D-CLASSIFIED NEWS

Sponsored by: Noonum
Team Members: Frank Chen, Apoorva Shetty, V J Kamala, Tharun Sikhinam

PROBLEM STATEMENT:

- Noonum is a fin-tech AI startup that leverages graphs and NLP to be a knowledge engine for business and finance
- Their current news dashboard is seen in fig 1, which contains both relevant and irrelevant news articles
- The aim of the project is to classify news articles as relevant or irrelevant based on their "market-relevance" and to explain why this classification was made.

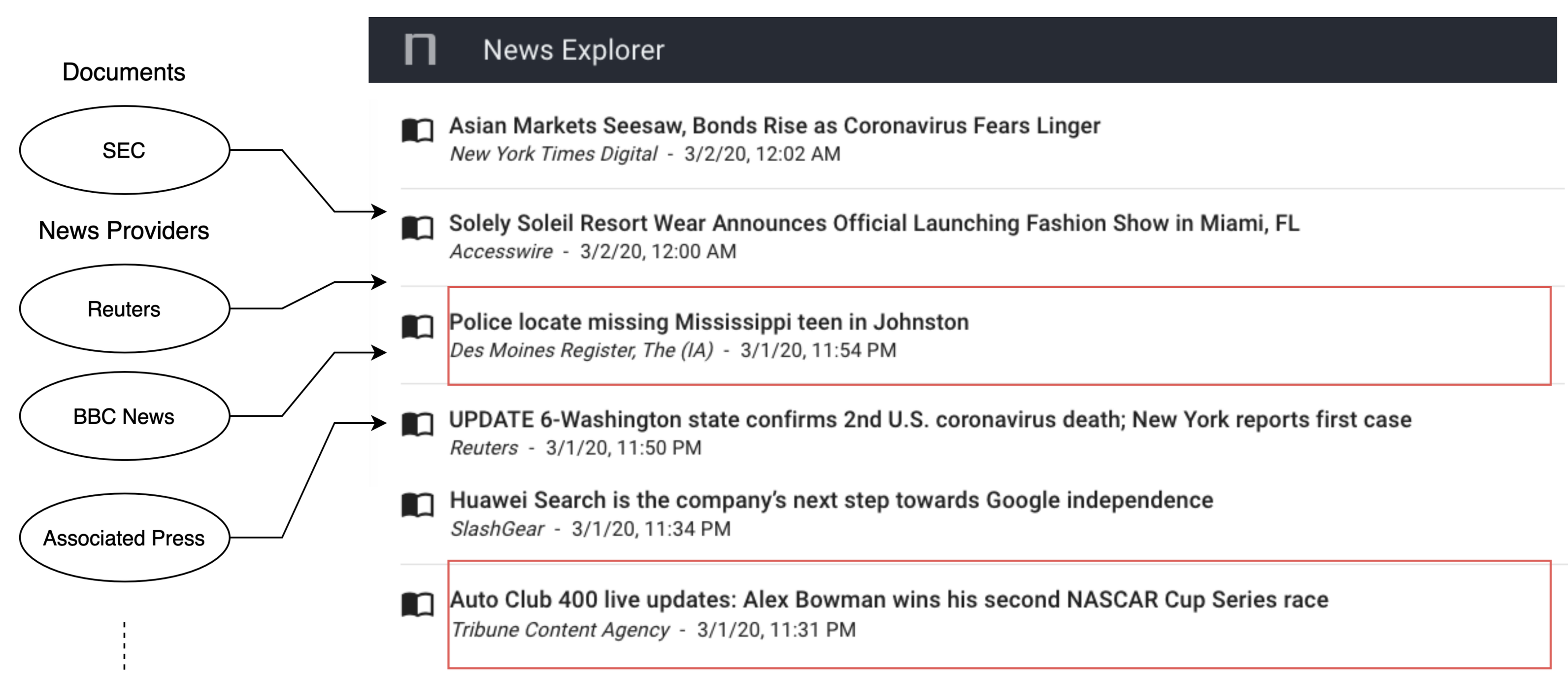
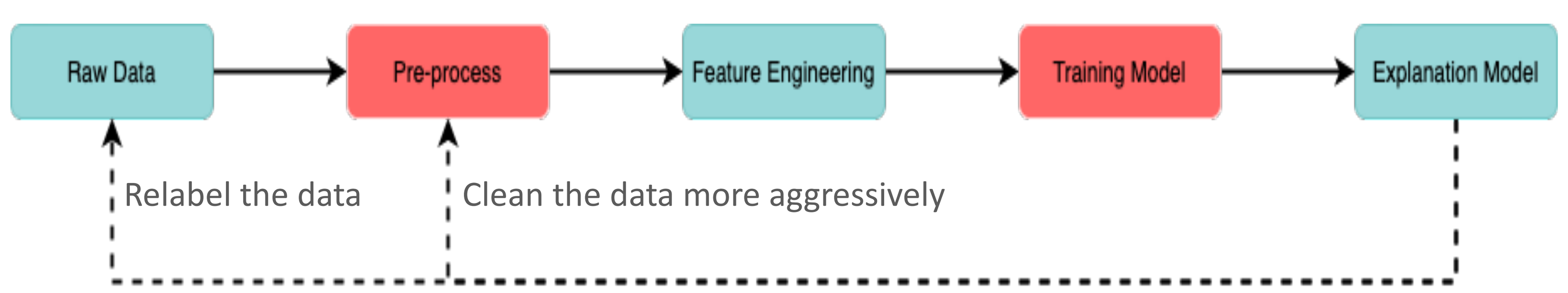


Fig. 1 Noonum dashboard

PROPOSED SOLUTION:



DATA

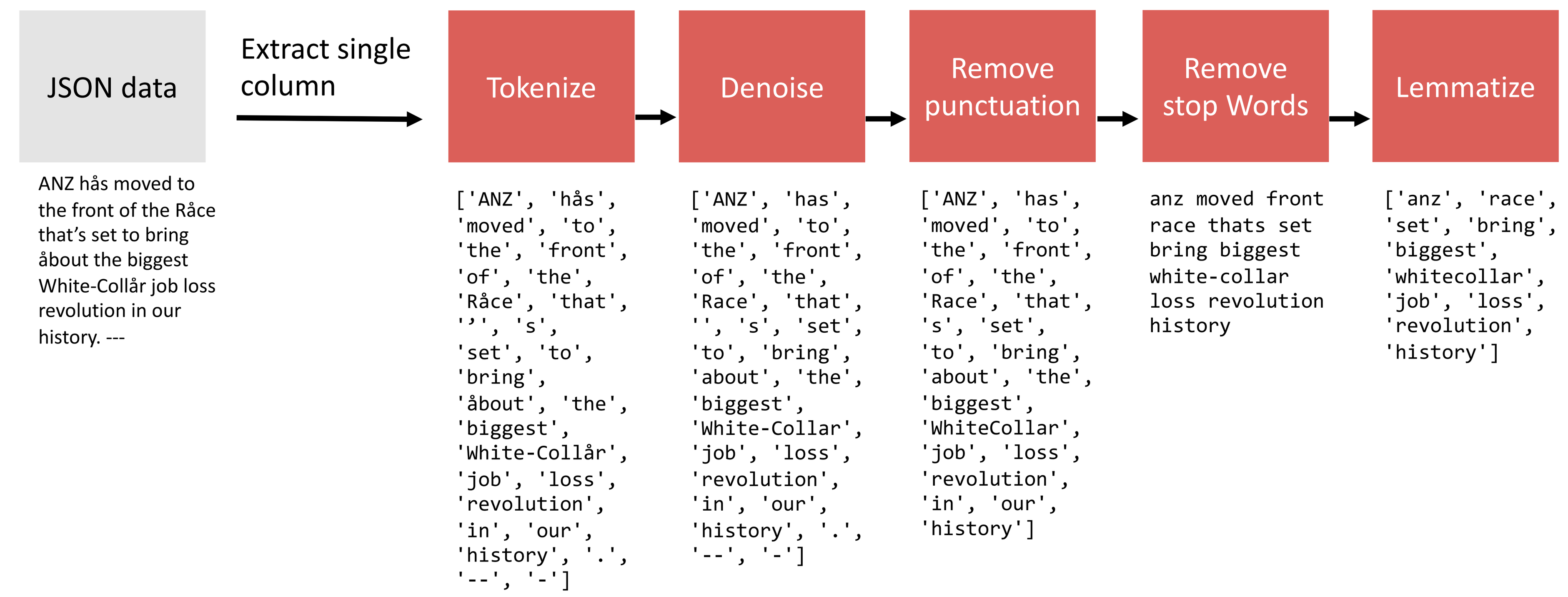
Irrelevant News Articles (10K records)
Relevant News Articles (10K records)
uid <alpha-numerical id>
source <text>
headline <text>
summary <text>
content <text>

- Dataset consists of 10k relevant and irrelevant articles from various news sources
- The dataset contains source, headline, summary and content of a news article
- The news articles were classified by looking for the presence of a publicly traded company in the articles

Relevant

Irrelevant

DATA PRE-PROCESSING



FEATURE ENGINEERING

Bag of Words (BOW)

banks	company	revolution	market	fallout	....
1	1	1	1	0	....

- Counts of each word in the document

TF-IDF

banks	company	revolution	market	fallout	....
0.03538	0.06532	0.00819	0.0213	0.0065	....

- Frequency of a word in current document times how rare the word is in the corpus

Named Entity Resolution

PERSON	ORG	GPE	LOC	PRODUCT	TIME
1	3	1	1	0	0

- Counts of named entity types such as PERSON, LOCATION

Topic Modeling

POLITICS	SPORTS	FINANCE	HEALTH	TECH	CRIME
0.432018	0.00000	0.56055	0.00000	0.000000	0.00000

- Topic Modeling scores after LDA

Embedding

weight_0	weight_1	weight_2	weight_3	....	weight_k
-0.466	0.256	-0.2795	0.08231	....	-0.28593

- Doc2Vec, BERT embedding map text to k-dim vector

TRAINING AND EXPLANATION MODEL

- For the training Model we chose XGBoost, Naïve Bayes and Logistic regression since these classifiers are proven to perform well on text classification from literature survey. To map with these, we had to choose the best model and feature combination, which was found to be:
  Feature: **Bag of Words**
  Model: **XGBoost**
  Classification Accuracy: **74%**
  Precision/Recall: **74%/73%**
- For our explanation model we decided to use the local explanation model LIME, since it gives us a clear idea of which specific words are weighted as irrelevant or relevant

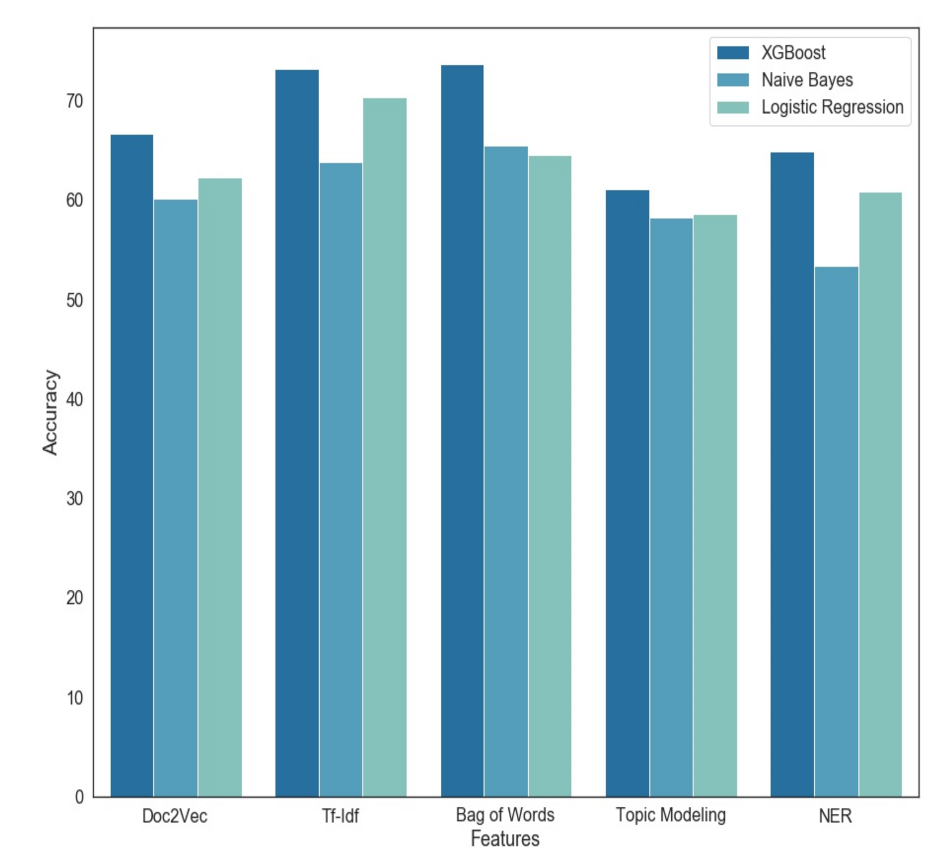
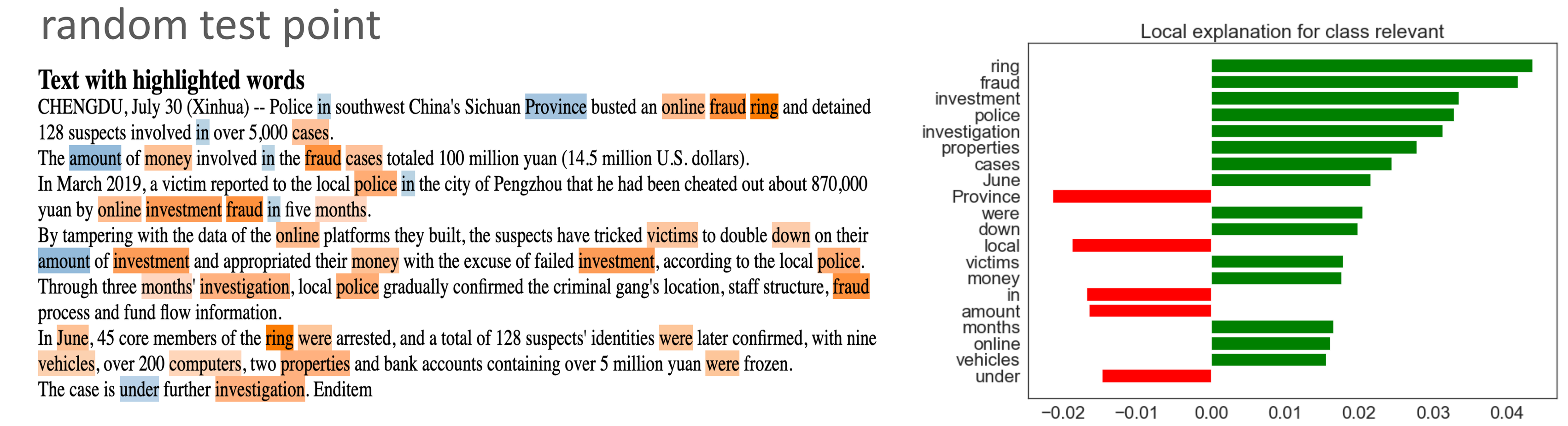


Fig. 2 Model and Feature performance

RESULTS

The following results were generated using LIME on our trained model for a random test point



INSIGHTS

- Running LIME on 100 random test points, we were able to get the frequency of most weighted words in both classes
- Using these insights we were able to clean the data further to get a higher accuracy.

Fig. 3 Iteration 1: Accuracy ~60%

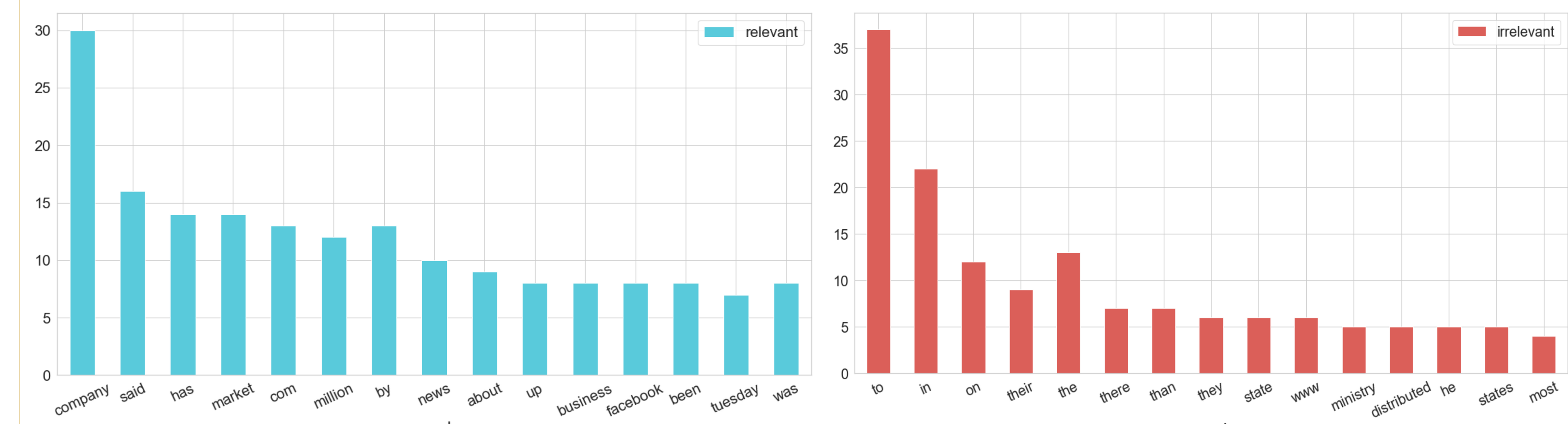
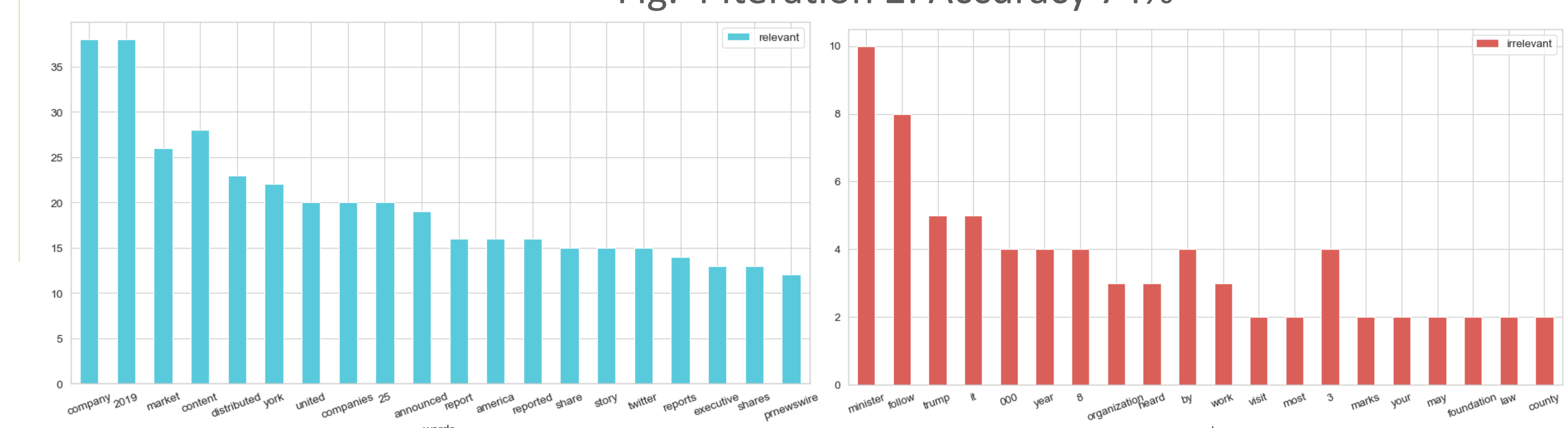


Fig. 4 Iteration 2: Accuracy 74%



CONCLUSIONS & FUTURE WORK

- Simpler feature sets such as BoW and TF-IDF performed well. Aggressive cleaning & pre-processing w.r.t to the context of the application improved the accuracy of the model.
- LIME gave us a clearer picture of what our model was truly learning; indicating that the market relevant terms were being captured by the models.
- The task lying ahead is to provide for a feedback loop to re-incorporate what we learn from our explanation into labelling the data, and also into our data pre-processing and feature-engineering steps.