1. Summary of problem statement, data and findings

This problem is regarding accidents happened in different countries in different industries and we are required to build a system which can provide associated risk based on accident detail.

We are provided with Date of Accident, countries where accident happened with Local detail, Industry sector where accident took place. Apart from this, we are provided detail of person with whom accident took place, like male or female plus employee or third party worker. In addition to that, accident description plus its level and associated critical risk is mentioned with potential accident level as mentioned by authorities, has been provided as part of accident details

All are categorical columns except date and description of accident.

1. There are total 425 rows and 11 columns.
2. Also, there was one 'Unnamed: 0' column, which we dropped.
3. There were no null values found in data.
4. There were 7 duplicate rows found, which we removed and after removal data shape was 418 rows and 10 columns
5. All columns have unique data, but at one cell from Column 'Critical Risk' have Not 'Applicable' data in cell, so this row, we removed.
6. Final Data size comes out to be 417 rows and 10 Columns.

2. Summary of the Approach to EDA and Pre-processing

For EDA univarite and BiVariate analysis was used. 5 number summary was used on word count for Description.

EDA Summary is as follows:

1. Country_01 seems to have largest accident occured, then Country_02 and very less accidents in Country_03
2. Local_03 city is the highest accident prone area
3. Local_01,Local_04,Local_05 also have second largest contribution to accidents.
4. Local_06 and Local_10 have third largest contribution to accidents.
5. Accident Level I has highest percentage.
6. Accident Level I were the maximum occurring throughout all seasons.
7. For Potential Accident Level too, I has hightest percentage, then II and the III.
8. Mining Sector is prone to highest accidents.
9. Metals Sector has second highest accidents
10. Third Party is prone to more accidents than Employee.
11. Males are prone to more accidents than female.

12. Others category of critical risk have more accidents.
13. Year 2016 have more accidents comparetively 2017
14. Feburary month have more accidents
15. After Feburary, March, April and June have more accidents.
16. Thursday and then Tuesday have more accidents compared to other day of week.
17. Day 8 and 4 seems to have more accidents.
18. After that Day 16 and 11 seems to have more accidents.
19. All countries seems to have more of Level I accident but Country_01 seems to have most.
20. Also, only Contry_01 have Level V accident history.
21. Local_03 have more Level I accidents
22. Local_09 and Local_11 have very less number of accident levels.
23. Level V accident is visible only in Local_03, Local_04 and Local_06
24. Mining industry is facing lot of accidents of every level.
25. Male employees met with accidents more and that too of Level 1.
26. Female employees met with accident Level of 1 or Level II.
27. Level 1 accident is borne more by Employee than Third Party.
28. But Level 2, Level 3 , Level 4 and Level 5 accidents happened more to Third Party than to Employee.
29. Accident happened more in November, and that too on thursday.
30. 2016 had most of accidents and that too of Level 1
31. There is maxium of 95 words and minimum of 9

Some features were derived from date, such as

1. 'Day',
2. 'Month',
3. 'Year',
4. 'Weekday' to find out in which month, in which year, in which day of week and in which day of month, employee met with most of accidents. We used this feature to find out the relationship between these with accident level.

We also used glove, to get atmost 50 new features out of Description data provided for accidents.

Before using glove, we preprocess description column with below:

1. Remove non-english words
2. change all in lowerecase
3. remove punctuation
4. remove special characters
5. remove spaces
6. lemmatize words

Also, to clean data, we change all categorical columns to numerical one using label encoder and get_dummies.

4. Overview of the final process

The features of the data which is used for training the neural network:

1. X_train and y_train Shape (339, 42) (339, 6)
2. X_test and y_test Shape (85, 42) (85, 6)

We target encoded columns:

1. Country
2. Local
3. Industry Sector
4. Gender
5. Employee Type
6. Critical Risk

Algorithms Used:

1. Simple Neural Network and using Batch Normalization, Dropout and activation functions such as 'relu' and 'softmax'.
2. Artificial Neural Networks with RBF.
3. LSTM with BiDirectional.
4. Neural Network with LSTM Layers.
5. Neural Network LSTM multiple layer.

5. Step-by-step walk through the solution

Algorithm 1-(optimiser SGD):

Built layers of model with activation 'relu' and 'softmax'. Used kernel initializer 'he_uniform'. Batch Normalization and Dropout were included. Compiled with loss 'categorical_crossentropy'. Train Accuracy 45.13 and Test Accuracy 24.71 .

Algorithm 1- (optimiser Adam):

Built layers of model with activation 'relu' and 'softmax'. Used kernel initializer 'he_uniform'. Compiled with loss 'categorical_crossentropy'. Train Accuracy 63.72 and Test Accuracy 34.12 .

Algorithm 2-(optimiser SGD):

Experimented with Radial basis function (RBF) networks for universal approximation and faster learning speed. Train Accuracy 43.95 and Test Accuracy 28.24.

Algorithm 2-(optimiser Adam): Experimented with Radial basis function (RBF) networks for universal approximation and faster learning speed. Train Accuracy 61.65 and Test Accuracy 28.24

Algorithm 3- (optimiser SGD):

Used Bidirectional LSTM along with 'GlobalMaxPool1D', activation 'relu' and final layer with 'softmax'. Used 'SGD' optimizer. Accuracy for train and test came out to be 34.81 and 29.41. Model is seems good fit as loss in training and validation both loss are decresing and not much gap between them

Algorithm 3- (optimiser Adam): Used Bidirectional LSTM along with 'GlobalMaxPool1D', activation 'relu' and final layer with 'softmax'. Used 'Adam' optimizer. Accuracy for train and test came out to be 57.82 and 31.76. Model is on of the closest as the loss in training and testing are decreasing with less gap

Algorithm 4-(optimiser SGD):

Experimented with embedding size on the LSTM layers with 'SGD' optimizer and loss 'categorical crossentropy'. Train Accuracy 34.81 and Test Accuracy 29.41

Algorithm 4-(optimiser Adam):

Experimented with embedding size on the LSTM layers with 'Adam' optimizer and loss 'categorical crossentropy', removing all dropout layer. Train Accuracy 44.25 and Test Accuracy 30.59. Again this is also one model where we can have the losses minimised but accuracy is too much compromised.

Algorithm 5-(optimiser SGD):

LSTM with modification of multiple layers. Result yielded: Train Accuracy 34.81 and Test Accuracy 29.41. Model is good fit as loss in both training and validation are decreasing to minimal point

Algorithm 5-(optimiser Adam): LSTM with modification of multiple layers. Result yielded: Train Accuracy 69.03 and Test Accuracy 38.82. Model is good fit as loss in both training and validation are decreasing to minimal point

4. Model evaluation

The final model solution that has yielded best result is logistic regression. The objective was to get best accuracy among different algorithms. We kept our eyes on the parameters such as PCA, oversampling and original data. Combination of PCA and oversampling improved our result.

5. Comparison to benchmark

The final solution i.e, Logistic Regression seemed to perform better quite relatively well as compared to the benchmark that we thought to be at the start. After attempting diffrent algorithm models we achieved the best result that we could get.

6. Visualizations

Test accuracy data

Model	PCA with OverSampled Data	PCA with OverSampled Data	OverSampled Data	OverSampled Data	Original Data	Original Data	Hypertuned Original Data	Hypertuned Original Data
Model	Accuracy	F1 Score	Accuracy	F1 Score	Accuracy	F1 Score	Accuracy	F1 Score
---	---	---	---	---	---	---	---	---
SVC	48	48	48	48	35	35	34	34
KNN	46.30	46	49	49	35.29	35	35.29	35
RandomForestClassifier	46.30	46	51.85	52.00	40	40	32.94	33
DecisionTreeClassifier	47.22	47	38.89	39	35.29	35	34.12	34
LogisticRegressionClassifier	49.07	49.00	38.89	39	36.47	36	32.94	33
AdaBoostClassifier	27.78	28	39.81	40	30.59	31	34.12	34
GradientBoostClassifier	47.22	47	43.52	44	28.24	28	_	-

Using SGD as optimiser :-

Model	Train Accuracy	Test Accuracy
Artificial Neural Networks	45.13	24.71

Model	Train Accuracy	Test Accuracy
Artificial Neural Networks with RBF	43.95	28.24
LSTM with BiDirectional	34.81	29.41
LSTM	34.81	29.41
LSTM with multiple layer	34.81	29.41

Using Adam as optimiser :-

Model	Train Accuracy	Test Accuracy
Artificial Neural Networks	63.72	34.12
Artificial Neural Networks with RBF	61.65	28.24
LSTM with BiDirectional	57.82	31.76
LSTM	44.25	30.59
LSTM with multiple layer	69.03	38.82

7. Implications

The accuracy of our solution is around 50 percent and this is the level of confidence of our model. Our model is quite impactful for providing associated risk based on accident detail.

Using adam as our optimiser we were able to attain train accuracy uptill 90-97 also but the model was over fit and thus it was not used to get any inferences.

8. Limitations

Limitation of our model:

1. With an accuracy of around 50 percent, our model falls short on the reliability.
2. The associated risk based on accident detail cannot be precisely measured.

What we can do:

1. Data manipulation of the given csv file through a better angle, which implies merging of the columnar data and dropping the non-benefitting columns from the dataset.
2. We can experiment with more hyperparameters to improve the model.

3. Maybe a new algorithm which we have not tried.

9. Closing Reflections

Learning:

1. We tried our best to play with data and provide whatever insights we could infer, however lack of experience in handling the data limited our model performance.
2. For our model to work on the real world, data needs to be understood and manipulated through different perspectives.
3. The accuracy result is the difficult part, where we have to experiment with different algorithms for our model to be of the best version.
4. We will still try improve our model with changes beyond the Project timeframe.