Monitoring and Predicting Social Unrest

Sumeet Aher
CSE Graduate Student
sumeetmi@buffalo.edu

Krati Sharma
CSE Graduate Student
kratisha@buffalo.edu

Jayesh Suryavanshi
CSE Graduate Student
jayeshki@buffalo.edu

Abstract

Social unrest is a major concern across the globe, what about getting the prediction insights and everyone is prepared to minimize the loss and damage or can avoid the situation completely- many researchers are training the models to get accurate predictions and collecting various data sources to figure out the pattern. In our project we have tried to train multiple models to get the best predictor to predict the event. We have also integrated the authentication of the news reported in ACLED to avoid training our model on rumors. We have linked the model to verify and extract information from various data sources such as Twitter, Bing Search Engine along with the ACLED dataset. There are multiple ideas we have integrated to analyze the news such as watching out the number of days a news is trending, the number of days additional news sources started capturing before it is reported in ACLED, the peak times. As the news can also be biased or politically influenced, we have performed the ground- check analysis to filter out such reports, sentiment analysis was performed and the ACLED dataset was used for benchmarking and evaluation.

1 Introduction

We have performed Information extraction from given texts and classified them as actors, inter, interaction between them, and date by using TfidfVectorizer, Named Entity Recognition, aggregated GloVe Embeddings, Transformer and BERT; the model is trained in Linear SVC, Logistic Regression and Random based classifier. The summary is generated using the ACLED data which was converted in RDF format and was finetuned and trained on T5 (text-to-text transfer transformer). For prediction, additional data sources such as Bing news search API v7, Twitter data have been used to authenticate news and Tf-idf Vectorizer, cosine similarity, the model is a trained Gaussian Naive bayes classifier, SGD (), Logistic Regression, multinomial Naive Bayes- these various models are being used to get the optimized result and perform analysis.

2 Related Work

2015 Mischler - Sentiment Analysis with Prediction of words; Social Networks to validate and improve performance.

2016 Korkmaz - Classification and Categorization, used character level convolutional model.

2017 Qiao - Hidden Markov Model used to predict indicators related to social unrest events. Used root level categories from the GDELT dataset.

2018 Galla - Used Random Forest, predicted 1 month prior and can also predict the tone of the event.

2019 Leen-Kiat Soh - The grouping of related events is often done by SMEs, creating profiles for countries or locations. Proposed two methods for prediction: - 1) spatio-temporal data clustering, and (2) agent-based modeling

2020 Coeckelbergs - Extracted keywords, then matched to an existing thesaurus to enhance the discoverability of holdings. Extracted keywords are proposed to be enriched via language models to further enhance the semantic indexation of archival contents.

2021 Eddie Yang - Showcased how word embeddings differ on concepts that pertain to democracy, equality, freedom, collective action and historical people and events, results demonstrate that these embeddings have downstream implications for AI models using a sentiment prediction task.

3 Methods/Model Architecture

3.1 Task 1 – Information Extraction

For *milestone 2*, we broke down the task as a classification problem for columns such as ***event_type, sub_event_type, inter1*** and ***inter2*** as these have a limited number of classes, unlike ***actor1*** and *actor2* where even though it may look limited but for a particular news, we may have a first of its kind actor such as a new rebel party that has come up in *Brazil*.

For both the scenarios, we have used **GloVe embeddings** for the words. Further, **NER tagging** is being done with **unigram** and **bigram** models to generate labels. Then, trained the model to extract words such as violence, protest, battles, explosions correlated words for Event type and Sub Event Type classification. For date and fatalities we have directly extracted it from notes, fatalities are assumed to be numbers except for the digits mentioned in the dates. Actor 1 and Actor 2 are nouns usually followed by words and based on their extraction we then further extract inter1, inter2 and interaction as classification of those Actors. For classification training purposes, we have experimented using **Linear SVC, Logistic Regression, Random Forest and Multinomial Naive Bayes** classifiers to obtain results and further analyze them.

For *milestone 3*, The model is trained using BERT as it is pre-trained on a huge dataset and helps in classification for event extraction along with transformers,then fine tuned it for NER tagging, we then tested it on the test data.

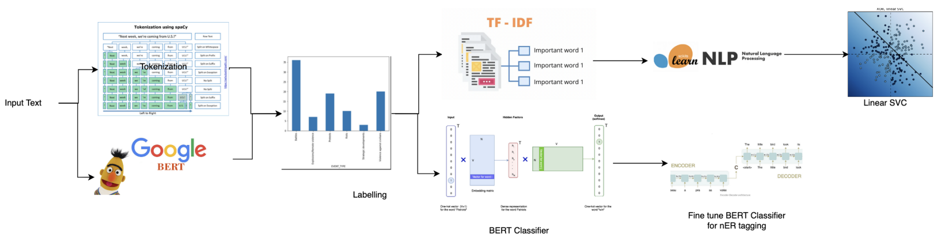


Fig 1: Architecture Diagram

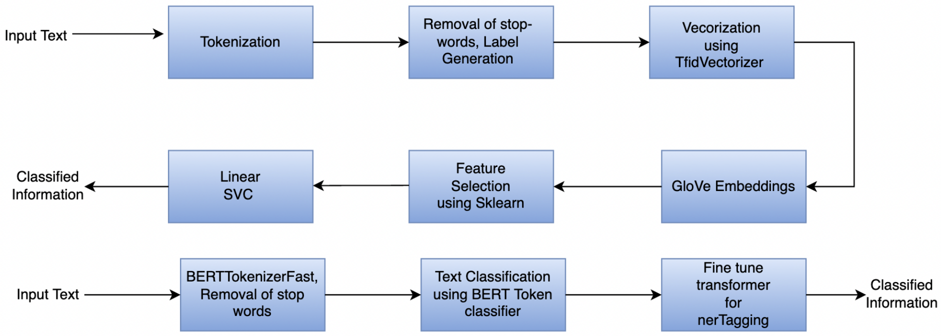


Fig 2: Flow Diagram

3.2 Task 2 – Event Summarization

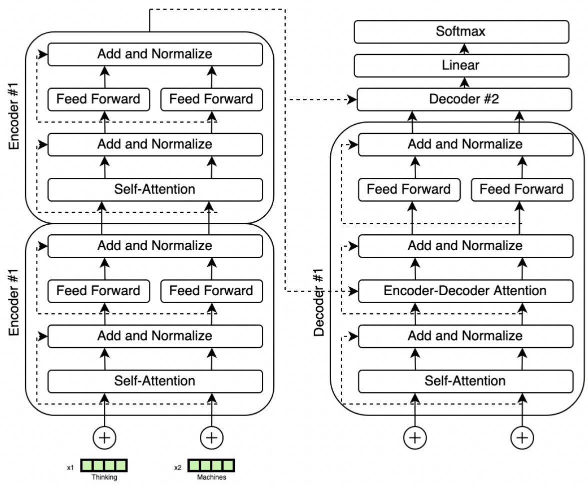


Fig 3: Architecture Diagram

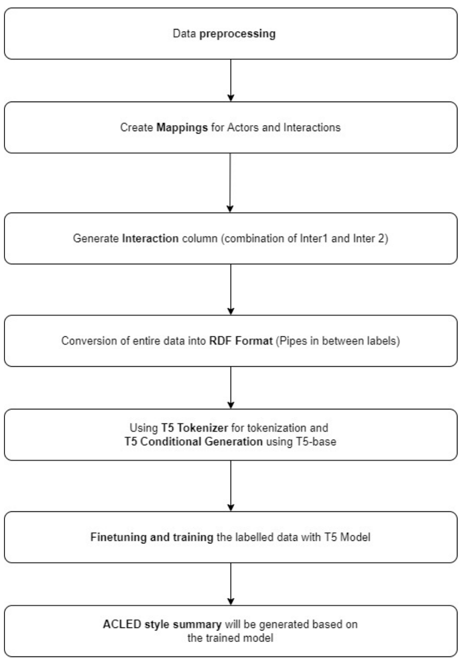


Fig 4: Flow Diagram

Task 2 was a **Natural Language Generation** (NLG) task, hence we wanted to generate a more insightful and intuitive ACLED style summary of the data provided in Task 2, we decided to go ahead with an **abstractive** summarization approach. First, OpenAI's **GPT-2** was a good option as it had powerful text generation capabilities but training it on our data did not give us fruitful results as the model did not converge at all. In addition to this, finetuning them using our use case often resulted in the sentences which were out of context. So, we decided to go ahead with **T5** which is a **Text-To-Text Transfer Transformer** which works in such a way that natural language processing tasks are restructured in a unified text-to-text format where text strings are input and output. How it differs from BERT-based models is that they can only output a span of input or class label. Whereas T5 allowed us to finetune the same model with hyperparameters and loss function on our task. Figure 3 specifies the architecture of the T5 transformer by Google and Figure 4 is the steps followed in order to summarize the data.

3.3 Task 3 – Event Prediction

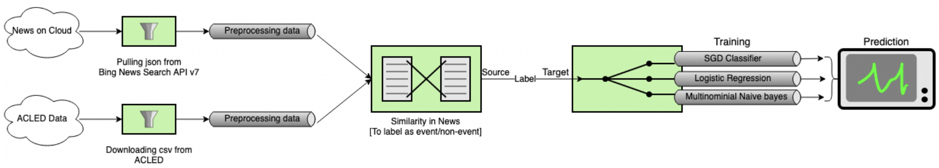


Fig 5: Architecture Diagram

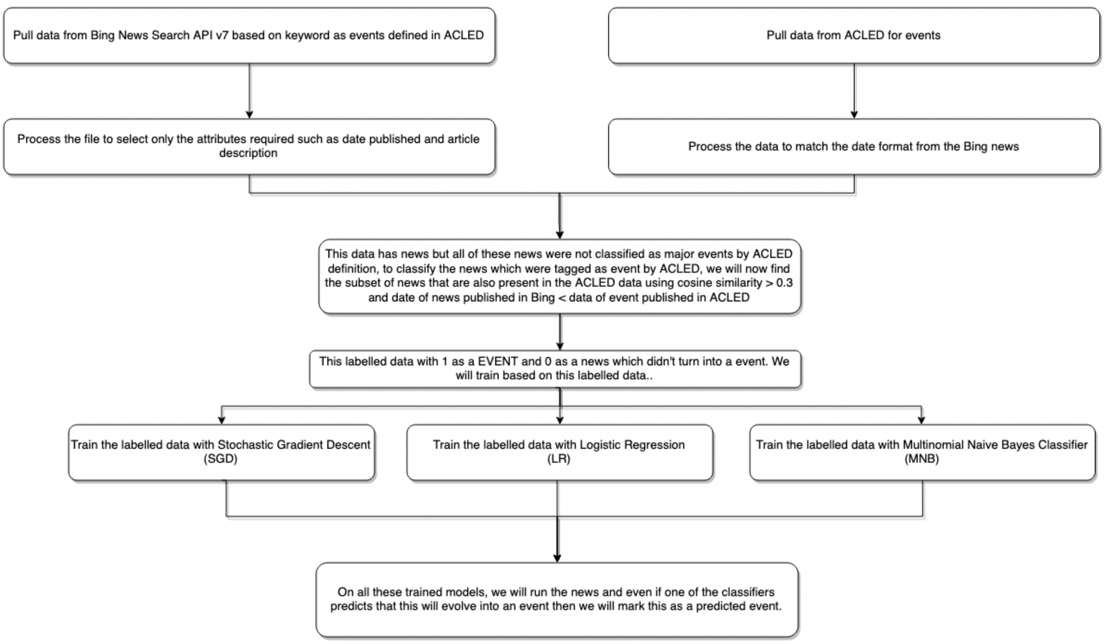


Fig 6: Flow Diagram

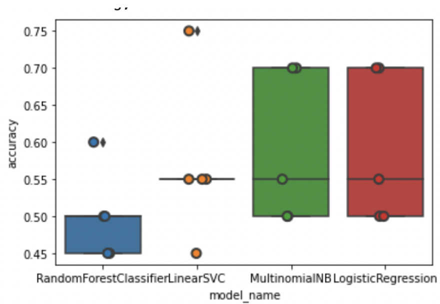
For milestone 2, we used the logic as follows; from the data, we begin with analyzing the distribution of events through locations and time and their frequency. For the text to be transformed into word vectors we used the glove.6B.300d.txt to pick up the already developed embeddings. Then on, we developed a function generate labels which is the crux of the logic which would label the event such as, If an event occurs from i to i + num_days, then the label is 1 else 0. We considered the num_days as 2 and days_window as 5 which is interrupted as the event should occur for a continuation of 2 times in a window of 5 consecutive days. Here on, we used the sentence vectors for training our LSTM model with 4 layers. First layer with 30 neurons and leaky relu as the activation function. Second layer with 15 neurons and LeakyRelu as the activation function. Third layer is just a dense layer with 10 neurons and no activation function. Finally, the output layer with 2 neurons and activated using softmax to give us probabilities for event prediction. We used binary_crossentropy as the loss function and Adam optimizer with 0.001 learning rate.

For milestone 3, we have modified our logic to utilize data from external sources. In this case, we have registered on the Azure cloud to get Bing News api to pull news data from the web. On getting all the news with keywords such as "protest+battles+violence+explosions" and similar words to get a super set of news articles which may or may not have been evolved into an event big enough to be classified as a protest or a riot or a battle. For example, let's say there are talks to prevent a battle, this would also be included in the news but it's not an actual battle. Now, for this pulled Bing News data, we would fetch the ACLED data which has classified events into protest, battles and such categories. So if we find a cosine similarity score of greater than or equal to 0.3 between the bing news and ACLED event notes and if the date_published of that news is before the date_published on ACLEd, then we label this news as 1 else 0. So, to summarize, that particular news actually evolved into a full fledged event. Now once we obtain this labeled data with 0,1 for each news, we train the multiple classifiers such as Stochastic Gradient Descent, Multinomial Naive Bayes and Logistic Regression. Using these trained models we have then concluded with comparisons between trained models.

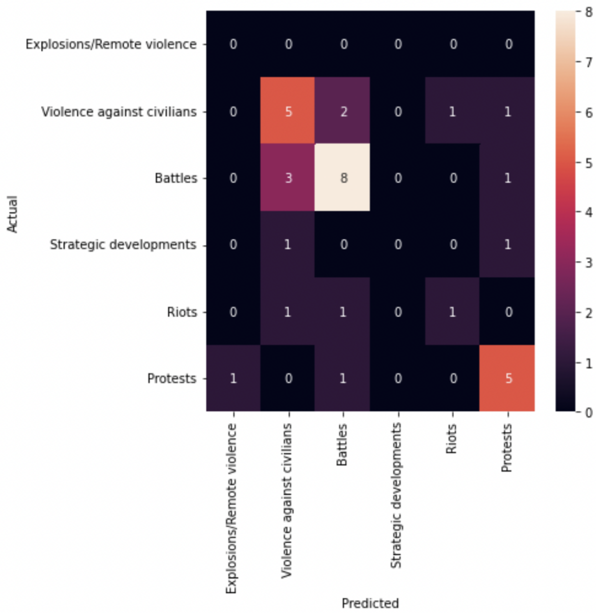
4 Results

4.1 Task 1 – Information Extraction

We see the following accuracies for classification with the highest accuracy for Multinomial Naive Bayes Classifier.



Performance Analysis



Confusion Matrix

The confusion matrix talks about the occurrence of correct classifications against the wrong ones. The more numbers we see on the diagonal, the better the results for that class. This visualization helps to not only understand the good predictions but also the faulty ones.

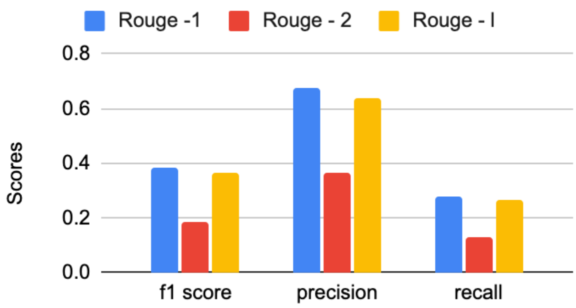
4.2 Task 2 – Event Summarization

```
[50] from rouge import Rouge
     rouge = Rouge()

     rouge.get_scores(o, tokenizer.batch_decode(labelvalidbatch), avg=True)

     {'rouge-1': {'f': 0.385932772890298,
       'p': 0.6737359307359304,
       'r': 0.2798819468075761},
      'rouge-2': {'f': 0.18807510104225045,
       'p': 0.3645579836829838,
       'r': 0.1314133276917336},
      'rouge-l': {'f': 0.36712197313716183,
       'p': 0.6387045870795868,
       'r': 0.2665197767803473}}
```

ROUGE Metrics



Graphical representation of ROUGE metrics

We can observe that we have a low recall score, slightly better f1 scores and precision scores.

4.3 Task 3 – Event Prediction

```
accuracy 0.8504672897196262

my_tags = ['1','0']
print(classification_report(labels_test, y_pred,target_names=my_tags))

             precision    recall  f1-score   support

          1       0.92      0.80      0.85        59
          0       0.79      0.92      0.85        48

avg / total       0.86      0.85      0.85       107
```

SGD Classifier

```
accuracy 0.8504672897196262
             precision    recall  f1-score   support

          1       0.89      0.83      0.86        59
          0       0.81      0.88      0.84        48

avg / total       0.85      0.85      0.85       107
```

Logistic Regression Classifier

```
accuracy 0.8317757009345794
             precision    recall  f1-score   support

          1       0.94      0.75      0.83        59
          0       0.75      0.94      0.83        48

avg / total       0.85      0.83      0.83       107
```

Multinomial Naive Bayes Classifier

We see consistent accuracy and other scores across all algorithms for predicting the events, with the Stochastic Gradient Descent classifier still slightly better than the rest.

5 Discussion and Error Analysis

We have performed Error Analysis to document the errors that appear in the Tasks assigned in the project, determined whether those errors are systematic, and explain what caused them along with the discussion of future scope in respective tasks.

5.1 Task 1

Based on the results, we see that Multinomial Naive Bayes gives the highest accuracy. On observation ,we understand the obtained accuracy is still low compared to benchmarking standards, which may improve on a better dissection of Notes rather than just passing it a vector. For example, for a particular event, we can add a special feature of location which will keep track of places with high social unrest and then extract the information such as event type more towards a riot than a protest. So instead of passing notes as the direct feature, we could first extract location and then pass it with notes with comparatively more weight to the location thus helping the model better predict. Accuracy increased when we used BERT (transformer). As BERT has the limitation of 512 tokens, we had to set the limit to 512 (which may have affected as the description of news articles can cross this number easily), introduced padding and used a drop-out of 0.2.

Further, all the models outperform the test baseline for violent protests on a split of 60% training and 40% validation dataset. Again, random forests showed an improved performance with both training and validation set size of split of 70% and 30% respectively. However, the expectation was that the smaller the train data becomes, the less the performance of the models would be. Surprisingly, random forests performed better with 30% less train data. This could be explained by some of the outliers that were skipped during the training for a particular timezone or location. From the findings it could be concluded that a 60% train set is still able to predict with a lower accuracy the tone of violent protests using the events from the ACLED dataset.

5.2 Task 2

In Task 2, the veracity of the data was really a lot. The train dataset itself had around 100k records. The amounts of attributes associated with these records made the tasks fairly challenging and computation heavy. For this specific subtask, we performed the training for T5 transformer which proved to be fairly compute-intensive and even a single epoch takes around an hour for completion. Hence, the number of epochs were less initially. We have specified epochs in the notebook as it will improve the metrics for summarization currently achieved.

Further, we also tried scraping the news using Source, Event Date and the Event using BeautifulSoup and we could summarize the content of the article in brief. Although, this was done separately, we believe that amalgamation of these two approaches will provide a very insightful and intuitive summary in a couple of sentences.

5.3 Task 3

For task 3, however we develop the logic, it's a challenge to predict if it will be a full blown event or just news for a couple of days. Definitely the result could be improved if we gather the news from all over the possible API's as this will be the ultimate superset of news. Currently, fetching news is a matter of location bias and thus may overshadow a lot of incidents which were politically covered and not famous over the globe even though the local newspapers covered it. So, an exhaustive superset of news from all over the globe would help us improve the current model thus eliminating bias in our training on location.

Having said that, we could also play with the cosine similarity which is labeling the training dataset to check if the news evolved into an event. A cosine similarity of 0.4 or even 0.5 could also improve the results but then risks having wrong predictions. This impacts the training as then we will have more labeled 1 news in our training dataset. We also need to consider news in local languages which is completely new for this current model. For example, the Chinese local newspaper will cover the local news in much detail but because it's in a different language, our model doesn't cover it. We also see some wrong predictions as the news were consistent but probably we should also consider a consistent increase in gravity of that particular event, based on the idea that an evolving event such as a protest should increase in gravity over consistent days.

6 Conclusion

Our project is able to classify the events as Riots,Protest and Battles along with the sub-task, with the fatalities the impact of the event is well demonstrated which could help the government and citizens to be well prepared if there is a prediction of the event. In the future, the impact i.e, density is where we could look forward to predicting it and acting accordingly. We are able to summarize the huge content into meaningful abstract helping people to get the gist of the news around. In the future, with the word limit enabler functionality could be embedded with the summary generator. And finally we are able to segregate the actual news from the rumors by implementing the ground reality check by performing multiple source searches to validate. Further we have enabled our model to predict if the event is going to be converted into a protest or not by monitoring the news for a 2 to 5 day window.

7 Work Distribution

All the authors have contributed equally in the project. Following is the work distribution upon agreement of all the team members.

- **Jayesh Suryavanshi** - Task 1, Task 2, Task 3
- **Sumeet Aher** - Task 1, Task 2, Task 3
- **Krati Sharma** - Task 1, Task 2, Task 3

8 Bibliography

- **Galla, D. and Burke, J.,** 2018, July. *Predicting social unrest using GDELT*. In International conference on machine learning and data mining in pattern recognition (pp. 103-116). Springer, Cham.
- **Agarwal, S. and Sureka, A.,** 2015. *Applying social media intelligence for predicting and identifying on-line radicalization and civil unrest oriented threats.* arXiv preprint arXiv:1511.06858.
- **Korolov, R., Lu, D., Wang, J., Zhou, G., Bonial, C., Voss, C., Kaplan, L., Wallace, W., Han, J. and Ji, H.,** 2016, August. On *predicting social unrest using social media*. In 2016 IEEE/ACM international conference on advances in social networks analysis and mining (ASONAM) (pp. 89-95). IEEE.
- **Mishler, A., Wonus, K., Chambers, W., Bloodgood, M.** (2017). *Filtering Tweets for Social Unrest*. 2017 IEEE 11th International Conference on Semantic Computing (ICSC), 17- 23
- **Brownlee, J.** (2016). *How to Make Baseline Predictions for Time Series Forecasting with Python*. Retrieved from https://machinelearningmastery.com/persistence-time-seriesforecasting-with-python

- **Zambezi, S.,** 2021. *Predicting social unrest events in South Africa using LSTM neural networks* (Master's thesis, Faculty of Science).
- **Leetaru, K., Schrodt, P. A.** (2013, April). *GDELT: Global data on events, location, and tone*, 1979–2012. Paper presented at the ISA Annual Convention.

Acknowledgements

First and foremost, we wish to express our sincere gratitude to our **Dr. Rohini K Srihari** for her constant guidance, inspiration and support throughout the semester. It has been a great honor being her students. She guided us in the right direction and kept us motivated through the whole journey.

We would like to appreciate the help from our Teaching Assistants, **Sougata Saha** and **Souvik Das** for the great recitation videos, support, invaluable advice, and encouragement.

References