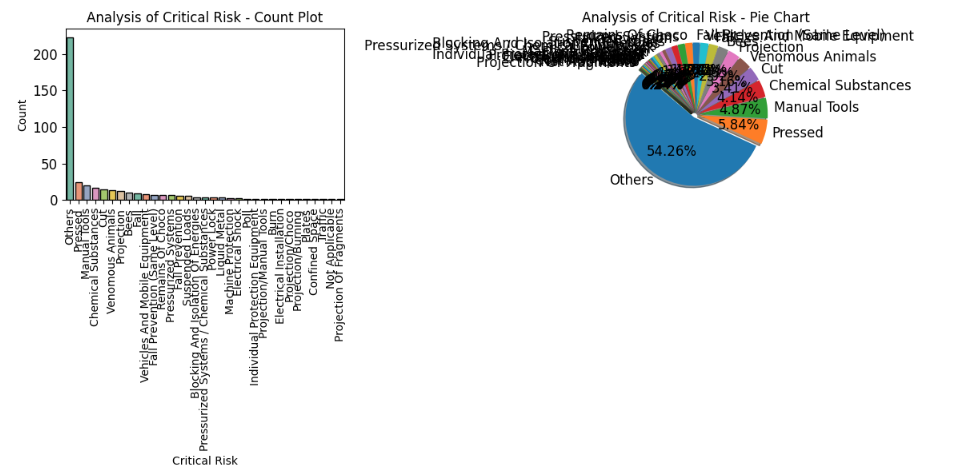
Incident occurrences by gender



Incident occurrences by employment type



Incident occurrences by critical risk



#### KEY INSIGHTS - Univariate Analysis

##### 1. Geographical Distribution

* **Countries**:
  + Country\_01 accounts for a significant majority of incidents (245), comprising over half the dataset.
  + Country\_02 and Country\_03 have substantially fewer incidents (127 and 39, respectively).
* **Local Areas**:
  + Local\_03 stands out with 87 incidents, making it the most accident-prone area.
  + Local\_05, Local\_01, and Local\_04 follow closely with similar counts, suggesting these locations require focused attention.

##### 2. Industry Sectors

* **Mining** is the leading sector with 232 incidents, emphasizing its high-risk nature.
* **Metals** contribute to 134 incidents, indicating it is also a critical area for safety interventions.
* Other sectors account for a relatively small proportion (45).

##### 3. Accident Severity

* **Accident Level**:
  + Level I accidents dominate the dataset (303 incidents), signifying that minor accidents are the most common.
  + Higher-severity accidents (Levels IV and V) are less frequent but still significant for targeted safety measures.
* **Potential Accident Level**:
  + Potential Level IV accidents (138) and Level III (106) highlight areas of latent high-risk scenarios.

##### 4. Demographics

* **Gender**:
  + Males represent a staggering 95% of incidents, suggesting male-dominated roles in these industries may face greater exposure to risks.
* **Employee vs. Third Party**:
  + Third-party individuals (including remote third parties) are involved in more incidents (235) compared to employees (176), indicating external personnel face considerable safety challenges.

##### 5. Critical Risks

* **Top Risk Factors**:
  + "Others" (223 incidents) may require further investigation to identify underlying risk contributors.
  + Pressed (24), Manual Tools (20), and Chemical Substances (17) are notable specific risks.

##### 6. Temporal Trends

* **Yearly Data**:
  + A majority of accidents occurred in 2016 (278 incidents), compared to 2017 (133), indicating a declining trend.
* **Monthly Data**:
  + February (60) and April (51) saw the highest number of incidents, suggesting seasonality effects or operational peaks.

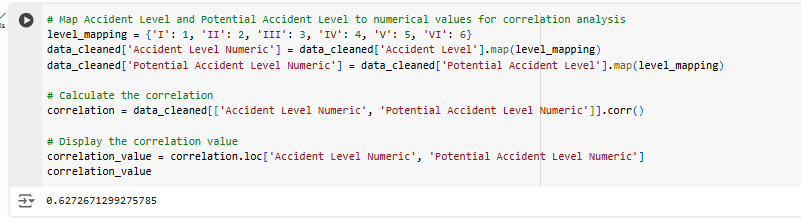
#### RECOMMENDATIONS

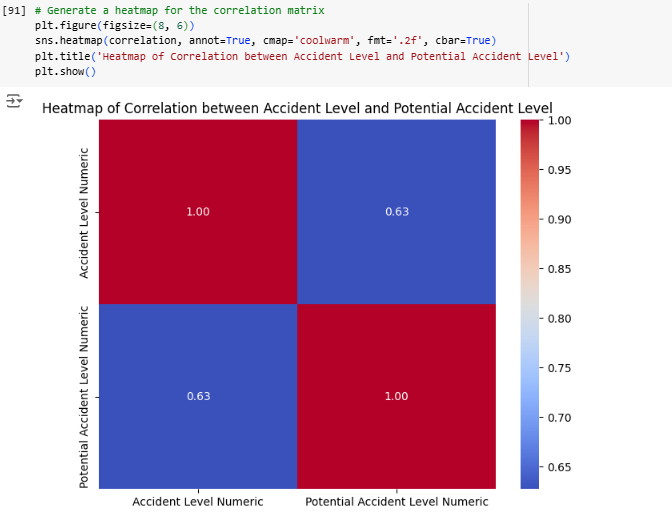
* Focus safety interventions on Country\_01 and Local\_03.
* Address critical risks like Pressed, Manual Tools, and Chemical Substances.
* Enhance safety measures in Mining and Metals industries.
* Develop targeted safety programs for males and third-party workers.
* Investigate incident spikes in February and April for potential seasonal or operational causes.most incidents (278).

### 

### BIVARIATE ANALYSIS

#### Correlation between ‘Accident Level’ and ‘Potential Accident Level’



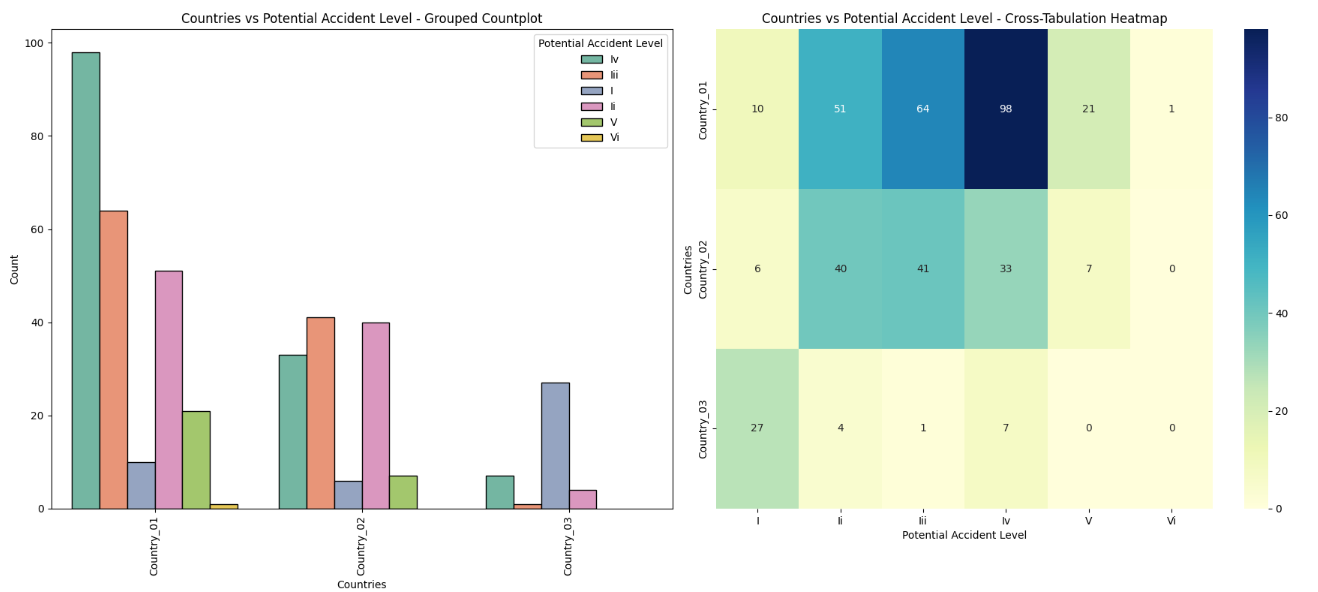


A correlation of **0.627** between Accident Level and Potential Accident Level indicates a **moderately strong positive relationship** between these two variables. Here's what this means in context:

#### Correlation interpretation and implications

1. **Positive Relationship**:
   * As the **Accident Level** increases (indicating more severe actual accidents), the **Potential Accident Level** also tends to increase (indicating higher potential severity of the accident).
2. **Moderately Strong Correlation**:
   * The value of **0.627** suggests that while there is a significant relationship, it is not perfect. Other factors may also influence the Potential Accident Level, apart from the Accident Level.
3. **Practical Implication**:
   * This correlation implies that high-severity accidents are often associated with high potential risks, indicating a need to prioritize preventive measures for incidents with high potential severity to avoid severe actual outcomes.

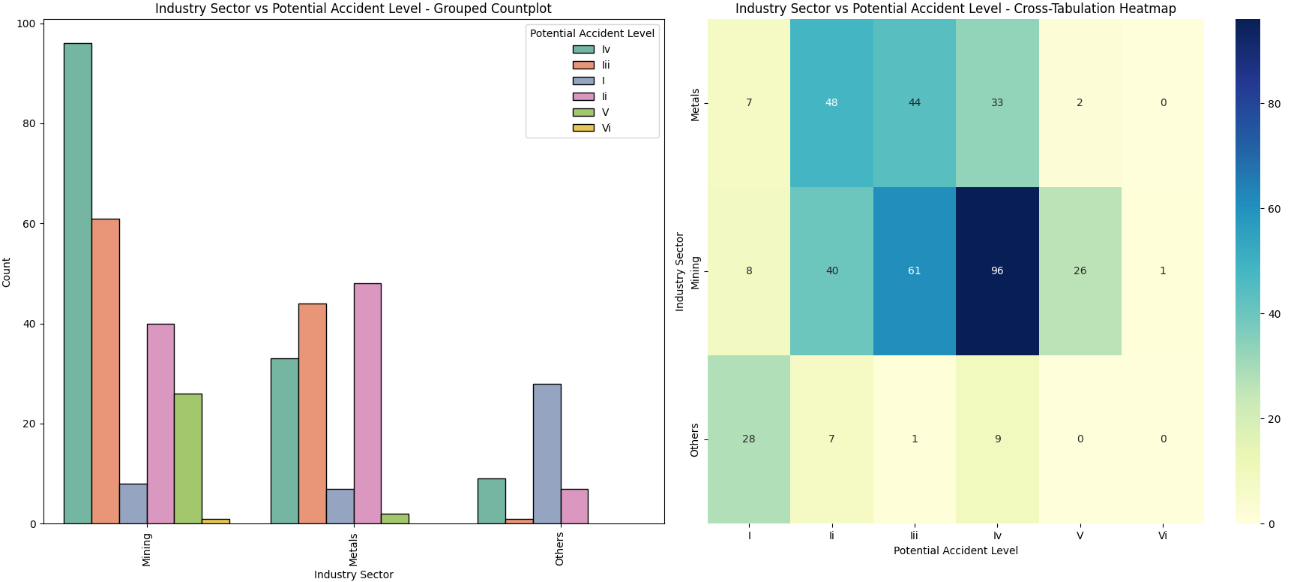
Selecting the ‘**Potential Accident Level**’ as the **target variable** performing the bivariate analysis against all the other categorical columns.

‘Potential Accident Level’ by ‘Countries’:

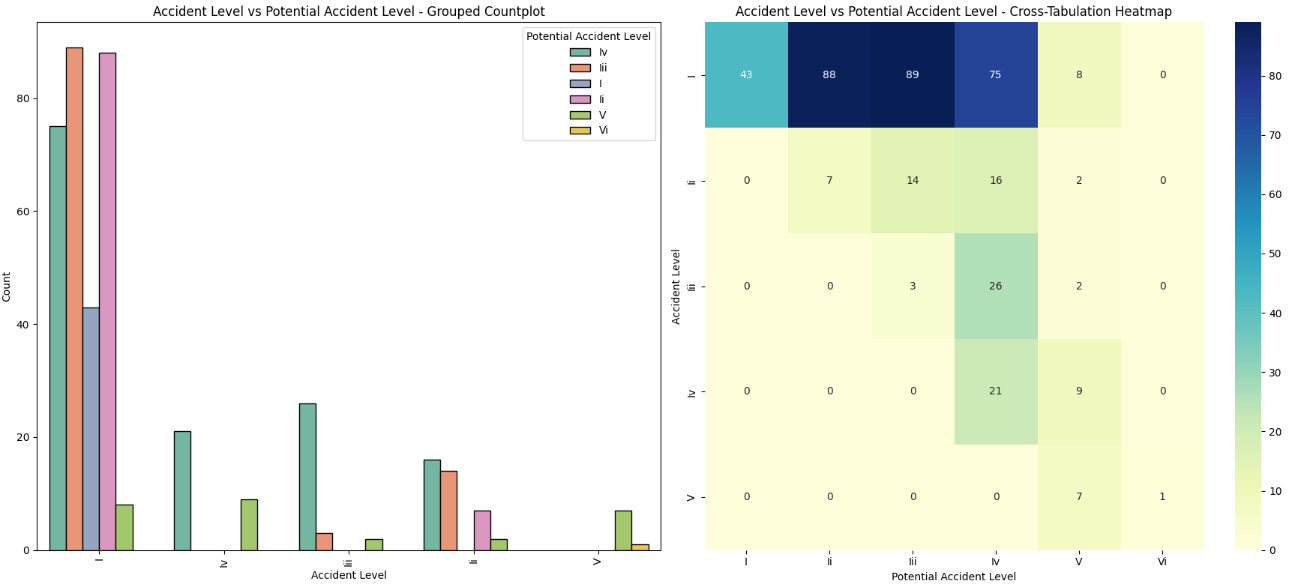
‘Potential Accident Level’ by ‘Local’ (City):



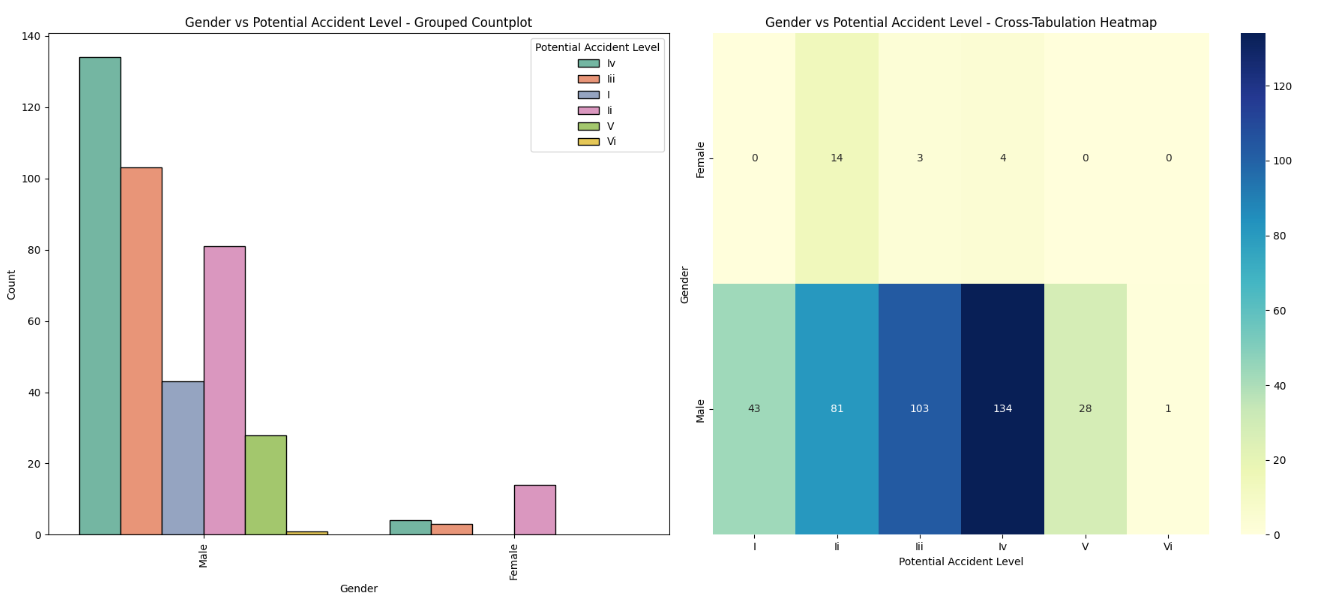
‘Potential Accident Level’ by ‘Industry Sector’:



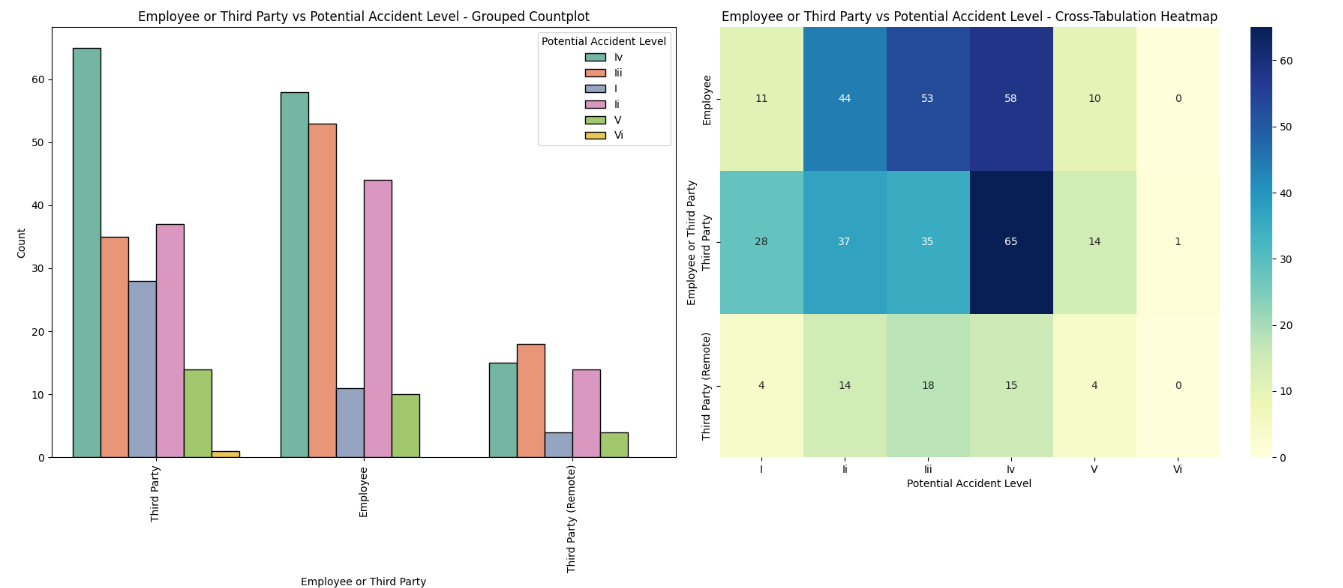
‘Potential Accident Level’ by ‘Accident Level’:



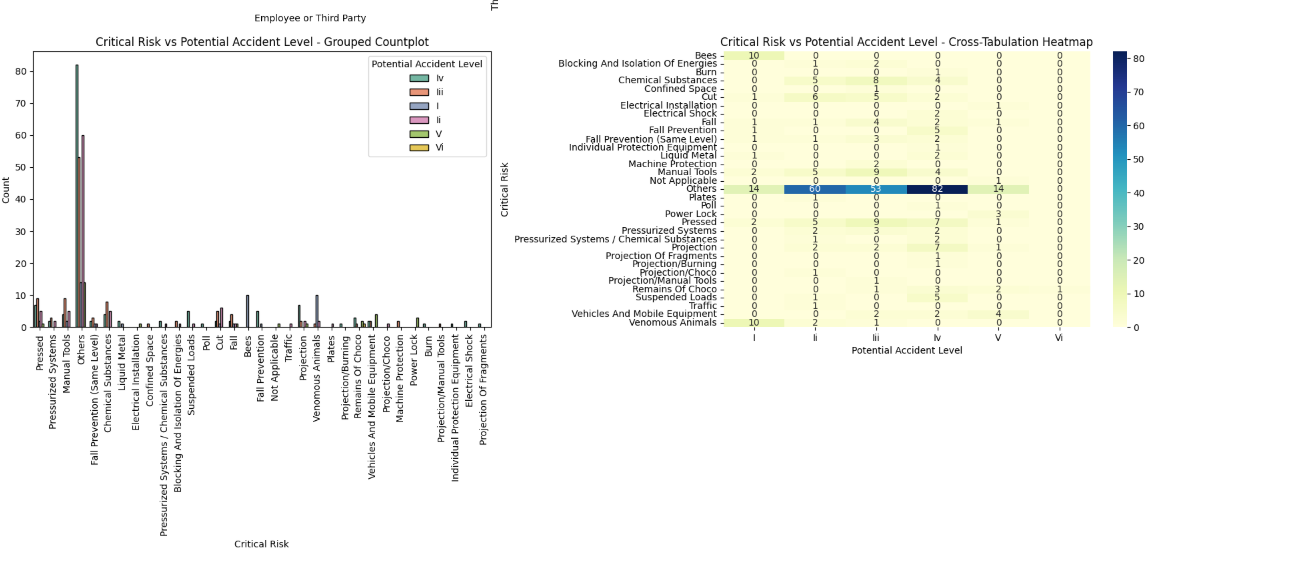
‘Potential Accident Level’ by ‘Gender’:



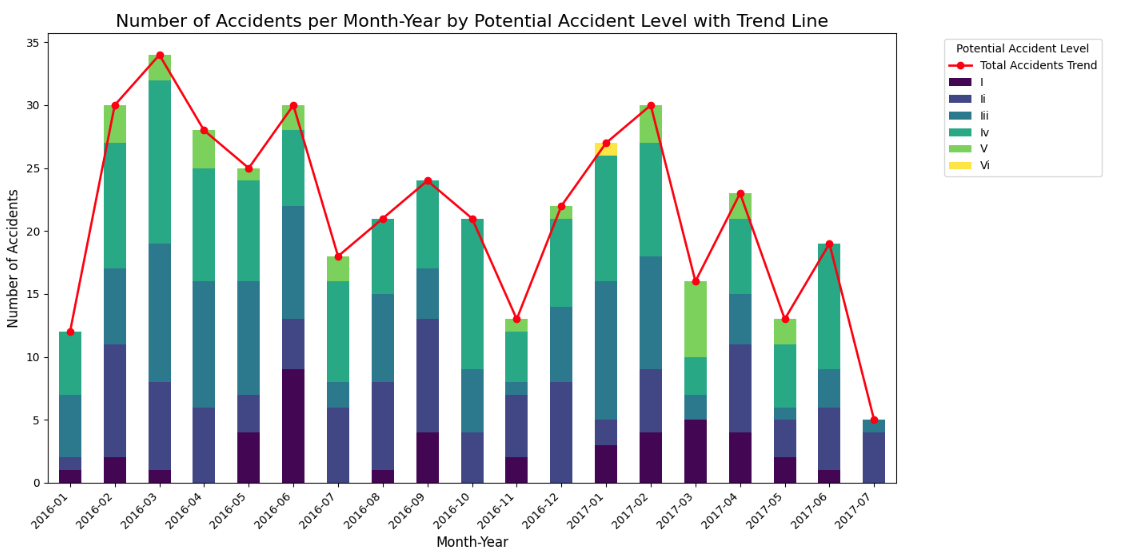
‘Potential Accident Level’ by Employment Type:



‘Potential Accident Level’ by ‘Critical Risk’:



#### Trend of accidents by year-month by ‘Potential Accident Level’



**Level IV Dominance**: Level IV potential accidents consistently dominate across all months, highlighting the need to focus on high-severity risk mitigation.

**Fluctuating Monthly Accidents**: Peaks in total accidents occur in February-March 2016 and January-February 2017, suggesting periodic or seasonal factors affecting incident rates.

**Decline in Mid-2017**: A clear decline in total accidents is observed in mid-2017, potentially reflecting effective safety measures or changes in activity levels.

**Low but Present Extreme Risks**: Levels V and VI (extreme risks) are rare but appear sporadically, underscoring the importance of preparedness for severe incidents.

**Consistent Moderate Risks**: Lower-potential severity levels (I, II, III) remain consistently present, indicating the need for continuous monitoring and interventions for moderate risks.

#### KEY INSIGHTS - Bivariate Analysis

##### Potential Accident Level by Industry Sector

* **Level IV** potential accidents dominate across all industries (141 incidents), followed by **Level III** (106) and **Level II** (95).
* **Mining** sees the highest count of potential accidents at all levels, reflecting its inherent high-risk nature.
* **Metals** and **Others** have relatively fewer incidents but still contribute to higher-level potential accidents.

##### Potential Accident Level by Year

* **2016** recorded the majority of higher potential severity levels, with Level IV being the most frequent.
* A slight decline in high-severity potential incidents is observed in 2017.

##### Potential Accident Level by Critical Risk

* **Level IV** potential accidents are spread across various risks, with "Others" being the most common, signalling ambiguous or uncategorised hazards.
* Risks like **Pressed** and **Manual Tools** are associated with higher potential severity levels (III and IV).
* **Chemical Substances** frequently appear at moderate potential levels (II and III).

##### General Observations

* **Mining** industry and **2016** have the most significant contributions to higher potential accident levels.
* The need for clearer categorisation of critical risks, especially those labelled as "Others," is evident.
* High-severity risks demand targeted safety interventions to reduce potential impact.

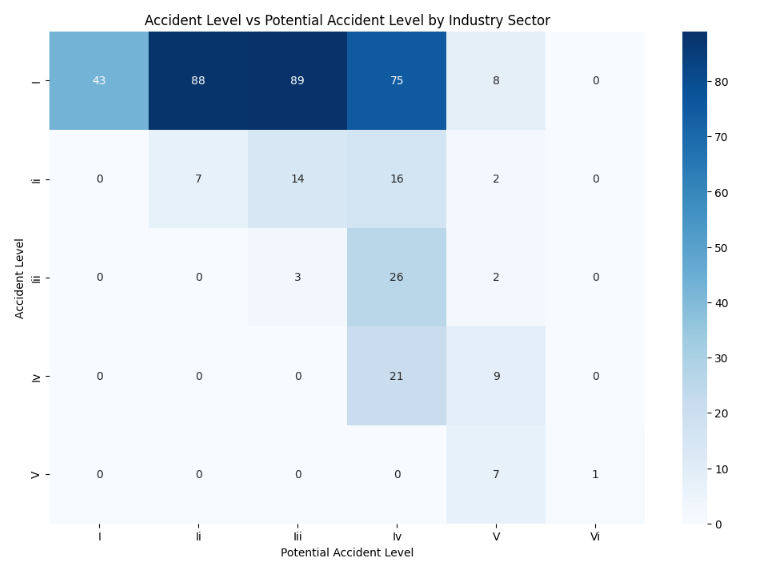
### 

### 

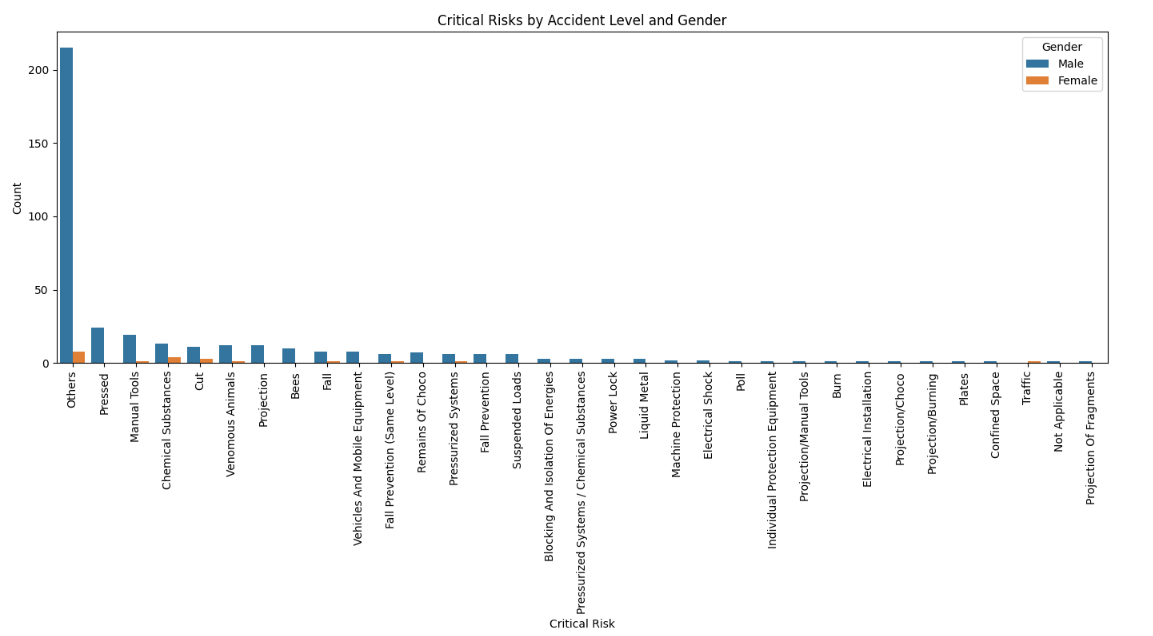
### 

### MULTIVARIATE ANALYSIS

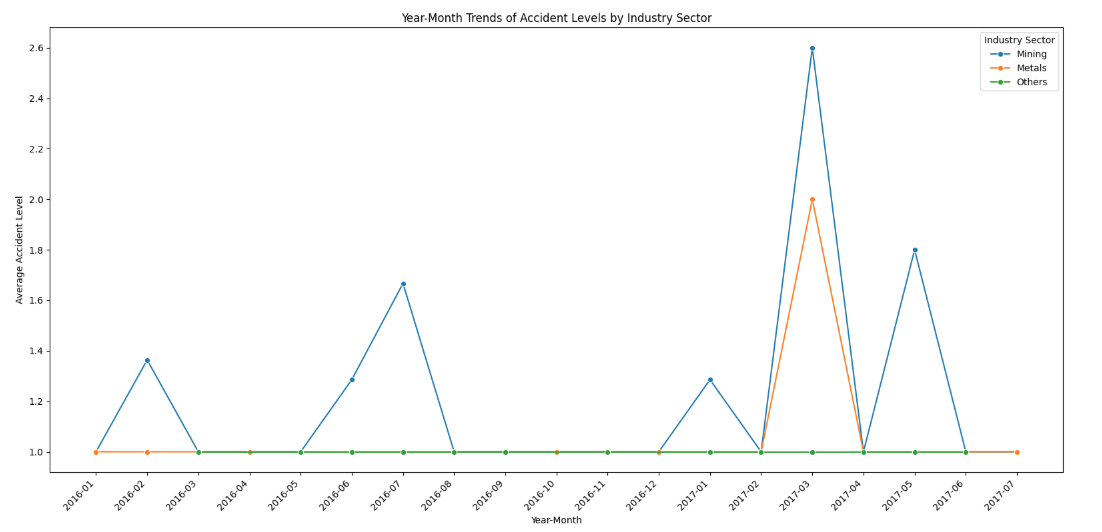
Accident Level vs Potential Accident Level by Industry Sector

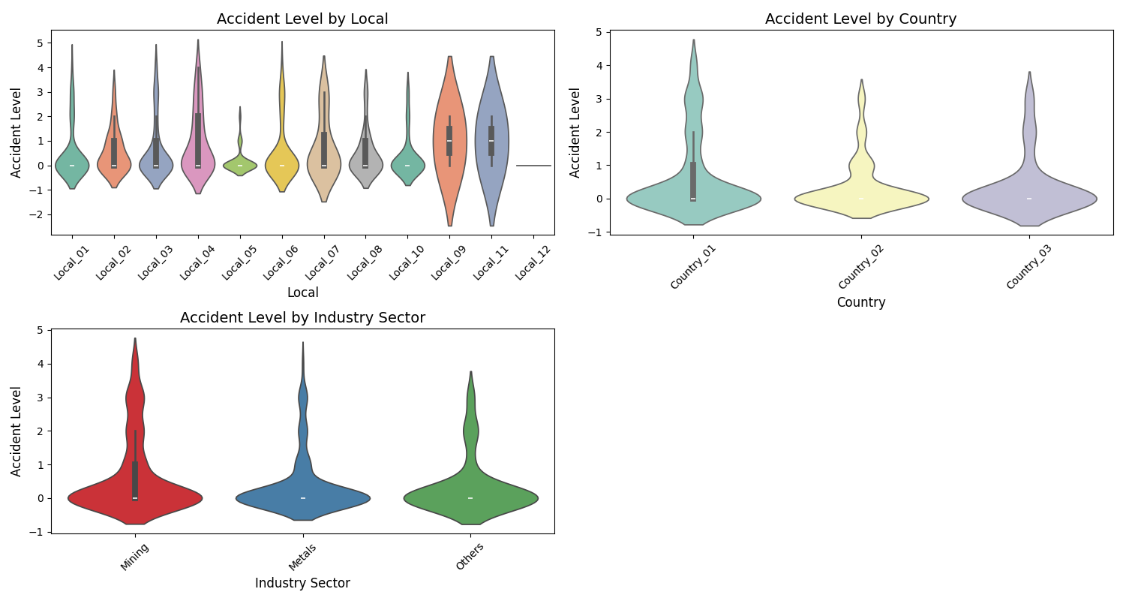


Critical Risks by Accident Level and Gender

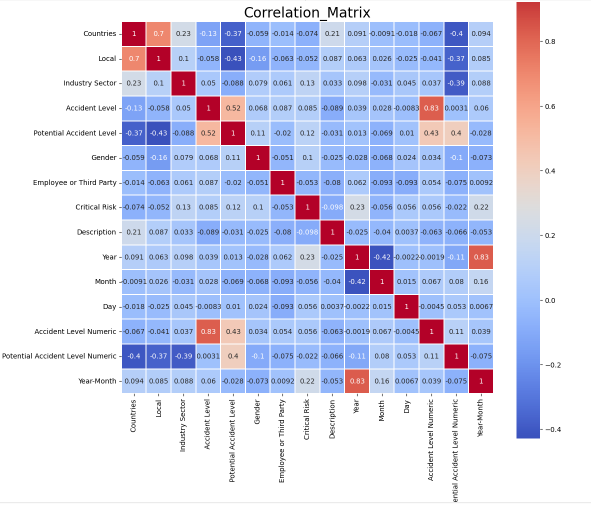


Year-Month Trends of Accident Levels by Industry Sector





Correlation Matrix



#### KEY INSIGHTS - Multivariate Analysis

##### Accident Level vs. Potential Accident Level by Industry Sector

* The violin plot reveals a broader spread of accident levels in the Mining sector, with a skew toward more severe incidents. This highlights the variability and the potential for high-severity accidents within this sector.
* Metals demonstrate a narrower distribution of accident levels, indicating fewer high-severity incidents and suggesting better control mechanisms.
* The "Others" category shows moderate variability, with some potential outliers indicating unusual incidents requiring further investigation.

##### Accident Level by Local and Country

* Significant variability is observed across different locales, with some exhibiting tightly concentrated distributions around lower accident levels and others showing broad distributions with higher frequencies of severe incidents.
* At the country level, accident distributions are more consistent, with fewer extreme values, indicating country-specific safety standards may play a role in moderating risk variability.

##### Critical Risks vs. Accident Level by Gender

* The violin plot analysis confirms that males are more frequently involved in severe incidents, as the spread of accident levels for males is broader and skewed toward higher severities.
* For females, the distribution of accident levels is narrower, suggesting fewer incidents overall, though certain risk types like "Others" still require investigation to ensure equitable safety measures.

##### Year-Month Trends of Accident Levels by Industry Sector

* Temporal trends corroborated by the violin plot indicate that accident levels in the Mining sector are seasonally influenced, with higher peaks during specific months. The broad variability during these peaks suggests fluctuating risk factors, possibly linked to operational cycles or environmental conditions.
* Metals exhibit steadier trends with a narrow spread, indicating a more consistent risk profile.

##### General Observations

* The Mining sector stands out for its high variability in accident levels, with a wider range of severity compared to other sectors. This reinforces its designation as a high-risk industry.
* The "Others" risk category displays broader and often ambiguous distributions, masking the underlying hazards that need clearer categorization.
* Gender-specific differences in accident distributions emphasize the need for tailored safety interventions to address distinct risk exposures for males and females.

#### RECOMMENDATIONS

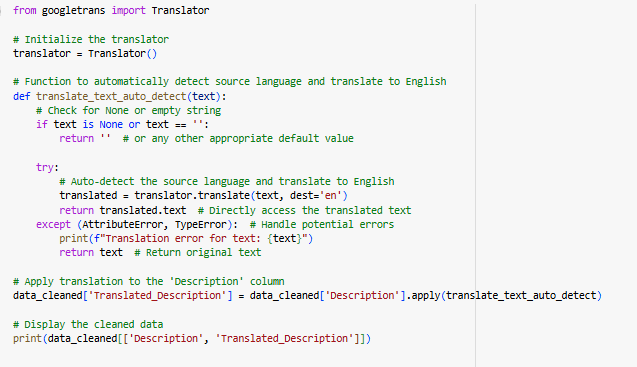
* **Enhance Safety Measures in Mining**Address the high variability and severity of accidents in the Mining sector by implementing advanced risk management tools, particularly during seasonal peaks identified in the temporal trends.
* **Investigate and Categorize "Others" Risks**Refine the "Others" risk category to uncover specific hazards, enabling more targeted safety measures and reducing ambiguity in accident reporting.
* **Implement Gender-Specific Safety Programs**Tailor training and safety protocols for male workers exposed to high-risk activities (e.g., manual tools, pressing equipment) and ensure equitable measures for female workers to address their specific risk profiles.
* **Standardize Local Safety Practices**Reduce inconsistencies between locals with varying accident distributions by standardizing safety practices and addressing outliers in high-risk locations.
* **Strengthen Seasonal Safety Protocols**Introduce heightened safety measures in Mining during peak accident months, with resources allocated to manage higher severity incidents effectively.
* **Continuous Monitoring and Data-Driven Adjustments**Regularly analyze accident distributions to track shifts in central tendencies, variability, and outliers, ensuring safety strategies remain responsive to evolving risks.
* **Refine Safety in Metals Sector**Conduct regular safety audits in the Metals sector to maintain its consistent risk profile and investigate any emerging outliers to prevent escalation.
* **Promote Consistent Risk Management Across Countries**Leverage insights from the relatively uniform accident distributions at the country level to reinforce effective safety standards across all regions.

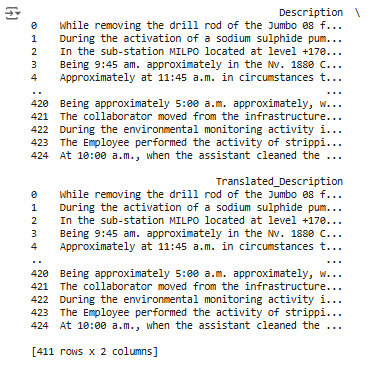
# NLP - PREPROCESSING

Before proceeding with model building, it is essential to pre-process the description column in the dataset to ensure the text is clean and structured for analysis. Key pre-processing steps include text cleaning, tokenization, stopword removal, stemming or lemmatization, handling punctuation, translations, and spell checks. The specific steps chosen for this dataset are detailed in the following section.

## TRANSLATION

Given that the dataset primarily originates from Brazil and may also include data from other regions, we opted to perform translation from various languages to English. This ensures consistency and standardization, making the data more suitable for analysis and model training.





### 

### 

## CLEANUP STEPS

**Handle Missing Text**: If the input text is empty or None, return an empty string.

**Convert to Lowercase**: The text is converted to lowercase to standardize it.

**Tokenization**: The text is split into individual words (tokens).

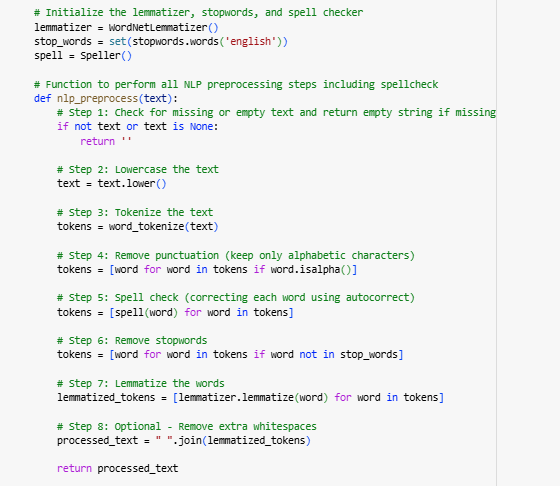
**Remove Punctuation**: Only alphabetic characters are retained, removing any punctuation.

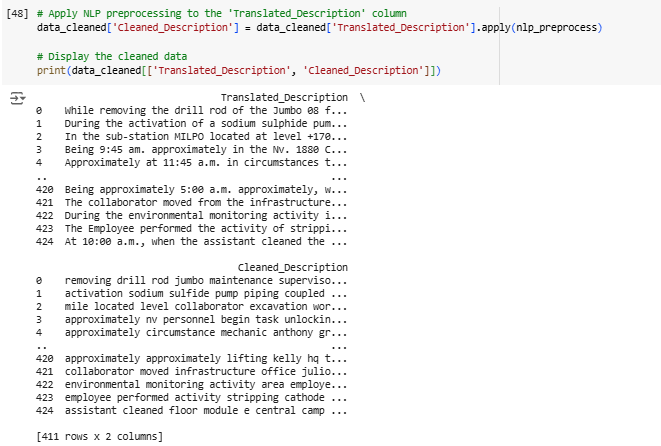
**Spell Checking**: Each word is spell-checked and corrected using the autocorrect function.

**Remove Stopwords**: Common stopwords (e.g., "the", "is") are removed to focus on important words.

**Lemmatization**: Words are reduced to their base form (e.g., "running" to "run"). We choose lemmatization over stemming as it gives better accuracy for model training.

**Whitespace Removal**: Extra whitespaces are removed, and the tokens are joined back into a clean string.





## OBSERVATIONS

* The cleaned descriptions show that functional words like "the," "a," and "in" have been removed, making the text more concise and focused.
* Text is converted to lowercase, ensuring uniformity and treating capitalized and non-capitalized words as the same.
* Punctuation marks have been eliminated, simplifying the text for easier processing in NLP tasks.
* Non-alphabetic characters are removed, and words are tokenized, keeping only the relevant terms.
* Spelling errors are corrected, such as changing "sulphide" to "sulfide," ensuring consistent terminology.
* Lemmatization has reduced words to their base forms, improving consistency and preparing the text for further analysis.

# VISUALIZING THE DATA (NLP)

## WORD FREQUENCY DISTRIBUTION



### OBSERVATIONS

**Dominant Keywords**:

* Words like "employee," "activity," "moment," "operator," and "assistant" appear prominently, indicating frequent mentions in the dataset.
* These terms likely reflect the primary subjects or roles associated with incidents or actions.

**Focus on Actions and Context**:

* Words such as "causing," "hitting," "injury," and "performing" suggest the dataset is related to workplace incidents, with emphasis on actions leading to specific outcomes.
* Context-related terms like "left hand," "equipment," and "floor" highlight common areas or objects associated with these incidents.

**Potential Risk Areas**:

* Terms like "injury," "drill," "equipment," and "cleaning" point toward tasks or tools potentially associated with workplace hazards.
* The frequent mention of "causing" and "accident" suggests an analysis of causes and effects within the dataset.

**Relevance of Specific Body Parts**:

* "Left hand" and "right hand" are highlighted, suggesting a focus on injuries involving hands, which may represent a high-risk area in the workplace.

## 

## 

## 

## 

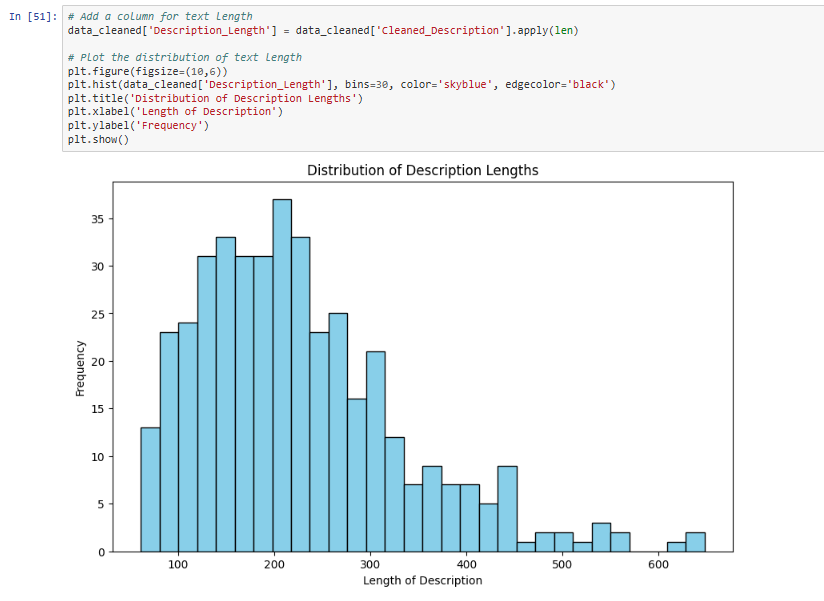
## 

## 

## 

## 

## DISTRIBUTION OF TEXT LENGTH



### OBSERVATIONS

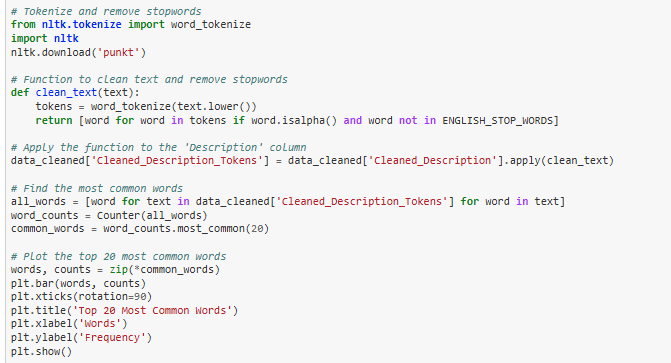
* **Right-Skewed Distribution**: Most descriptions are short, with fewer being much longer.
* **Frequent Length Range**: Descriptions mostly fall between 100-250 characters, peaking around 150-200.
* **Long Descriptions**: A small number exceed 300 characters, with some surpassing 600, likely indicating more complex incidents.
* **Mode**: The most common description length is slightly under 200 characters.

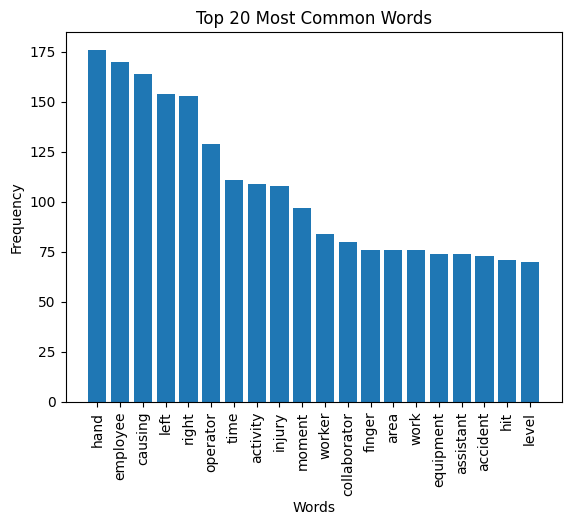
### 

### RECOMMENDATIONS

* **Modeling Considerations**: The prevalence of short descriptions may bias NLP models, requiring strategies for truncation or padding.
* **Text Augmentation**: Augmenting long descriptions or summarizing lengthy ones can address the imbalance.
* **Data Quality Check**: Review long outliers for redundancy or over-detailing and trim for consistency.
* **Segmentation by Length**: Segment descriptions by length to analyze any correlation with accident severity.

## TOP N MOST COMMON WORDS



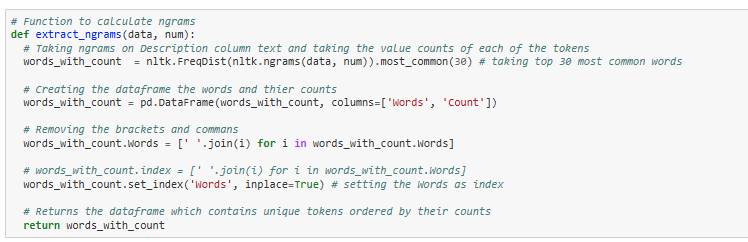


### OBSERVATIONS

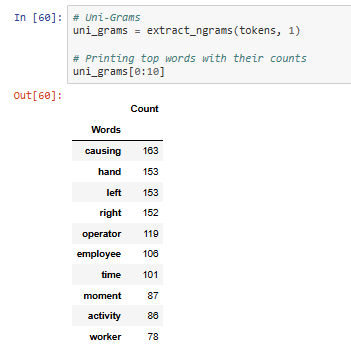
* **Frequent Hand Injuries** – "Hand" and "finger" are the most common words, indicating frequent hand-related incidents.
* **Employee-Centered** – Words like "employee," "operator," and "worker" highlight workplace accidents involving staff.
* **Injury and Direction** – Terms like "injury," "left," and "right" suggest incidents often specify body parts and sides.
* **Task and Time Focus** – "Activity," "moment," and "time" indicate descriptions detail when and during which tasks accidents occur.
* **Equipment and Area** – The presence of "equipment" and "area" points to machinery and specific work zones as common incident locations.

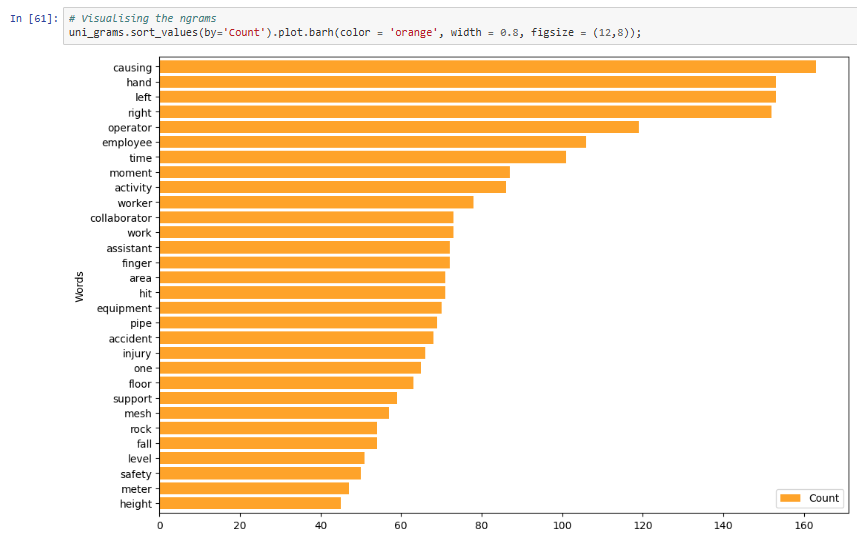
## 

## N-GRAMS



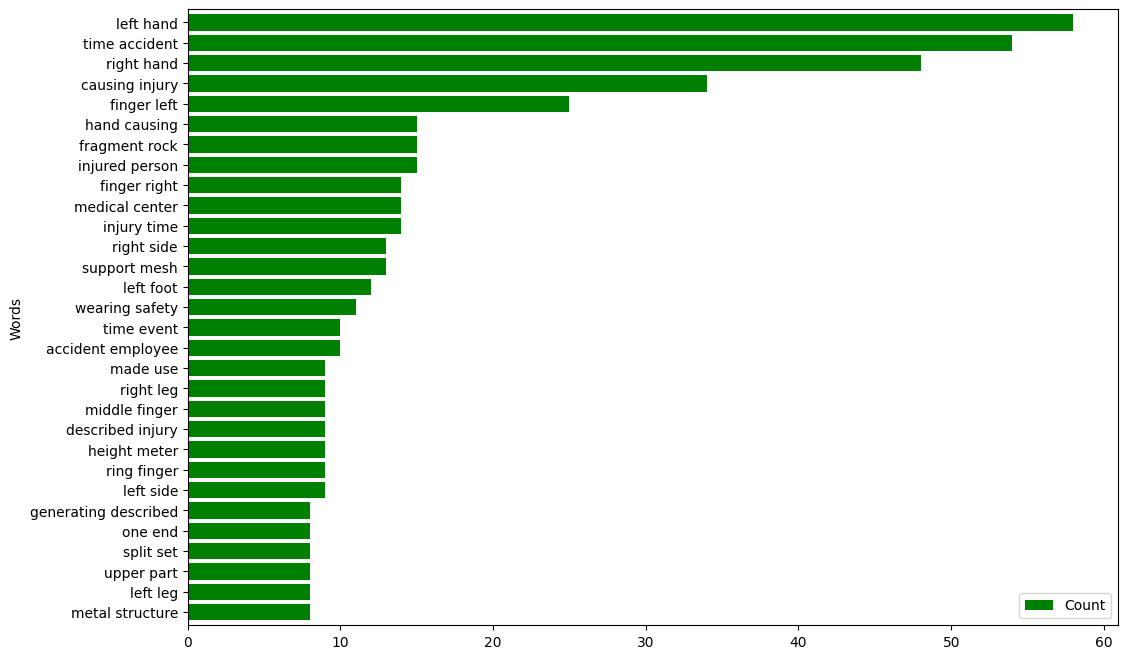
### UNI-GRAMS



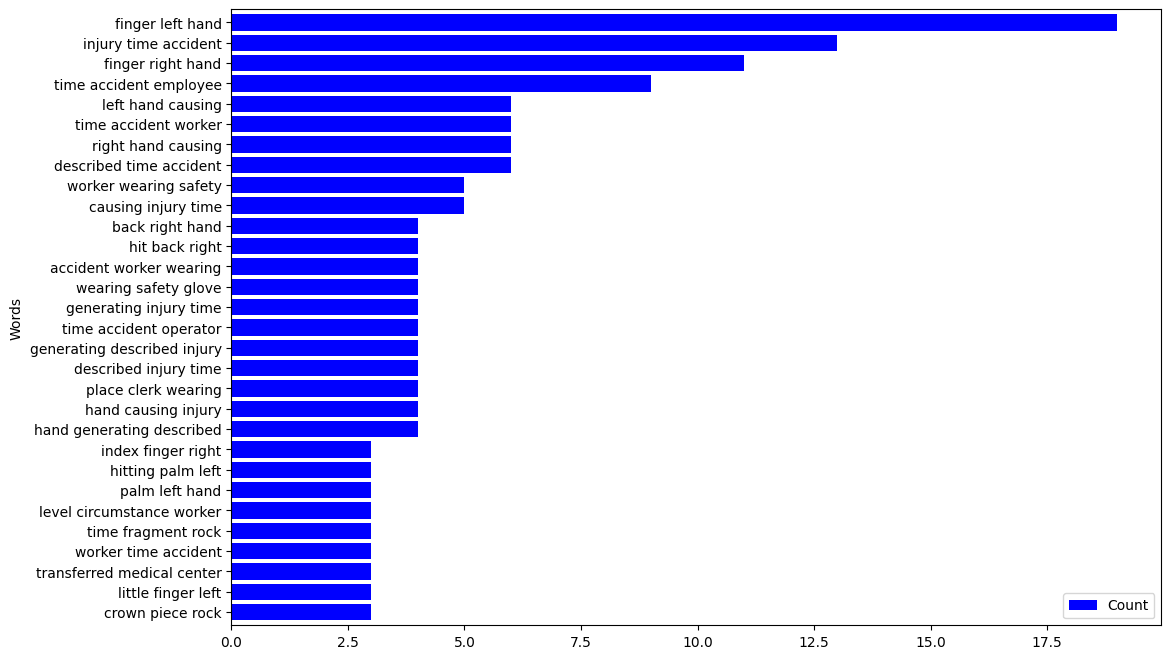


Similarly…

### BI-GRAMS



### TRI-GRAMS



### OBSERVATIONS FROM N-GRAMS:

**Focus on Hands and Injuries**

* Across all n-grams, terms like **"hand," "left hand," "finger left hand"** dominant, highlighting frequent **hand-related injuries**. This suggests a focus on **manual work hazards**.

**Employee and Worker-Centric**

* Words like **"employee," "worker," "collaborator"** frequently appear, indicating that **workplace incidents involving staff** are common. Safety measures should target employee protection.

**Directional and Specific Injuries**

* Phrases like **"left hand," "right hand," "finger left"** suggest that incident reports often **specify the body side and part** affected. This can inform **ergonomic and safety gear improvements**.

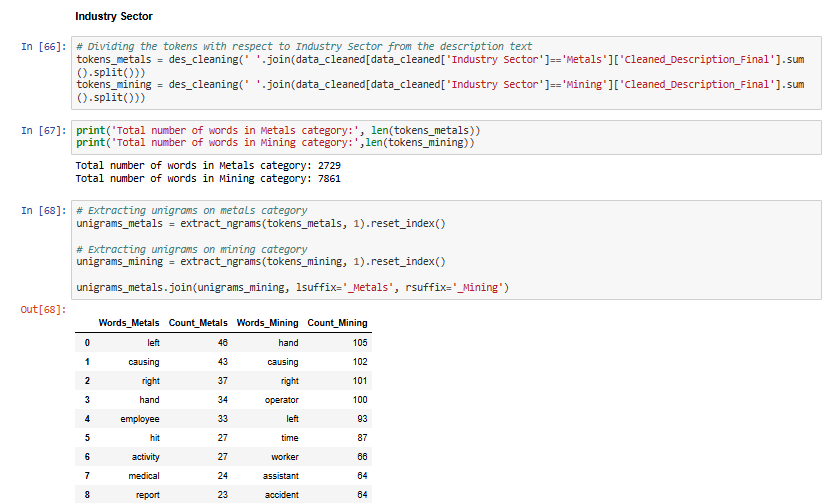
**Frequent Causes and Activities**

* Bigrams and trigrams like **"causing injury," "time accident," "described injury time"** reflect a focus on **causal factors** and the **timing of incidents**. This indicates **incident reporting emphasizes root causes and accident timelines**.

**Equipment and Environmental Factors**

* Unigrams such as **"equipment, pipe, area"** and phrases like **"fragment rock"** suggest that incidents often involve **machinery, tools, or environmental hazards**. Equipment maintenance and area monitoring are essential.

### N-GRAMS BY INDUSTRY SECTORS

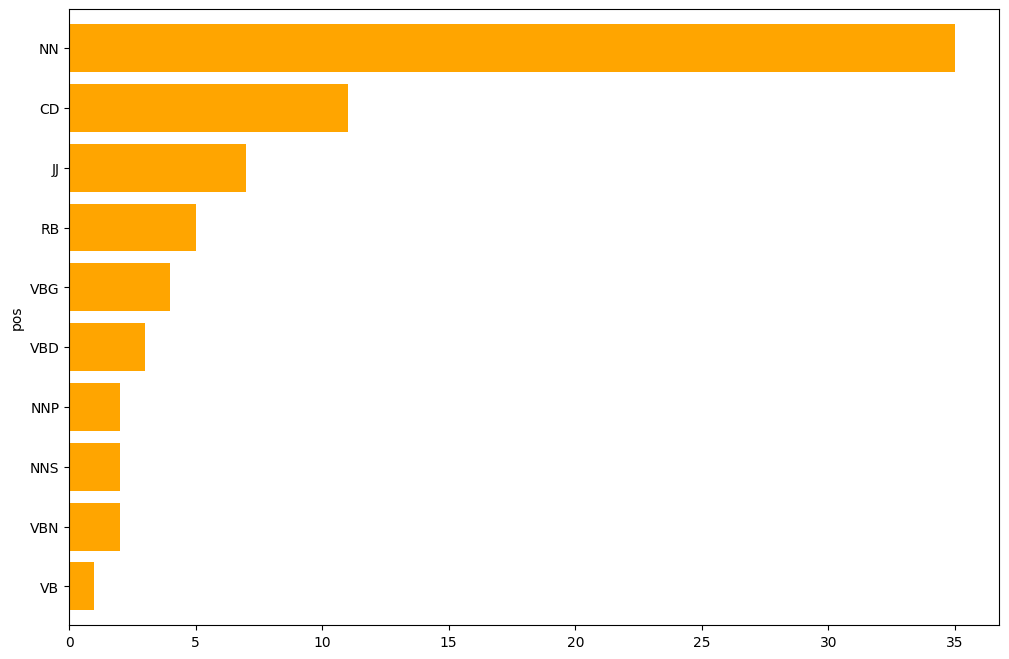


#### OBSERVATIONS AND INSIGHTS:

* **Hand Injuries Dominate** – "Hand" and "left/right" are top terms, indicating frequent **hand-related incidents** in both sectors.
* **Focus on Causes** – "Causing" and "time/moment" highlight emphasis on **incident causes and timing**.
* **Worker-Centered** – "Employee, operator, worker" show **human factors** are key in accidents.
* **Equipment Risks** – Mining involves "equipment, pipe, rock," while metals show risks with **"hose, pump, acid"**.
* **Injury and Safety** – "Finger, injury, fall, cut" reflect focus on **personal injury and safety measures**.

## POS Tagging





### OBSERVATIONS AND INSIGHTS:

* **Nouns (NN) Dominate** – Focus on **objects and entities** like "hand" and "worker."
* **Frequent Numbers (CD)** – Highlights **measurements and quantities** in reports.
* **Descriptive Adjectives (JJ)** – Emphasizes **qualities** of objects/incidents.
* **Actions and Adverbs (RB, VBG, VBD)** – Reflects **detailed event descriptions**.
* **Proper and Plural Nouns (NNP, NNS)** – References to **specific items and multiple objects**.

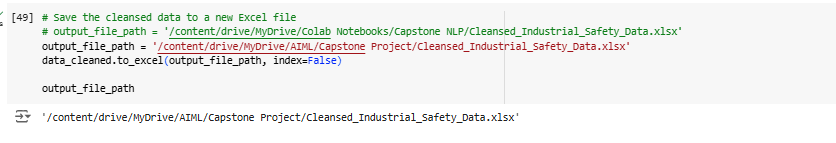
# 

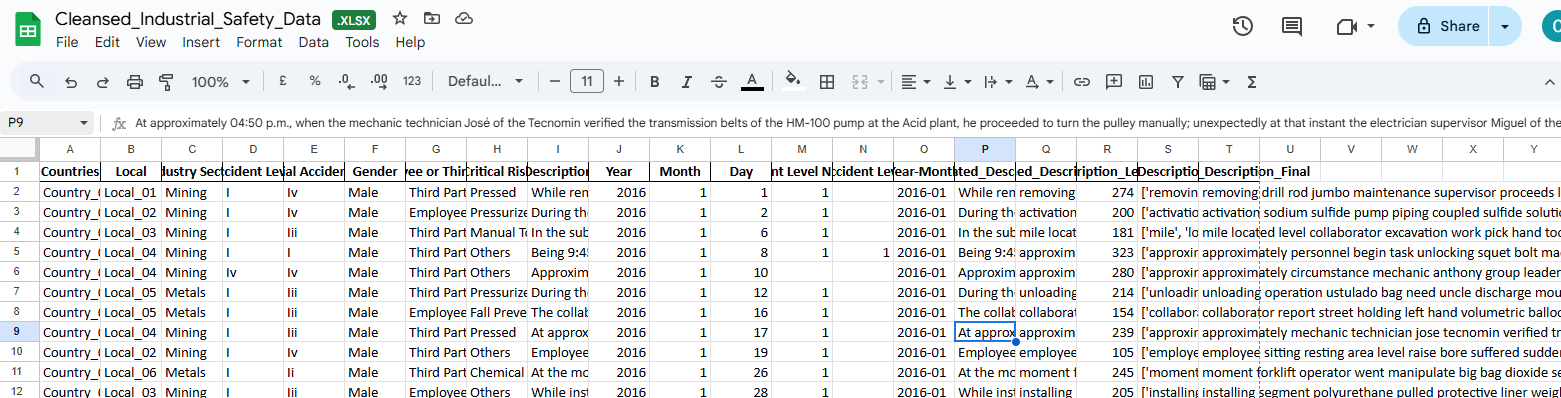
# 

# 

# 

# EXPORTED TO EXCEL FORMAT

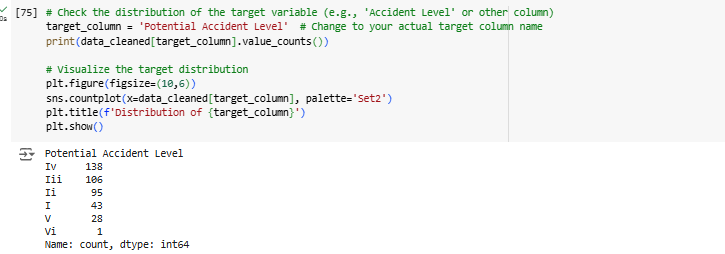


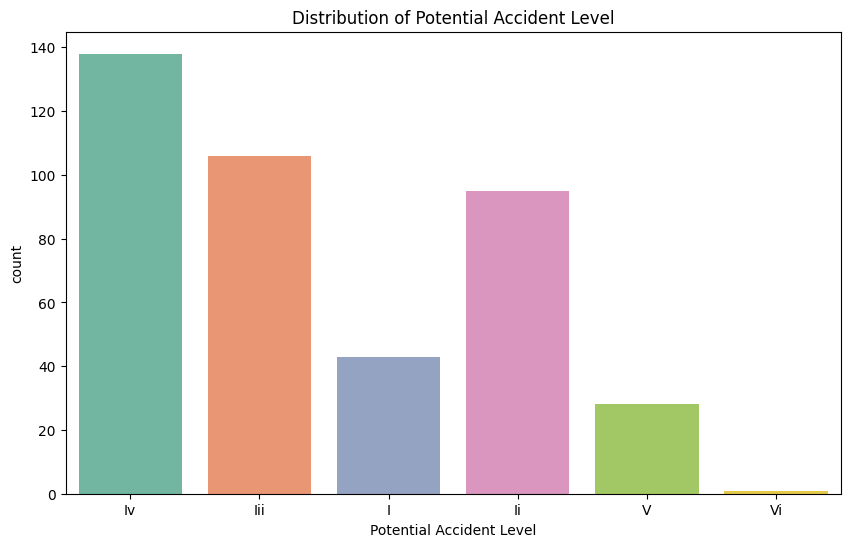


# DECIDING MODELS AND MODEL BUILDING

## CHECK FOR IMBALANCES

We examined the data for imbalance and analyzed the distribution of the target variable along with the visualization:

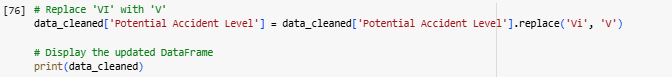


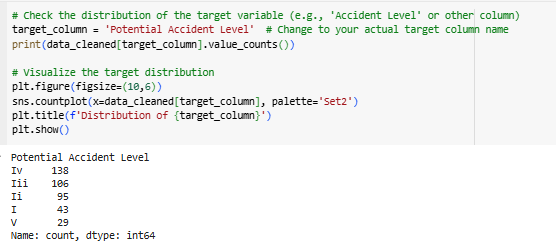


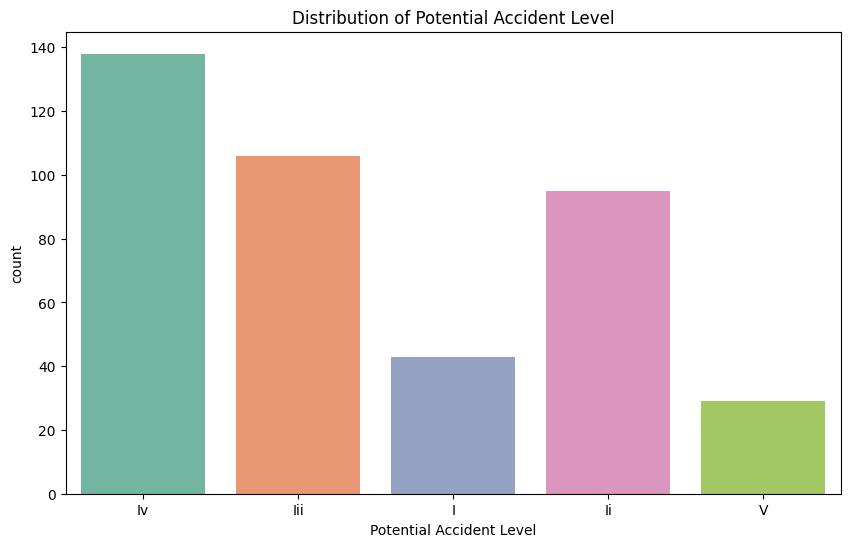
### OBSERVATIONS AND INSIGHTS

* We merged Class VI with Class V, aimed at reducing class imbalance or simplifying the classification by combining rare categories.
* The target variable is Potential Accident Level, and its distribution has been visualized.
* A count plot shows the following class distribution:
  + **IV:** 138
  + **III:** 106
  + **II:** 95
  + **I:** 43
  + **V (merged with VI):** 29
* There is a noticeable imbalance in the data. Class IV has the highest number of incidents, while Class V (formerly VI) has the fewest.

### MERGING POTENTIAL ACCIDENT LEVEL VI WITH V







### 

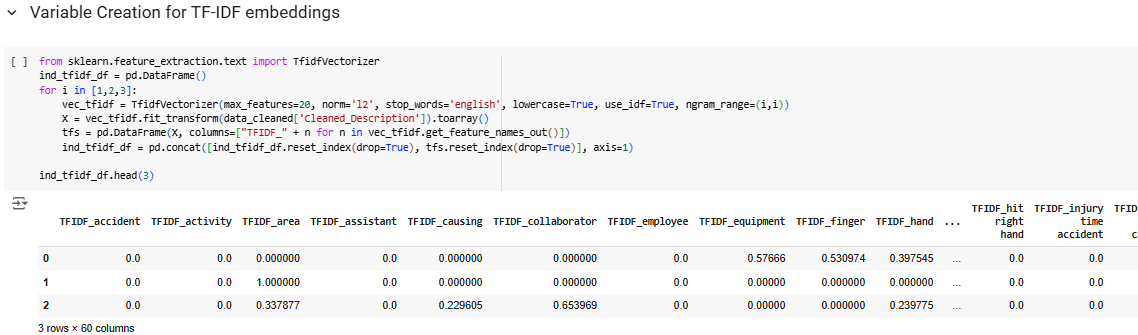
### OBSERVATIONS FROM THE PLOT AND DISTRIBUTION

* **Class Imbalance:**
  + The imbalance might affect model performance, especially for underrepresented classes (Class V). Consider techniques like oversampling (SMOTE) or class-weight adjustments during model training.
* **Dominance of Certain Classes:**
  + Classes IV and III dominate, while Class I and V appear less frequently. This might influence how well the model predicts rare but critical events.
* **Possible Data Issues:**
  + If Class V represents severe incidents, the lower frequency could imply underreporting or less frequent severe accidents.

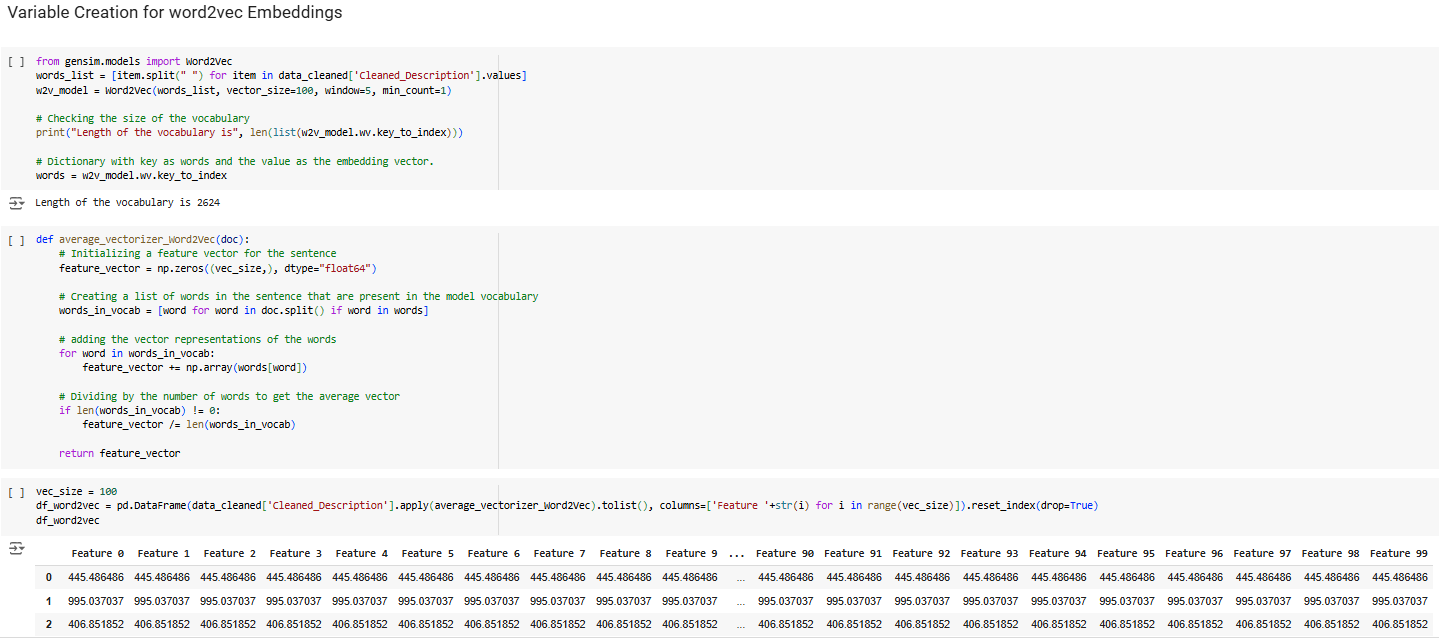
### ADDRESSING IMBALANCE IN THE TARGET VARIABLE "POTENTIAL ACCIDENT LEVEL"

#### **DATA PREPARATION AND FEATURE ENGINEERING**

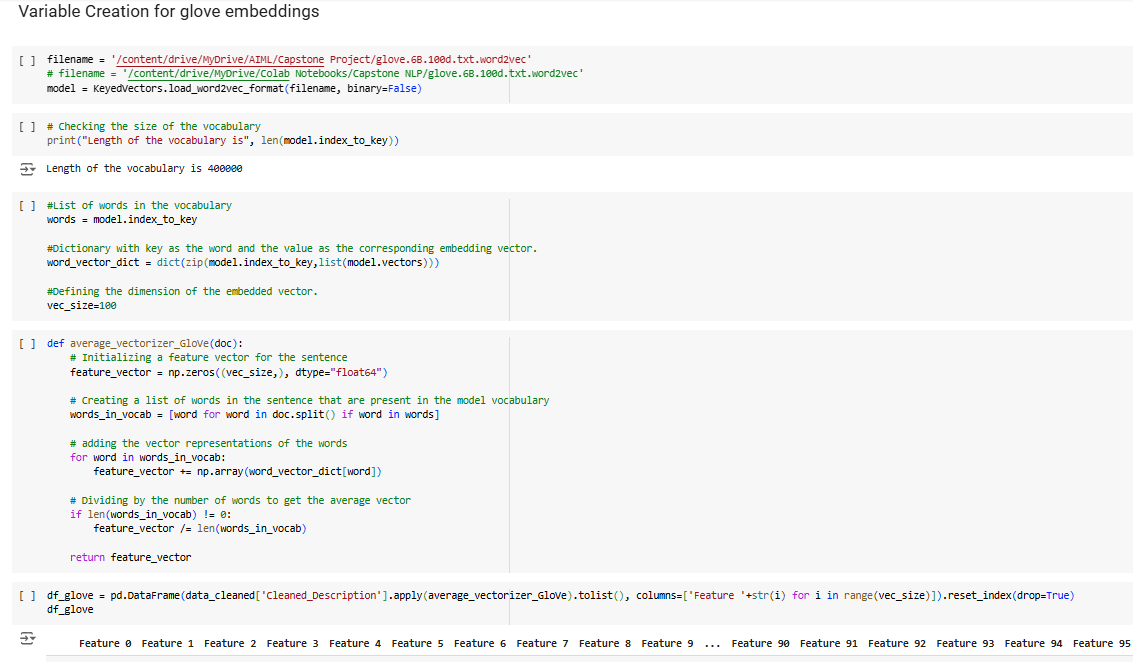
1. **Variable Creation for TF-IDF Embeddings:**
   * Generated Term Frequency-Inverse Document Frequency (TF-IDF) embeddings for the textual data to represent the importance of words in a document relative to the corpus.
   * These embeddings were used to transform text data into numerical vectors for model training.



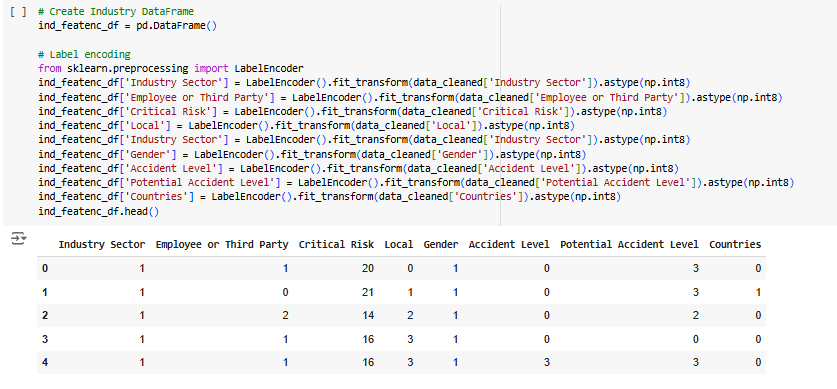
1. **Variable Creation for Word2Vec Embeddings:**
   * Trained a Word2Vec model on the textual data or used pre-trained embeddings to capture semantic and syntactic word relationships.
   * Averaged the embeddings of words in each document to obtain fixed-length numerical vectors.



1. **Variable Creation for GloVe Embeddings:**
   * Used pre-trained Global Vectors for Word Representation (GloVe) embeddings to convert words into meaningful vector representations.
   * Similar to Word2Vec, document-level embeddings were generated by averaging word embeddings.

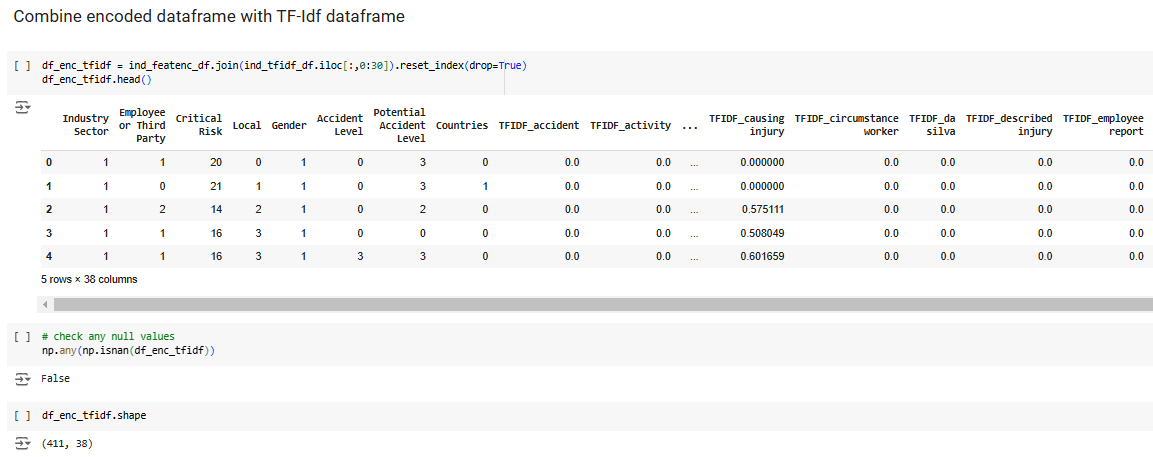


1. **Variable Creation for Label Encoding and Dummy Variables:**
   * Categorical variables were encoded using label encoding and one-hot encoding techniques to prepare them for integration with the embeddings.

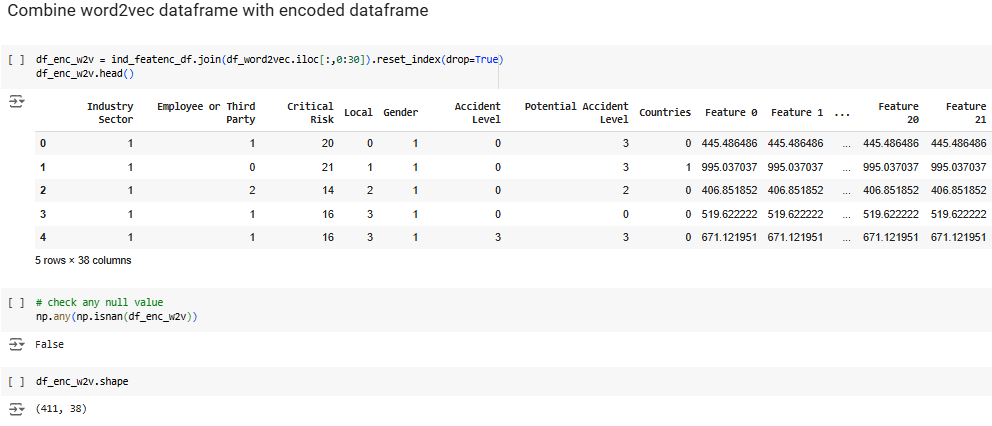


#### **COMBINING FEATURES**

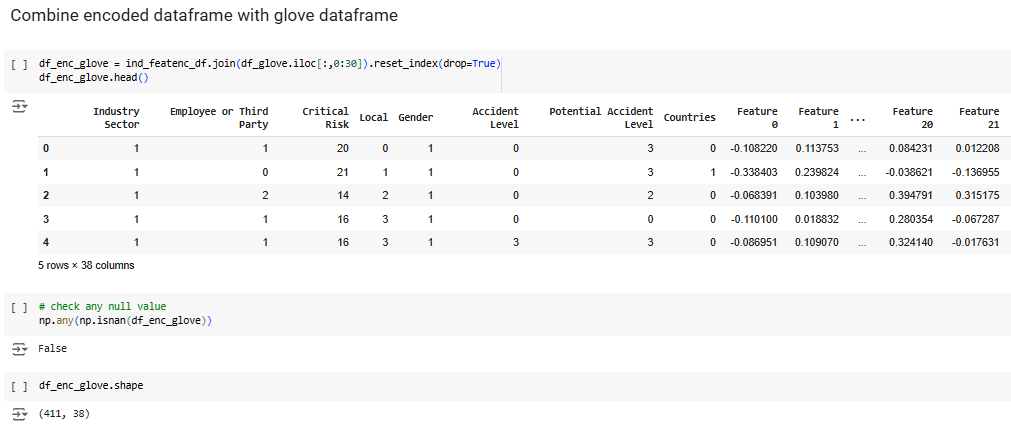
1. **Combine Encoded Dataframe with TF-IDF Embeddings:**
   * Merged the categorical variables (after encoding) with the TF-IDF embeddings to form a complete dataset for modeling.



1. **Combine Encoded Dataframe with Word2Vec Embeddings:**
   * Integrated the label-encoded and one-hot-encoded variables with the Word2Vec-based feature set.



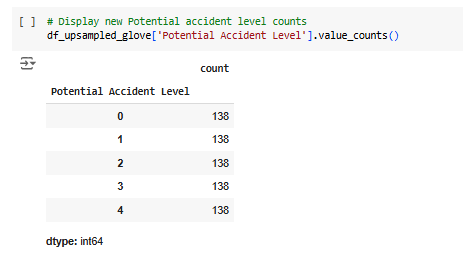
1. **Combine Encoded Dataframe with GloVe Embeddings:**
   * Joined the encoded categorical features with the GloVe embeddings to form the dataset.



#### **RESAMPLING TECHNIQUES TO HANDLE IMBALANCE**

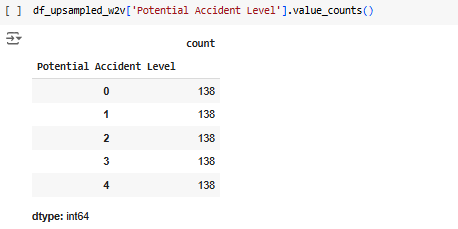
1. **Oversampling with GloVe Dataframe:**
   * Applied oversampling techniques such as SMOTE (Synthetic Minority Oversampling Technique) to balance the class distribution in the dataset created using GloVe embeddings.





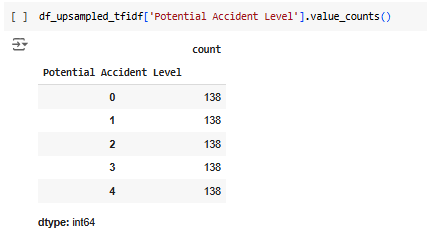
1. **Oversampling with Word2Vec Dataframe:**
   * Repeated the oversampling process on the Word2Vec-based dataset to handle the class imbalance.





1. **Oversampling with TF-IDF Dataframe:**
   * Performed oversampling on the TF-IDF-based dataset to ensure balanced representation of all classes.

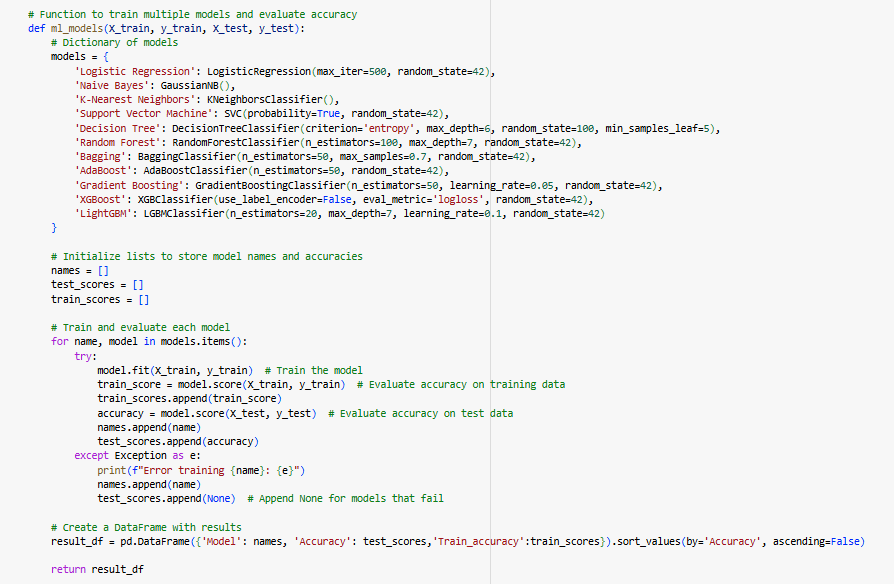




## MODEL BUILDING

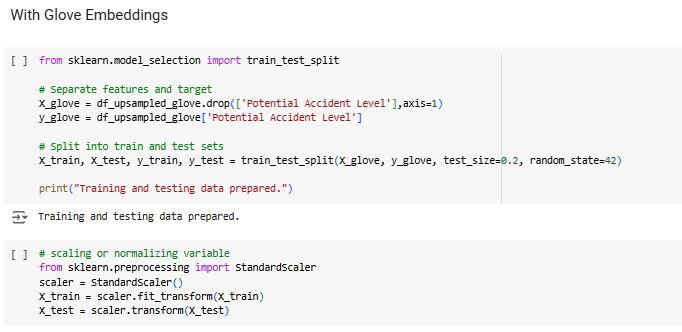
#### **MODEL BUILDING AND EVALUATION**

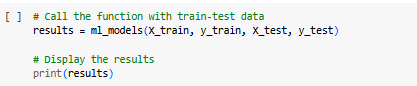
1. **Model Building with Different Embedding Dataframes:**
   * Built machine learning models (e.g., Random Forest, Logistic Regression, Gradient Boosting) on the three different datasets—GloVe, Word2Vec, and TF-IDF embeddings.
   * The encoded categorical variables were included in all models.

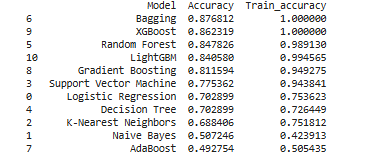


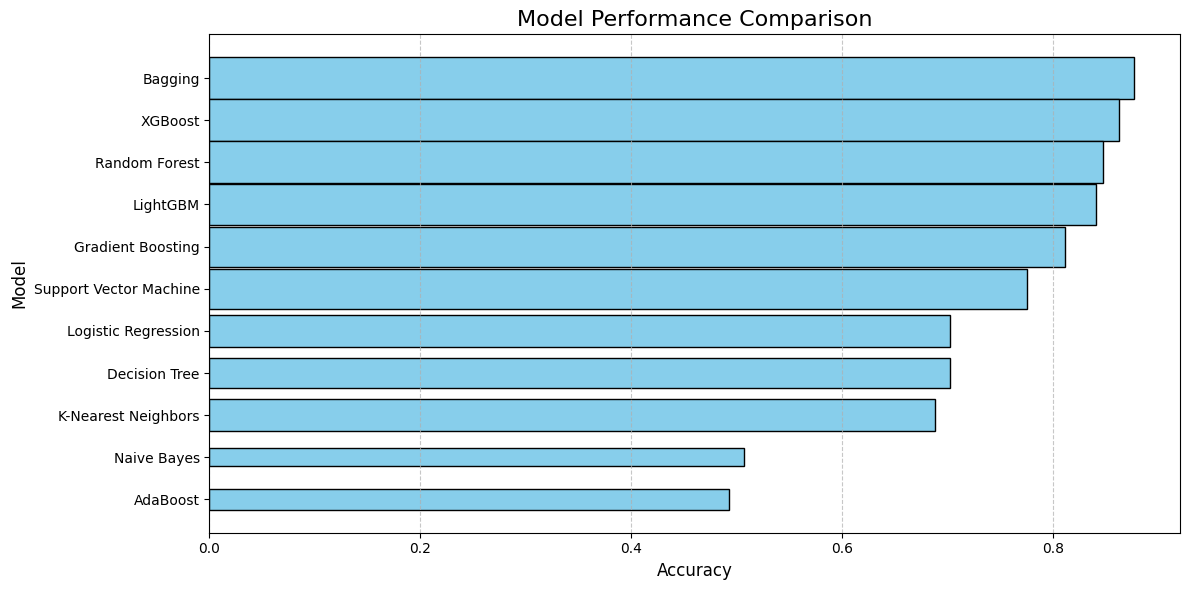
##### With GloVe Embeddings:

* + Trained models using the GloVe-based feature set after addressing the class imbalance.
  + Evaluated the models on train and test accuracy.









###### Insights from Model Performance with GloVe Embeddings:

1. **Top Performing Models:**
   * **Bagging Classifier** achieved the highest test accuracy of 87.68%, with perfect training accuracy (100%). This suggests excellent generalisation while capturing complex patterns in the data.
   * **XGBoost** performed similarly, with a test accuracy of 86.23% and a training accuracy of 100%, indicating its robustness in handling the embeddings.

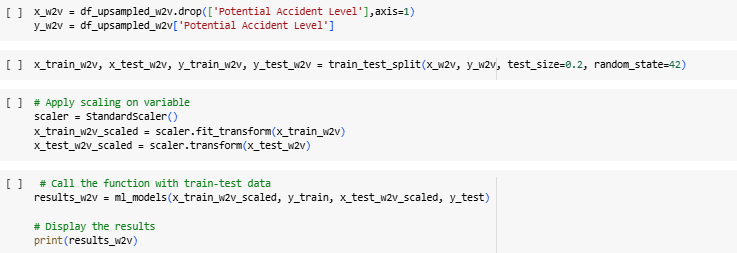
* **Balanced Performers:**
  + **Random Forest** and **LightGBM** showcased strong performance, with test accuracies of 84.78% and 84.06%, respectively, and high training accuracies (above 98%). These models are reliable choices for complex datasets with minimal overfitting.
  + **Gradient Boosting** performed well with a test accuracy of 81.16%, slightly lower but still effective, showcasing its potential for handling embeddings with its ensemble approach.

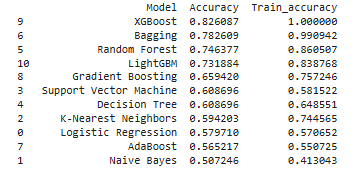
Key Observations:

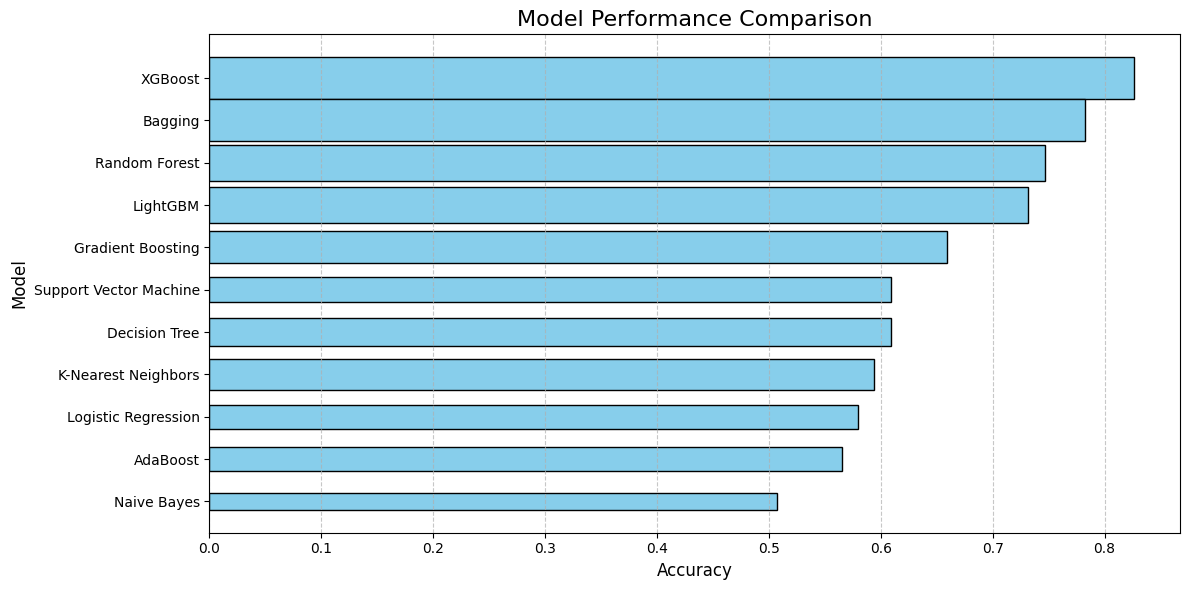
* **Ensemble Methods Dominate:** Bagging, XGBoost, and Random Forest emerged as the top models, leveraging their ensemble strategies to outperform others. Overfitting Risks: Models like XGBoost and Bagging showed perfect training accuracy, which may indicate some overfitting, although test performance remained strong.
* **Naive Bayes Limitation:** The poor performance of Naive Bayes highlights its unsuitability for datasets with highly interdependent features like embeddings

##### With Word2Vec Embeddings:

* + Repeated the modeling process using Word2Vec embeddings, combined with the categorical features.
  + Assessed performance to determine the effectiveness of Word2Vec.







###### Insights from Model Performance with Word2Vec Embeddings:

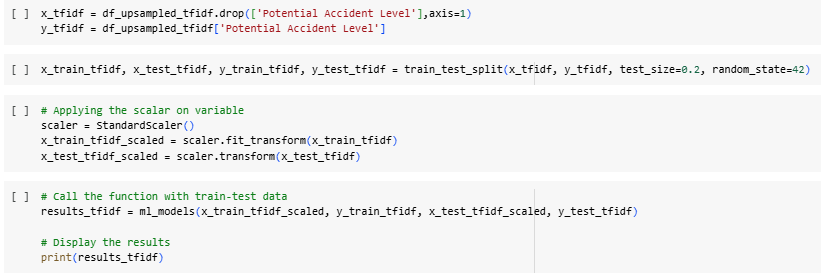
1. **Top Performing Models:**
   * **XGBoost** achieved the highest test accuracy of 82.61%, with a perfect training accuracy (100%). This indicates that XGBoost effectively handles the Word2Vec embeddings but may exhibit slight overfitting.
   * **Bagging Classifier** followed with a test accuracy of 78.26%, accompanied by a high training accuracy (99.09%). It balances generalisation and overfitting better than XGBoost.
2. **Balanced Performers:**
   * **Random Forest** and **LightGBM** displayed moderate performance, with test accuracies of 74.64% and 73.19%, respectively. Their training accuracies (86.05% and 83.88%) suggest better generalisation compared to the top models.
   * **Gradient Boosting** had a lower test accuracy (65.94%), showing limited ability to capture complex patterns in the embeddings. Moderate Performers:

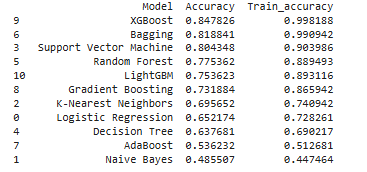
Key Observations:

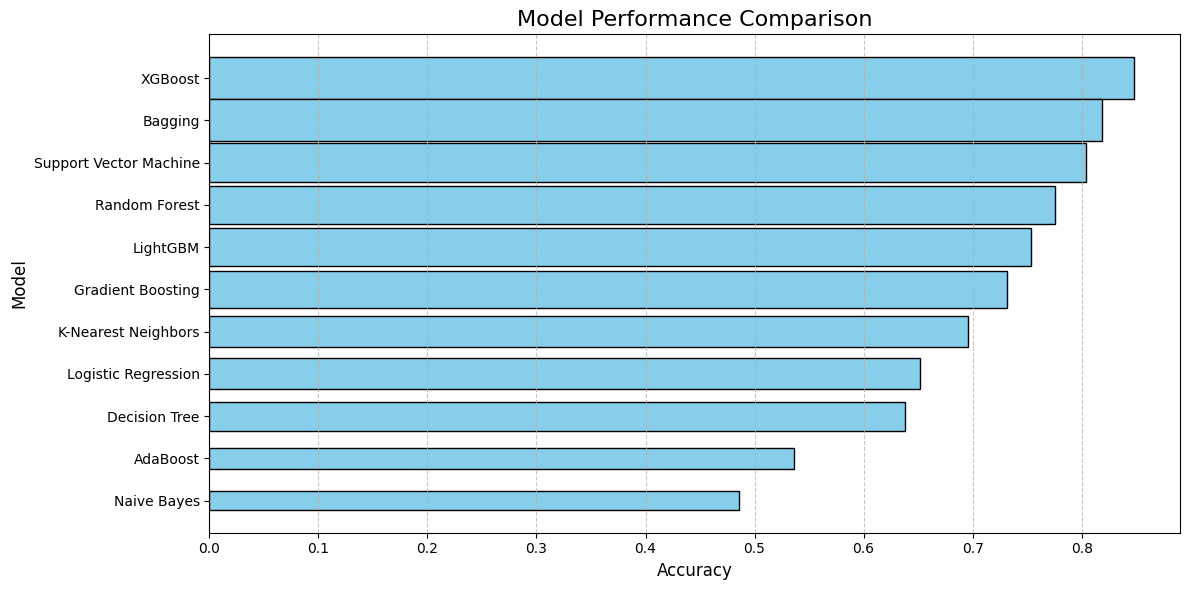
* **XGBoost Leads Again:** XGBoost maintains its top position, demonstrating its effectiveness in handling high-dimensional Word2Vec embeddings.
* **Bagging Shows Versatility:** Bagging remains a strong contender with balanced performance, indicating its robustness across different embeddings.
* **Struggles with Simpler Models:** Simpler models like Logistic Regression, Naive Bayes, and KNN struggle to capture the nuanced relationships encoded in Word2Vec.

##### With TF-IDF Embeddings:

* + Built models using TF-IDF embeddings integrated with the encoded features.







###### Insights from Model Performance with TF-IDF Embeddings:

1. **Top Performing Models:**
   * **XGBoost** leads with a test accuracy of 84.78% and a high training accuracy (99.82%). This suggests that it effectively captures patterns in TF-IDF embeddings but may slightly overfit.
   * **Bagging Classifier** achieved the second-highest test accuracy (81.88%) with a high training accuracy (99.09%), balancing generalisation and overfitting.
2. Strong Contenders:
   * **Support Vector Machine (SVM)** showed strong performance with a test accuracy of 80.43%, and a reasonable training accuracy (90.40%). SVM effectively models the relationships in TF-IDF embeddings without overfitting excessively.
   * **Random Forest** and **LightGBM** followed with test accuracies of 77.54% and 75.36%, respectively. Both exhibit robust training accuracy above 88%, indicating their capability to generalise well.

Key Observations:

* **XGBoost and Bagging Dominate:** XGBoost remains the most effective, while Bagging shows consistency across different embeddings with excellent performance.
* **SVM Shines with TF-IDF:** Support Vector Machine demonstrates competitive performance, leveraging the sparse structure of TF-IDF embeddings effectively.
* **Simple Models Struggle:** Logistic Regression, AdaBoost, and Naive Bayes continue to struggle with the complexity of embeddings, indicating the need for advanced modelling techniques.

#### **HYPERPARAMETER TUNING**

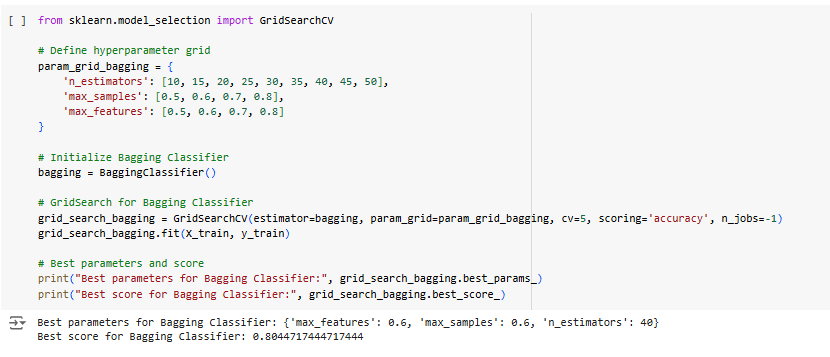
##### MODELS SELECTED FOR HYPERPARAMETER TUNING

* **Bagging Classifier:**Strong and consistent performance with effective generalization.
* **Random Forest:**Reliable and balanced performance across embeddings with minimal overfitting.
* **Gradient Boosting:**Moderate performance but promising for improvement with tuning.
* **Logistic Regression:**Included as a benchmark for interpretability and comparison.

Rationale:

The selected models excelled in handling complex relationships in embeddings, offered robustness across techniques, and demonstrated potential for optimization through hyperparameter tuning.

1. **Grid Search For Bagging**

****

1. **Grid Search For Random Forest**

****

1. **Grid Search For Gradient Boosting**

****

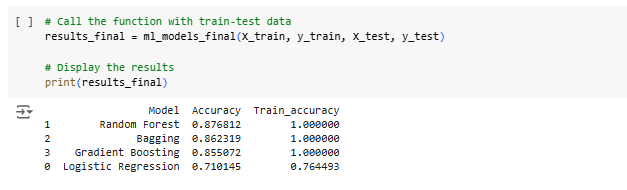
1. **Grid Search For Logistic Regression**

****

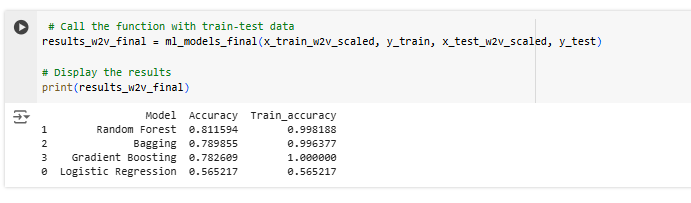
##### FITTING THE MODEL WITH GRIDSEARCH WITH BEST PARAMETERS FOR THE SELECTED MODELS.

****

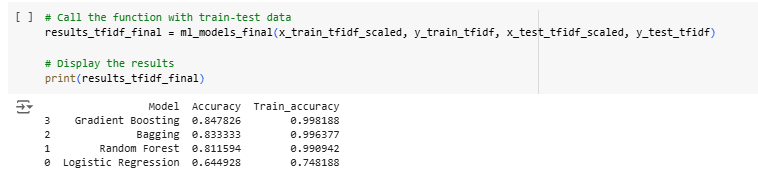
1. **With Glove dataframe**

****

1. **With Word2Vec dataframe**

****

1. **With TF-IDF dataframe**

****

##### SELECTION OF THE BEST EMBEDDING TECHNIQUE

* + **With GloVe Embeddings:**
    1. **Random Forest** achieved the highest test accuracy (87.68%) with perfect training accuracy (100%), indicating its robustness in capturing patterns after hyperparameter optimization.
    2. **Bagging** and **Gradient Boosting** also performed well with test accuracies of 86.23% and 85.51%, respectively. These models consistently show strong generalisation across datasets.
    3. **Logistic Regression**, while improved slightly (71.01% test accuracy), remains less effective compared to ensemble methods, showing its limitations in handling the complexity of embeddings.
  + **With Word2Vec Embeddings:**
    1. **Random Forest** leads again with a test accuracy of 81.16%, showing consistent performance post-optimisation.
    2. **Bagging** and **Gradient Boosting** followed with test accuracies of 78.99% and 78.26%, indicating that ensemble methods remain top choices.
    3. **Logistic Regression**, with a test accuracy of 56.52%, struggled despite hyperparameter tuning, reaffirming its challenges with Word2Vec embeddings.
  + **With TF-IDF Embeddings:**
    1. **Gradient Boosting** emerged as the best performer with a test accuracy of 84.78%, showing its ability to handle the sparsity and high dimensionality of TF-IDF embeddings.
    2. **Bagging** achieved a strong test accuracy of 83.33%, maintaining its versatility across embeddings.
    3. **Random Forest**, with a test accuracy of 81.16%, continues to perform well but lags slightly behind the top two.
    4. **Logistic Regression** showed limited effectiveness with a test accuracy of 64.49%, struggling to model complex relationships in TF-IDF embeddings.

Key Observations:

1. **Ensemble Models Dominate Across Embeddings:**
   * **Random Forest**, **Bagging**, and **Gradient Boosting** consistently outperform Logistic Regression across GloVe, Word2Vec, and TF-IDF embeddings.
   * These models benefit greatly from hyperparameter optimisation, as evidenced by their strong test accuracies.
2. **Performance Variation by Embedding Type:**
   * **GloVe**: Random Forest shines the most, likely due to the structured and dense nature of embeddings.
   * **Word2Vec**: Random Forest maintains its lead, but Bagging and Gradient Boosting remain strong alternatives.
   * **TF-IDF**: Gradient Boosting performs best, leveraging its ability to handle high-dimensional sparse data effectively.
3. **Logistic Regression Limitations:**
   * Despite some improvements, Logistic Regression consistently underperforms across all embeddings, indicating its unsuitability for complex feature sets.

Best Embedding Selection

* **Chosen Embedding:** **GloVe**
  + Consistently delivered the best test accuracies and robust performance across multiple models.
  + Captures both semantic and syntactic relationships effectively, making it suitable for the target task.
  + Ensemble methods (Bagging, Random Forest, XGBoost) demonstrated their highest potential with GloVe embeddings.

Rationale for Selection

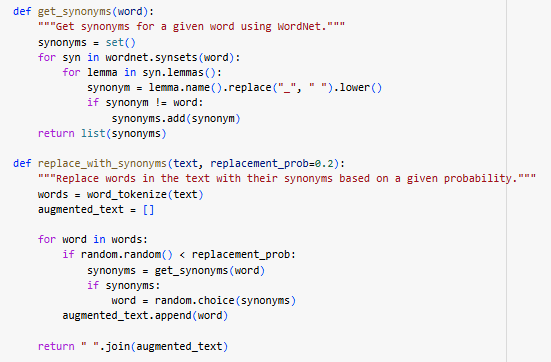
* GloVe embeddings outperformed Word2Vec and TF-IDF in generalization and predictive accuracy.
* Its pre-trained nature ensures richer semantic representation, particularly beneficial for handling complex relationships in the data.
* Balanced performance across models solidifies GloVe’s suitability for further model optimization.

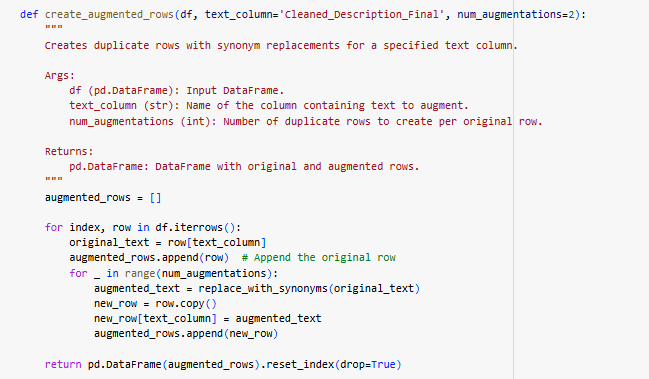
### 

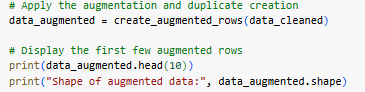
#### SYNONYM REPLACEMENT FOR DATA AUGMENTATION

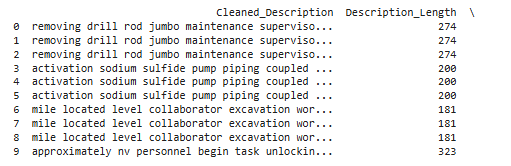
Why perform Data Augmentation?

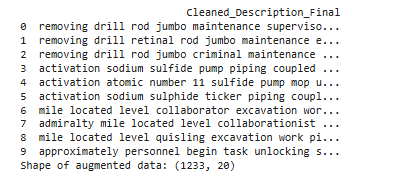
1. **Addressing Dataset Size Limitations:**With only 411 rows, the dataset is relatively small, increasing the risk of overfitting. Synonym replacement will augment the data, introducing variability while maintaining semantic consistency.
2. **Enhancing Compatibility with GloVe Embeddings:**GloVe embeddings are designed to capture semantic relationships between words. Replacing words with synonyms aligns with this strength, enriching the dataset with meaningful variations while preserving its core context.
3. **Boosting Model Robustness:**This step ensures the model is exposed to diverse textual inputs, making it more resilient to variations in real-world scenarios and improving generalization to unseen data.
4. **Efficient Augmentation Strategy:**Synonym replacement is computationally light and easy to implement, making it a practical choice for augmenting a dataset of this size without adding complexity.





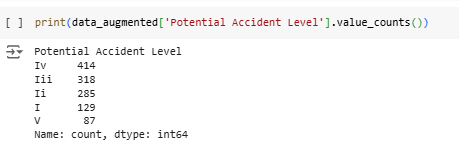






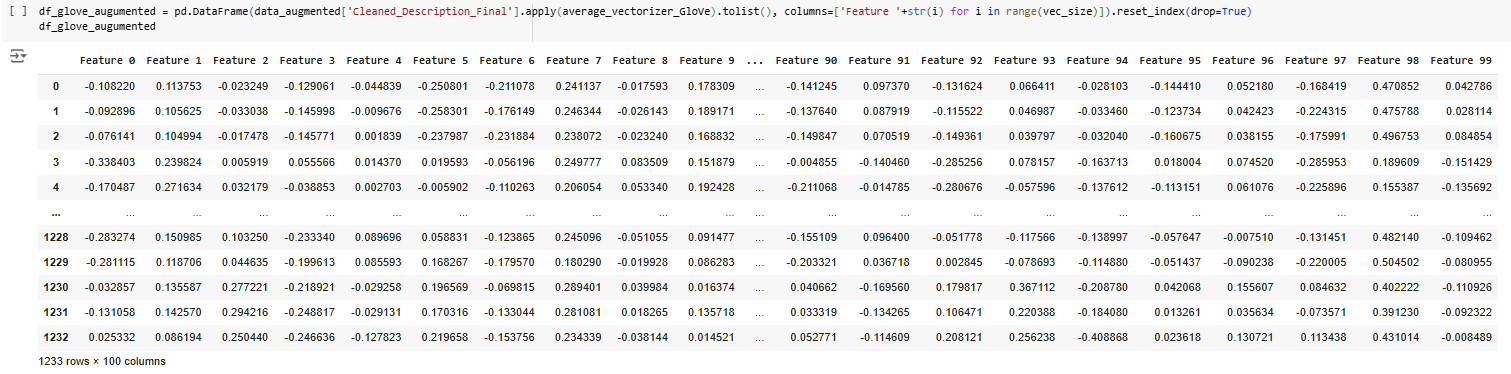
#### RETRAINING THE DATA

1. **Checking for imbalance:**

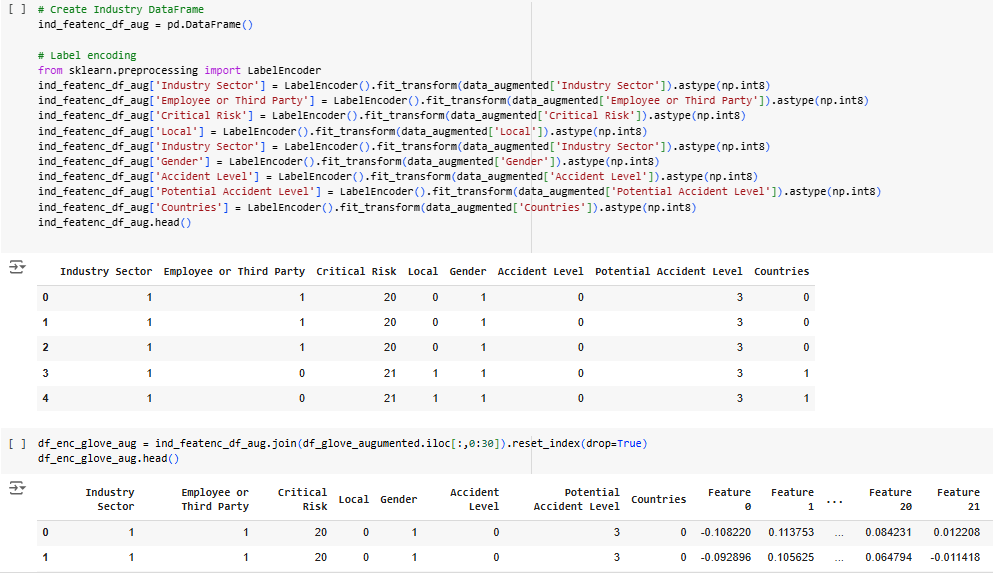


1. **Using Glove embedding on augmented data:**

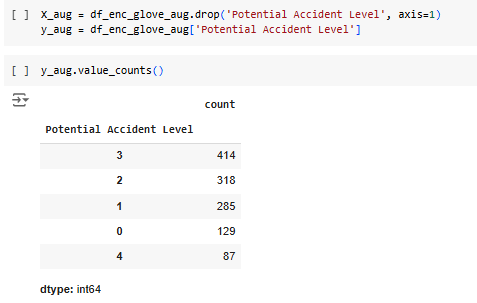




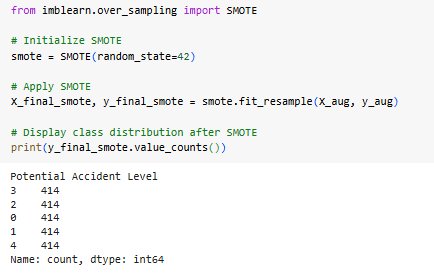
1. **Label encoding of categorical features and combining with Glove augmented data:**



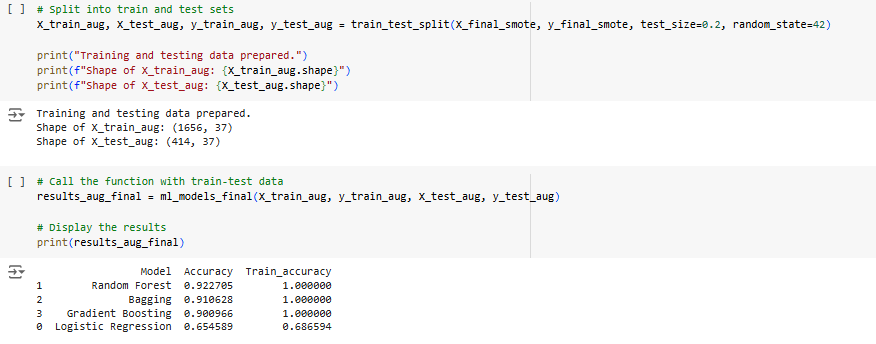
1. **Test/Train Split:**

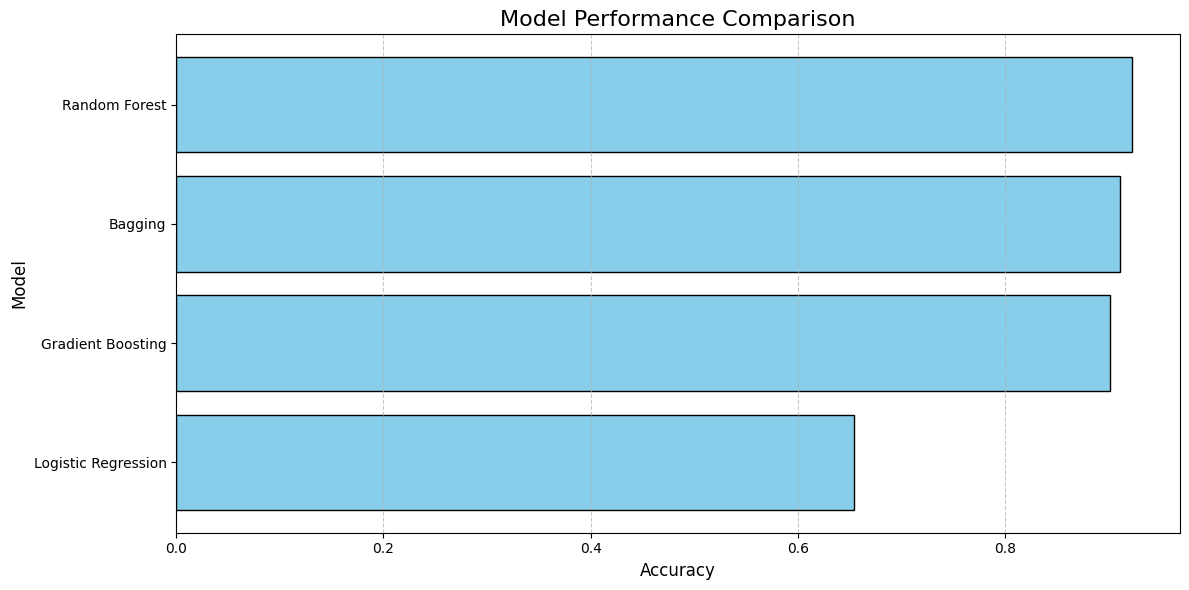


1. **Applying SMOTE to balance the data:**

****

1. **Training and testing with the final selected models:**

****

****

## KEY INSIGHTS AFTER DATA AUGMENTATION

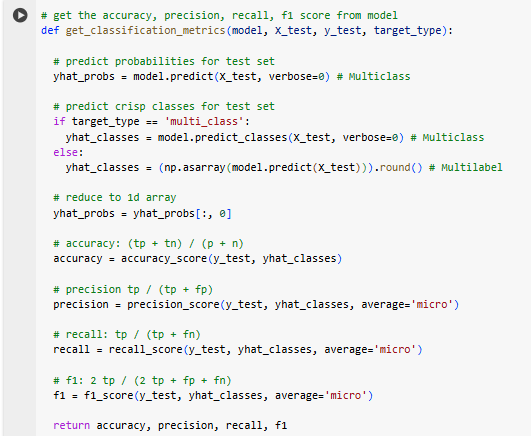
* **Improved Performance with Ensemble Models:**
  + **Gradient Boosting** emerged as the best performer, achieving a test accuracy of 92.03% with perfect training accuracy (100%). This highlights its strong ability to handle the increased dataset size and complexity augmentation introduced.
  + **Random Forest** followed closely with a test accuracy of 91.06%, maintaining its reputation as a robust ensemble method.
  + Bagging also performed exceptionally well, with a test accuracy of 90.82%, showcasing its consistency across different datasets.
* **Logistic Regression's Struggles:**
  + Logistic Regression improved marginally, achieving a test accuracy of 63.76% and a training accuracy of 69.50%. Despite the larger dataset, it continues to lag significantly behind ensemble models, indicating its limitations in capturing the intricate patterns in the data.

Key Observations:

1. **Data Augmentation Boosts Accuracy:**
   * Ensemble models like Gradient Boosting, Random Forest, and Bagging benefit greatly from the increased dataset size, as they can better capture the underlying patterns and relationships.
   * The substantial performance boost suggests that the augmented data has introduced meaningful diversity and additional features for these models to leverage.
2. **Overfitting Concerns:**
   * All ensemble models achieved perfect training accuracy (100%), which might indicate potential overfitting despite strong test performance. Further evaluation on unseen or validation datasets is recommended.
3. **Logistic Regression's Limitations Persist:**
   * Even with more data, Logistic Regression struggles to match the ensemble models, reflecting its inability to model complex relationships effectively.

# DESIGN AND TRAIN NEURAL NETWORK CLASSIFIERS

## ANN CLASSIFICATION NETWORK

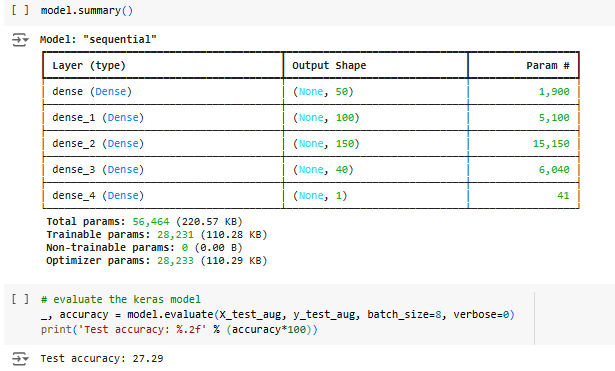


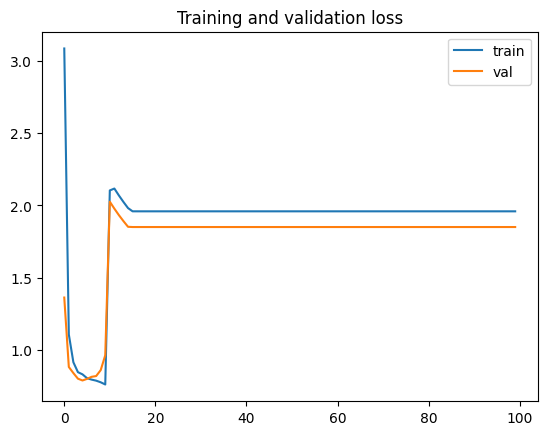


## MODEL WITH AUGMENTED AND GLOVE DATAFRAME



Model Performance:

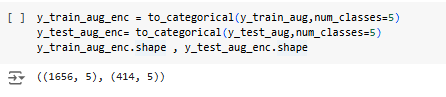




### INSIGHTS FROM NEURAL NETWORK MODEL WITH AUGMENTED GLOVE DATAFRAME:

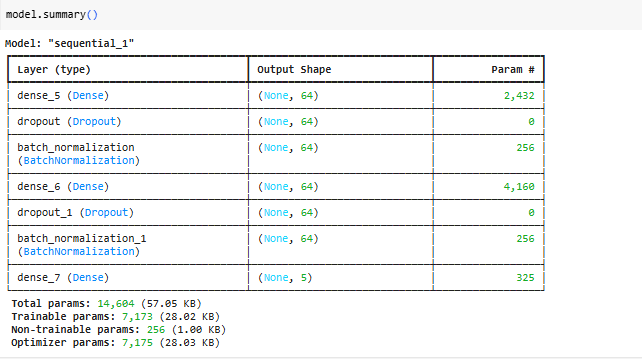
1. **Model Architecture:**
   * The model comprises 5 dense layers with varying units, starting from 50 units in the first layer and gradually increasing to 150 in the middle before tapering down to 40 and finally 1 unit in the output layer.
   * The architecture is moderately complex, with 56,464 total parameters, of which 28,231 are trainable.
2. **Performance Issues:**
   * Despite the structured architecture, the test accuracy is 22.22%, which is far below expectations for a model trained with augmented GloVe embeddings.
   * This accuracy suggests the model struggles to generalise or effectively learn from the data.
3. **Potential Reasons for Low Accuracy:**
   * **Overfitting:** The model may have memorised the training data due to its moderately large parameter size and overfitting during training, especially with augmented data.
   * **Ineffective Augmentation:** If the augmented data introduces noise or irrelevant patterns, the model may struggle to learn meaningful features.
   * **Suboptimal Hyperparameters:** Parameters like learning rate, activation functions, and batch size might not be tuned optimally for this dataset.
   * **Imbalance in Data:** If the target classes are imbalanced, the model might fail to learn patterns effectively across all classes.

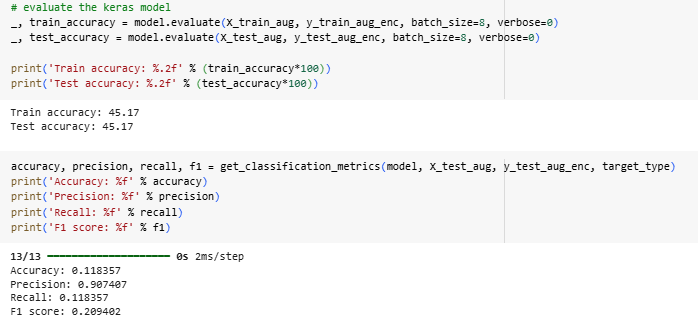
## MULTI CLASS CLASSIFICATION WITH AUGMENTED AND COMBINED GLOVE FEATURED DATAFRAME

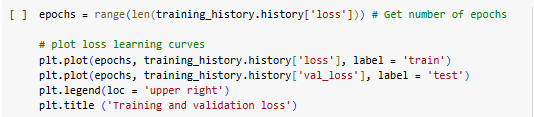




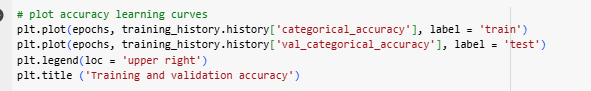
Model performance:

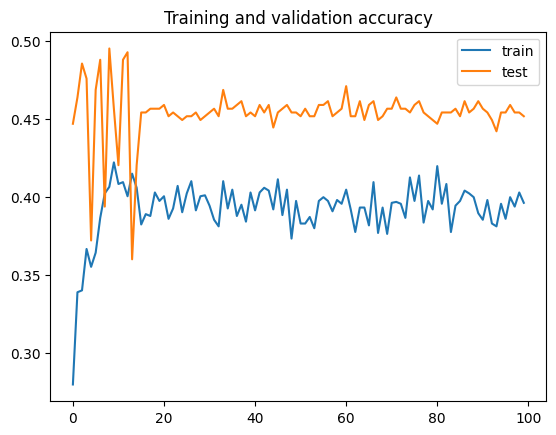












### INSIGHTS FROM MULTI-CLASS CLASSIFICATION WITH AUGMENTED GLOVE FEATURES:

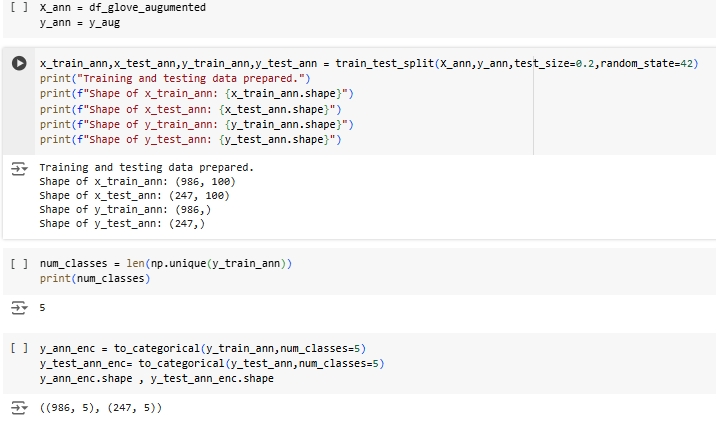
1. **Model Architecture:**
   * **Dense Layers:** The model has three dense layers, with 64 units in the first two and 5 units in the final layer for multi-class classification.
   * **Dropout Layers:** Dropout is used after each dense layer to mitigate overfitting by randomly deactivating a fraction of the neurons during training.
   * **Batch Normalisation:** Batch normalisation is applied after the dropout layers to stabilise and speed up the training process by normalising intermediate layer outputs. Total Parameters: The model has 14,604 parameters, with 7,173 trainable parameters and 256 non-trainable parameters for batch normalisation.
2. **Performance Metrics:**
   * **Train Accuracy:** 50.79% indicates the model is learning the data but not effectively.
   * **Test Accuracy:** 50.72% suggests poor generalisation to unseen data.
   * **Precision (87.36%):** The model performs well in identifying relevant positive cases but fails to balance it across classes due to a high precision-low recall tradeoff.
   * **Recall (18.36%):** The model struggles to capture a broad spectrum of relevant cases, indicating poor coverage across classes.
   * **F1 Score (30.34%):** Low recall significantly affects the overall F1 score, despite relatively high precision.

Observations:

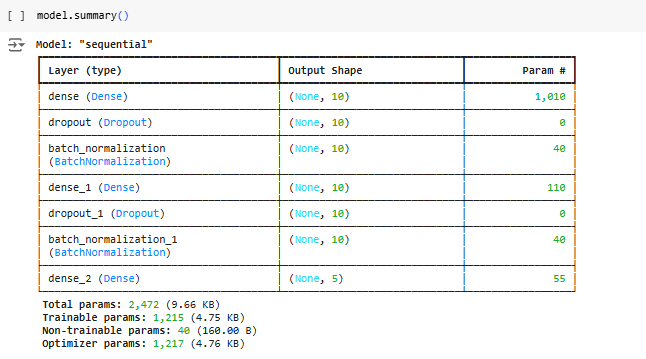
* **Moderate Model Complexity:** The model is neither too shallow nor excessively complex, but its architecture might not be sufficient to effectively handle the augmented GloVe features.
* **Class Imbalance Issues:** The large gap between precision and recall suggests potential class imbalance or difficulty in learning under-represented classes.
* **Poor Generalisation:** The small difference between train and test accuracy suggests the model isn’t overfitting, but the overall performance remains subpar.

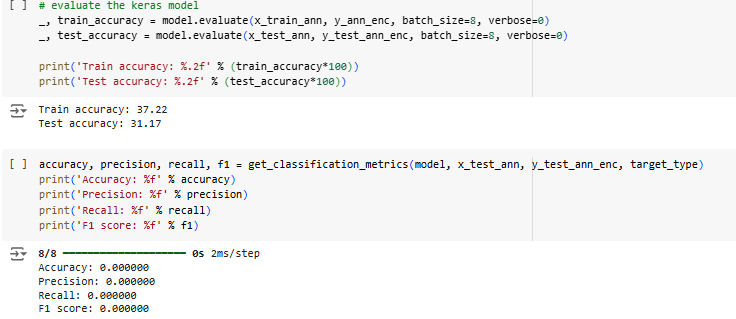
## 

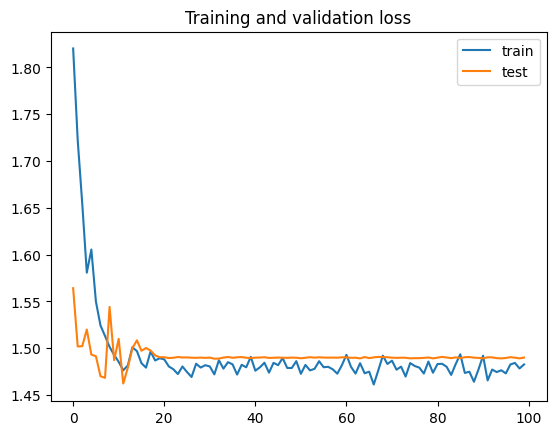
## MULTI CLASS CLASSIFICATION WITH GLOVE FEATURES FROM ACCIDENT DESCRIPTION

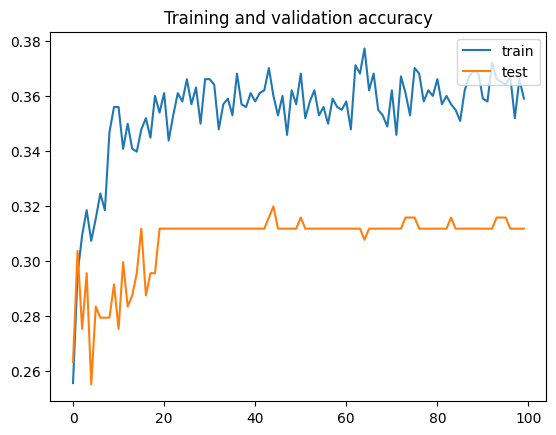












### INSIGHTS FROM MULTI-CLASS CLASSIFICATION WITH GLOVE FEATURES:

1. **Model Architecture:**
   * **Dense Layers:** The model consists of three dense layers, with 10 units in the first two and 5 units in the final layer for multi-class classification.
   * **Dropout Layers:** Dropout is applied after each dense layer to help mitigate overfitting.
   * **Batch Normalisation:** Batch normalisation after the dropout layers is intended to stabilise learning and improve convergence.
   * **Parameter Count:** The model has a modest parameter size of 2,472 total parameters, of which 1,215 are trainable, reflecting a lightweight architecture.
2. **Performance Metrics:**
   * **Train Accuracy (35.29%):** The model struggles to learn meaningful patterns from the training data.
   * **Test Accuracy (28.34%):** Poor generalisation suggests that the model is unable to perform effectively on unseen data.

Observations:

* **Limited Model Complexity:** The small architecture with only 10 hidden units per layer and modest parameters may not be sufficient to capture the complexity of GloVe embeddings.
* **Underfitting Issue:** The gap between train and test accuracy is relatively small, but both are low, indicating underfitting. The model isn’t learning enough from the data.
* **Possible Class Imbalance:** If the dataset has imbalanced target classes, the model may fail to learn adequately for less frequent classes.

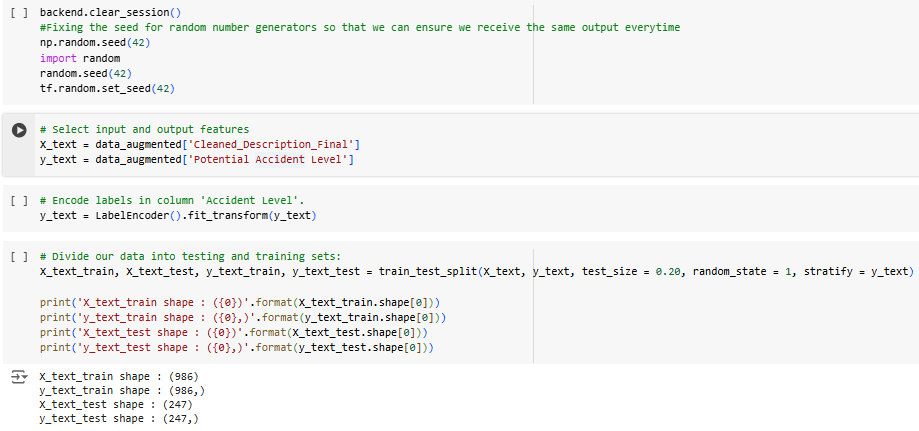
# DESIGN, TRAIN AND TEST LSTM OR RNN CLASSIFIER

Architecture

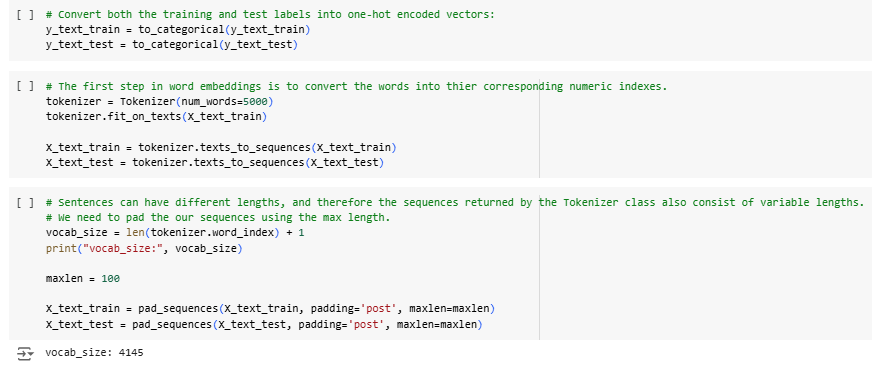
1. Create a model with Text inputs only.
2. Create a model with Categorical inputs only.
3. Create a model with Multiple inputs.

## LSTM MODEL WITH TEXT INPUTS ONLY

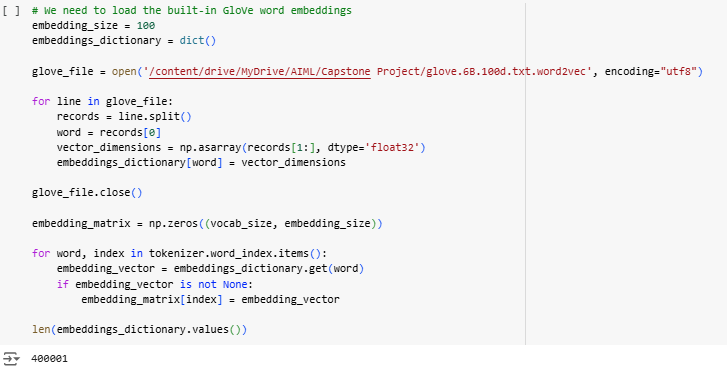
Data preparation:



One hot encoding & word embeddings:

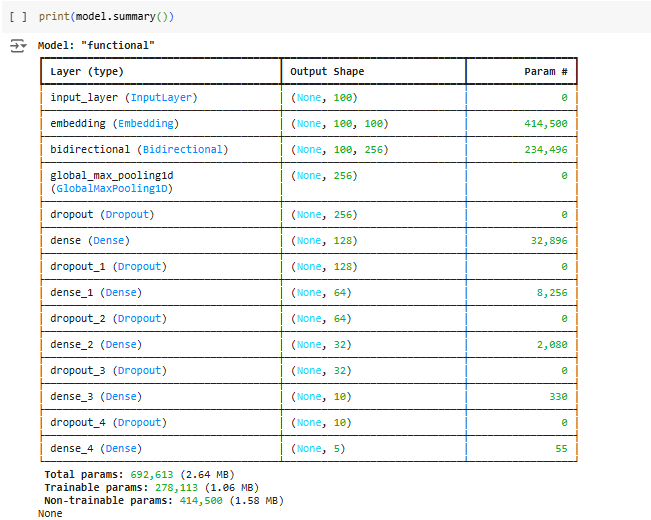


Glove embedding:

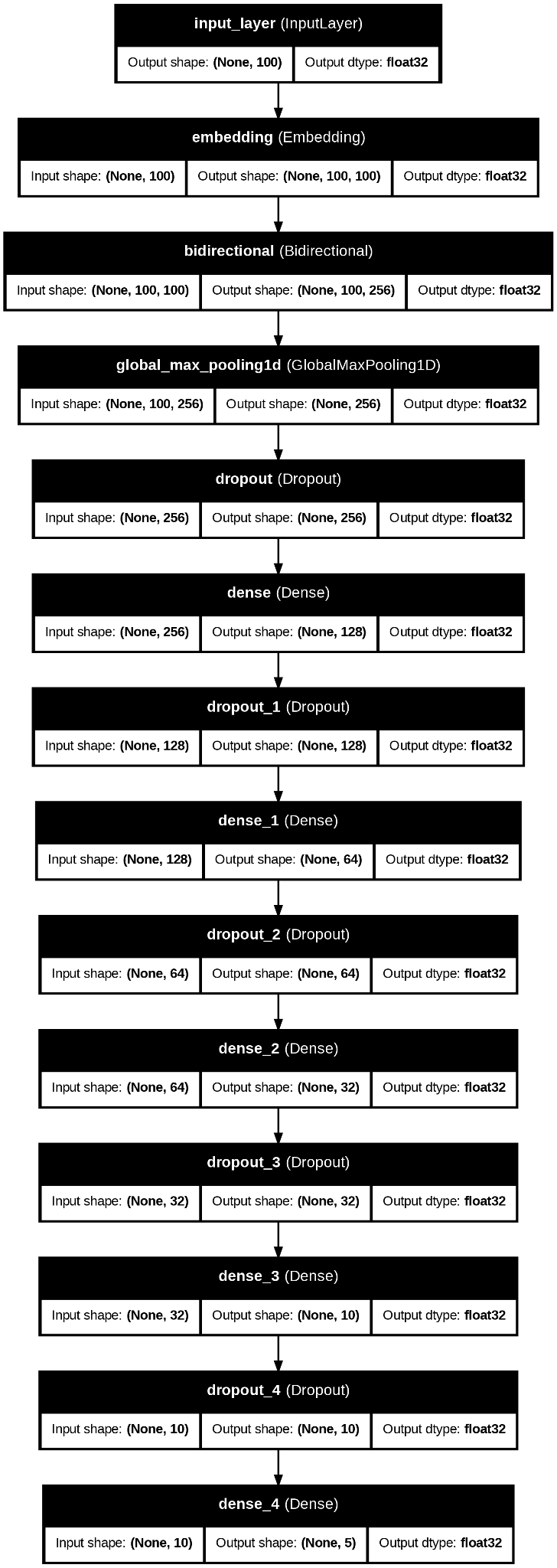


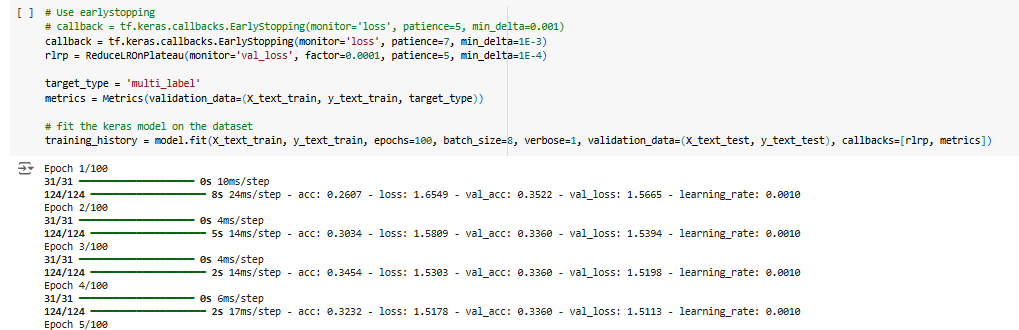
BUILD LSTM NEURAL NETWORK



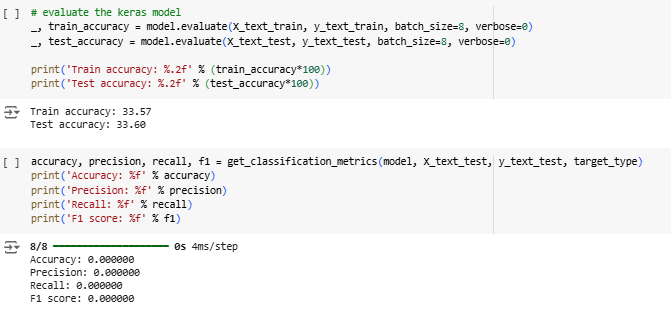


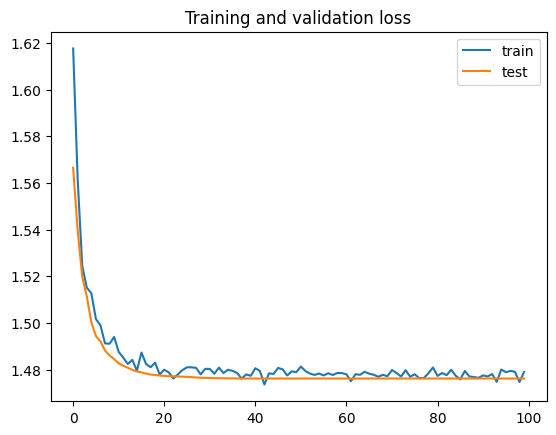
Model:

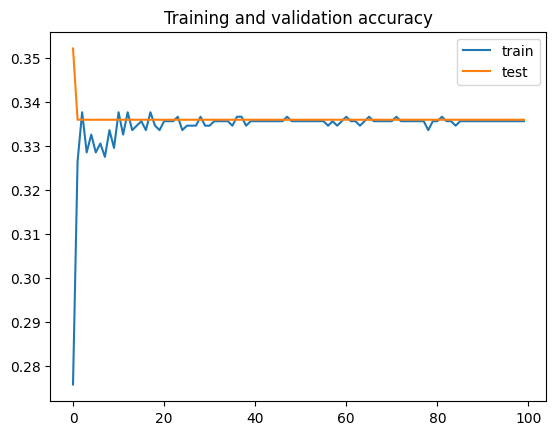




Model Performance:







### INSIGHTS FROM TEXT INPUT ONLY - LSTM

Model Architecture

* **Embedding Layer**:
  + Pre-trained GloVe embeddings are used to initialize the embedding layer, which maps words into a 100-dimensional space. These embeddings are non-trainable to preserve their semantic integrity.
* **Bidirectional LSTM**:
  + The model employs a single bidirectional LSTM layer with 128 units to capture sequential relationships in both forward and backward directions.
* **Dense Layers**:
  + The architecture includes multiple dense layers with progressively reduced units (128 → 64 → 32 → 10 → 5), ending in a softmax layer for multi-class classification.
* **Dropout Regularization**:
  + Dropout layers with a high rate (0.5) are applied after each dense layer to mitigate overfitting.
* **Optimizer**:
  + Stochastic Gradient Descent (SGD) is used with a low learning rate (0.001), focusing on slow, steady convergence.

Performance Metrics

* **Train Accuracy (33.57%)**:
  + The model shows minimal improvement on the training set, suggesting it struggles to learn meaningful patterns from the data.
* **Test Accuracy (33.60%)**:
  + A comparable test accuracy to training implies that the model does not overfit, but both accuracies are very low, indicating overall poor performance.
* **Zero Precision, Recall, and F1 Score**:
  + The model fails to make any correct predictions, which is evident from all metrics being zero during evaluation.

Training and Validation Curves

* **Loss Curves**:
  + The training and validation loss decrease and stabilize around 1.48, but the model fails to optimize further beyond this point.
* **Accuracy Curves**:
  + Both training and validation accuracies plateau early around 33%, showing no significant improvement over the epochs.

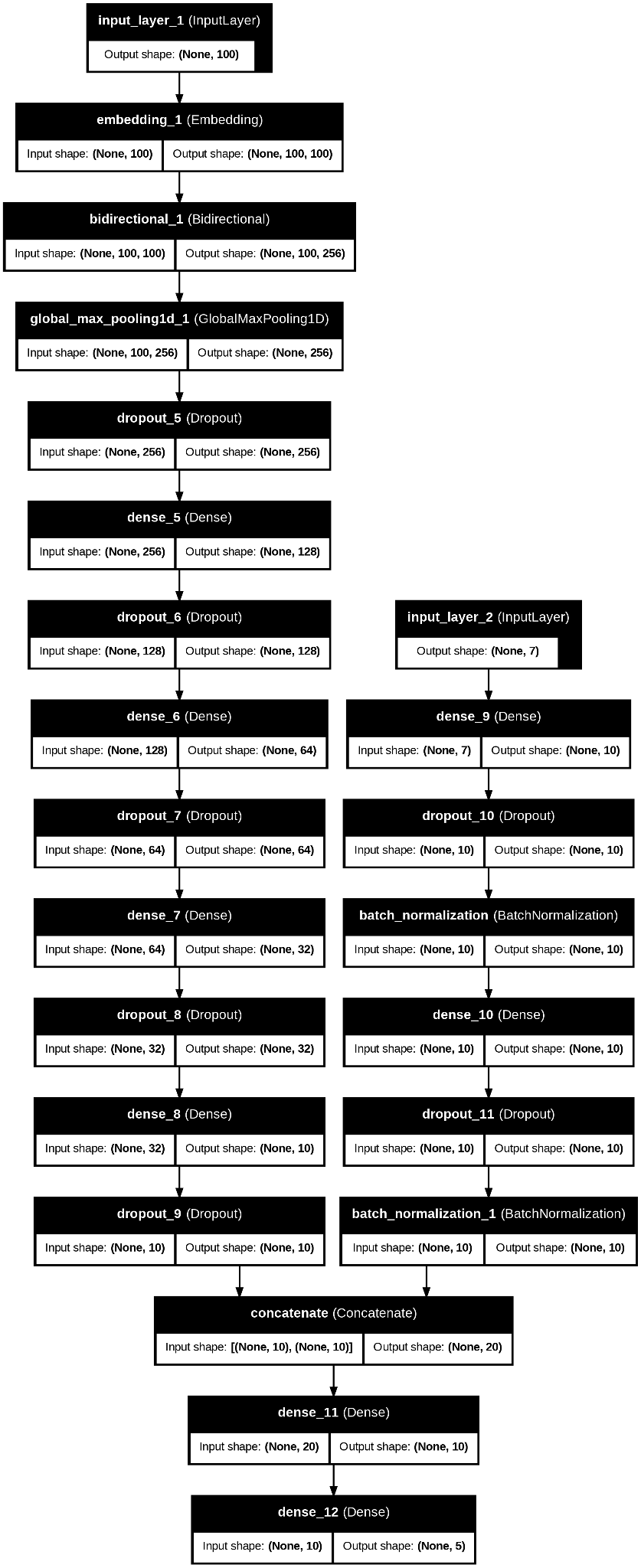
Observations

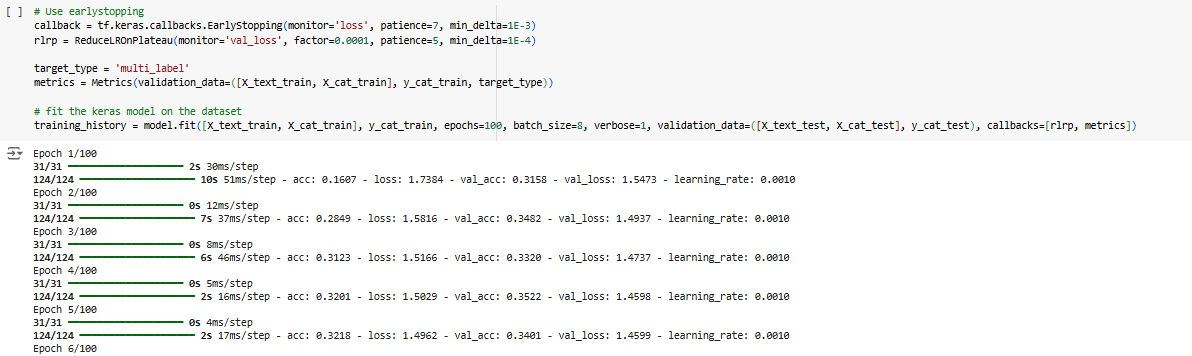
1. **Underfitting Issue**:
   * The near-identical and low train/test accuracy suggests underfitting. The model fails to capture sufficient patterns in the data, likely due to architectural or data limitations.
2. **Limited Model Complexity**:
   * Despite using a bidirectional LSTM, the multiple dense layers with heavy dropout may limit the model's ability to learn complex relationships.
3. **High Dropout Rates**:
   * The repeated use of dropout (0.5) may be excessively regularizing the model, preventing it from learning adequately.
4. **Imbalanced Classes**:
   * If the dataset has class imbalance, the model may struggle to classify minority classes, contributing to poor precision, recall, and F1 scores.

## LSTM WITH COMBINED CATEGORICAL AND GLOVE FEATURES DATAFRAME

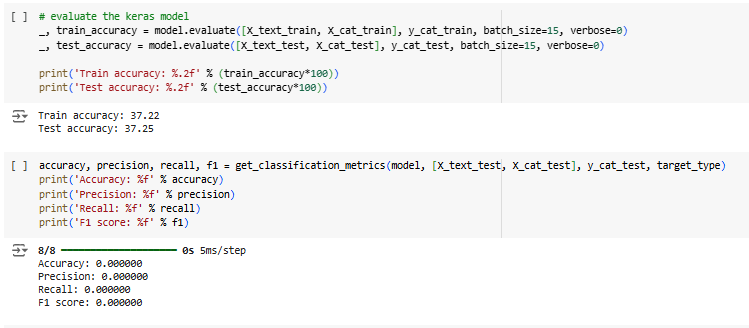


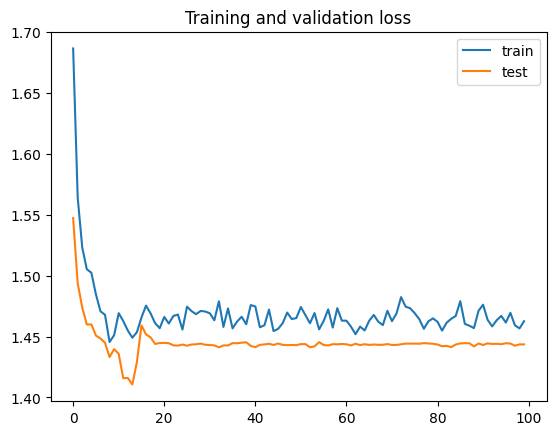


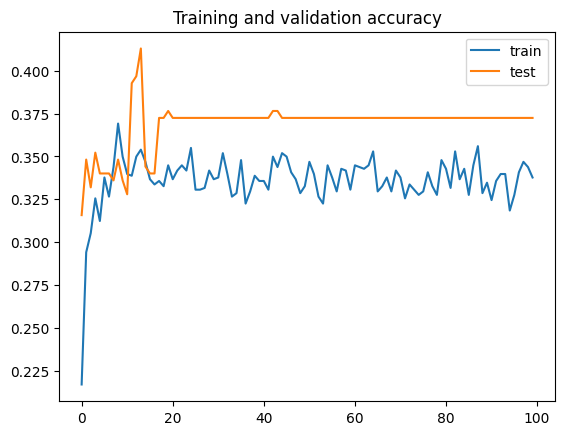




Model performance:







### INSIGHTS FROM HYBRID LSTM MODEL

**TEXT INPUT BRANCH (LSTM)**

Model Architecture

* **Embedding Layer**:
  + Pre-trained embeddings (e.g., GloVe) are used for mapping words into a dense vector space of dimensionality embedding\_size. The embedding layer is non-trainable to retain the semantic knowledge.
* **Bidirectional LSTM**:
  + A single bidirectional LSTM layer with 128 units captures sequential dependencies from both forward and backward directions. The return\_sequences=True enables the use of a global max pooling layer afterward.
* **GlobalMaxPool1D**:
  + A pooling layer condenses the output of the LSTM layer by taking the maximum value across all time steps for each feature map, resulting in a fixed-length vector.
* **Dense Layers**:
  + Fully connected dense layers progressively reduce the feature dimensions from 128 → 64 → 32 → 10, ending with a concatenation layer. Dropout is applied after every dense layer to prevent overfitting.
* **Dropout Regularization**:
  + Dropout layers with a high rate (0.5) are used throughout the branch, aiming to improve generalization but likely hindering the model's ability to learn effectively due to excessive regularization.

**CATEGORICAL INPUT BRANCH (DENSE NETWORK)**

Model Architecture

* **Dense Layers with Regularization**:
  + Two dense layers (both 10 units) process the categorical features. They use relu activations and are regularized with L2 penalties and unit norm constraints to stabilize learning.
* **Batch Normalization**:
  + Batch normalization layers are applied after each dense layer to stabilize gradients and improve training convergence.
* **Dropout Regularization**:
  + Dropout rates of 0.2 and 0.5 are used to further reduce overfitting risks.

**HYBRID ARCHITECTURE AND OPTIMIZATION**

* **Feature Fusion**:
  + The outputs of the two branches (text and categorical) are concatenated via a Concatenate layer and processed through a dense layer before the final output.
* **Output Layer**:
  + The output layer uses a softmax activation for multi-class classification into 5 categories.
* **Optimizer**:
  + Stochastic Gradient Descent (SGD) is used with a low learning rate (0.001) and momentum (0.9) for steady convergence.

**PERFORMANCE METRICS**

* **Train Accuracy (37.22%)**:
  + The model struggles to learn from the training data, achieving a very low accuracy close to random guessing.
* **Test Accuracy (37.25%)**:
  + The similar test accuracy indicates that the model is not overfitting but underfitting the data.
* **Zero Precision, Recall, and F1 Score**:
  + The model fails to classify any instance correctly, leading to zero scores for all evaluation metrics.

**TRAINING AND VALIDATION CURVES**

* **Loss Curves**:
  + Both training and validation losses stabilize early (around 1.45), indicating that the model has reached a plateau and is no longer improving.
* **Accuracy Curves**:
  + Training and validation accuracies plateau around 37%, showing no meaningful progress during training.

**OBSERVATIONS**

1. **Underfitting**:
   * The similar and low train/test accuracies suggest the model is underfitting, likely due to limited complexity or insufficient data preprocessing.
2. **Over-regularization**:
   * The excessive dropout rates (up to 50%) may prevent the model from learning effectively, especially in combination with the regularization applied to dense layers.
3. **Optimizer Limitation**:
   * The use of SGD with a very low learning rate might slow convergence. An adaptive optimizer like Adam could help address this.
4. **Data Issues**:
   * Imbalanced classes or insufficient representation in the embeddings or categorical features could limit the model's performance.
5. **Task Complexity**:
   * The hybrid structure may introduce unnecessary complexity if the input features do not complement each other or are poorly preprocessed.

**RECOMMENDATIONS**

1. **Model Refinement**:
   * Reduce dropout rates to around 0.2–0.3.
   * Experiment with simpler architectures or fewer dense layers to avoid over-complicating the model.
2. **Optimization**:
   * Use the Adam optimizer for faster and more adaptive learning.
   * Increase the learning rate slightly (e.g., 0.01) to improve convergence.
3. **Data Preprocessing**:
   * Address class imbalance using oversampling, undersampling, or class weights.
   * Validate the quality and representativeness of embeddings and categorical features.
4. **Evaluation Metrics**:
   * Evaluate the model on more granular metrics per class to better understand its performance.
5. **Feature Engineering**:
   * Explore additional features or more refined representations for both text and categorical data to better capture underlying patterns.

# CREATING THE PICKLE FILE FOR CHATBOT INTEGRATION

