Exploring Text Classification Techniques:

Disaster Tweet Classification

Jong-Cheol Lee

Department of Statistics
Pusan National University

September 23, 2024

Contents

Project Background and Objective

Background

- Real-time emergency reporting via smartphones.
- But it's unclear if words indicate an actual disaster.

Objective

Goal

Build a machine learning model to classify **"real disaster tweets"**.

Data

Dataset Variables Information

Column	Description
id	A unique identifier for each tweet
text	The text of the tweet
location	The location the tweet was sent from (may be blank)
keyword	A particular keyword from the tweet (may be blank)
target	Whether a tweet is about a real disaster (1) or not (0)

Data Overview

	shape	id	keyword	location	text	target
Train set	(7613,5)	*1,4,5,..*	*wildfire*	*UK*	*Wow! Cooool :)*	*0 or 1*
Test set	(3263,4)	*0,2,3,..*	*crash*	*Italy*	*#Flood in US..*	

Text Preprocessing Steps

Original Text

http://t.co/GKYe6gjTk5 Had a #personalinjury accident this summer? Read our advice & see how a #solicitor can help. #OtleyHour

Step 1: URL Removal

"http://t.co/GKYe6gjTk5"

Step 2: HTML Form Removal

"&"

Step 3: Punctuation Removal

"?", "."

Step 4: Stopword Removal

"Had", "a", "this", "our", "and", "see", "how", "can"

Final Processed Text(lowered)

#personalinjury accident summer read advice #solicitor help #otleyhour

Lemmatization and Tokenization

1. Morphological Analysis Result:

('#personalinjury', 'n'), ('accident', 'n'), ('summer', 'n'), ..., ('help', 'v'), ('#otleyhour', 'v')

2. Lemmatization Examples:

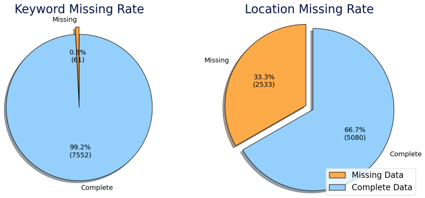
- Original: `"better"` → Lemmatized: `"good"` (Adjective)
- Original: `"running"` → Lemmatized: `"run"` (Verb)
- Original: `"geese"` → Lemmatized: `"goose"` (Noun)

3. TweetTokenizer Characteristics and Result:

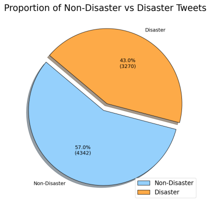
- TweetTokenizer preserves hashtags and mention.(ex. #, `@`)

'#personalinjury', 'accident', 'summer', ..., '#solicitor', 'help', '#otleyhour'

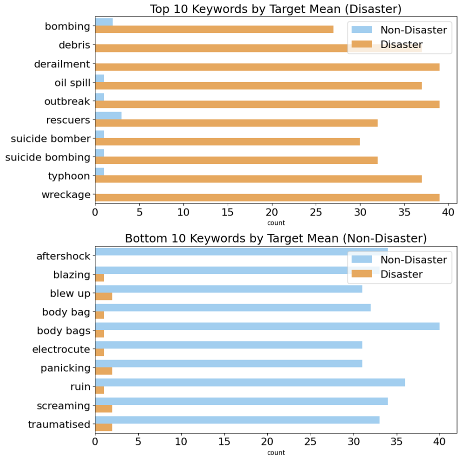
EDA - Missing values & Target distribution



Missing Rate

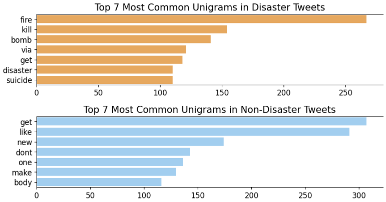


Target Proportion

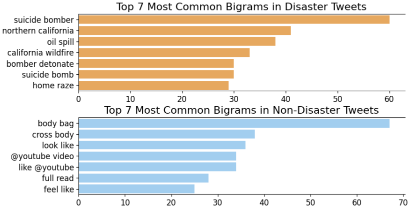


Top&Bottom 15 Keywords by Target Mean

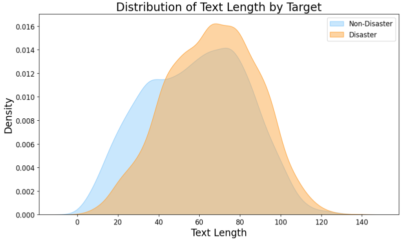
EDA - Word Count & Text Length

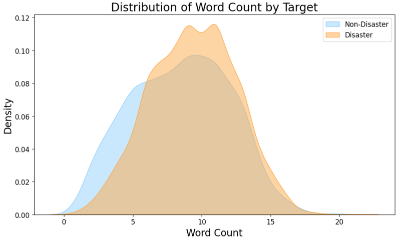


Most Common Unigrams



Most Common Bigrams



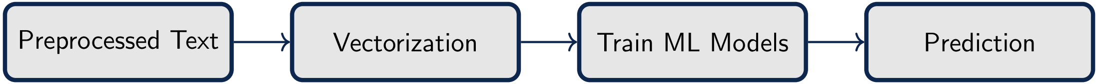


Text Vectorization

Why Vectorize Text?

- ML models require **numerical data**.
- Transforms unstructured data into a **structured format**.
- **Better understanding** by vectorization.

Flowchart



Count-based methods

Bag of Words(BoW)

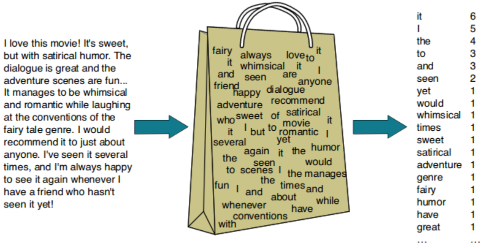


Image of BoW

Sentence	he	she	is	very	good
He is very good	1	0	1	1	1
She is good	0	1	1	0	1

Term Frequency-Inverse Document Frequency (TF-IDF)

Term Frequency(TF):

$$TF(t, d) = \frac{\text{Term frequency of } t \text{ in document } d}{\text{Total terms in } d}$$

Inverse Document Frequency (IDF):

$$IDF(t) = \log\left(\frac{\text{Total \# of documents}}{\text{\# of documents containing } t}\right)$$

$$\textbf{TF-IDF(t, d)} = TF(t, d) \times IDF(t)$$

Sentence	he	she	is	very	good
He is very good	0.6	0.0	0.4	0.6	0.4
She is good	0.0	0.7	0.5	0.0	0.5

Embedding methods

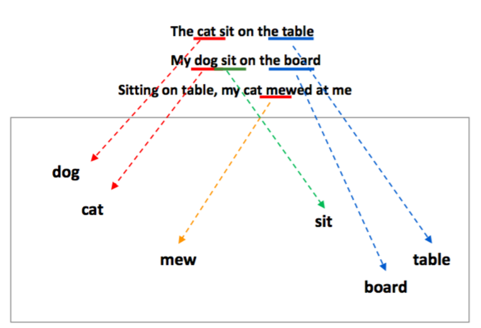
Word2Vec

Frequency based:

- **BoW**: Only count.
- **TF-IDF**: Only count and importance.

Why Word Embedding?

- **Contextual Meaning**
- **Dimensionality**
- **Semantic Relations**



Word Representation on Vector Space

Final Dataset & Classification Models

Final Dataset:

	id	combined_str	target
0	1	deed reason #earthquake may allah forgive	1
1	4	forest fire near ronge sask canada	1
2	5	resident ask shelter place notify officer evac...	1
3	6	13000 people receive #wildfires evacuation ord...	1
4	7	get sent photo ruby #alaska smoke #wildfires p...	1
...	...	...	...
7608	10869	two giant crane hold bridge collapse nearby home	1
7609	10870	@ariaahrary @thetawniest control wild fire cal...	1
7610	10871	m194 0104 utc 5km volcano hawaii	1
7611	10872	police investigate ebike collide car little po...	1
7612	10873	late home raze northern california wildfire ab...	1

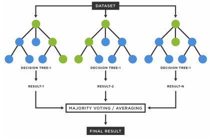
7613 rows × 3 columns

Training Set

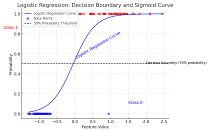
- Adding **keyword**
- Excluding the **location**

Classification Models:

1. Random Forest



2. Logistic Regression



3. Emsemble of 2 Models (Soft Voting)

Results

Vectorization Method	Model	Accuracy
BoW	Logistic	0.76003
TF-IDF	Logistic	**0.79129**
Word2Vec	Logistic	0.77198
BoW	RF	**0.79681**
TF-IDF	RF	0.78516
Word2Vec	RF	0.76064
BoW	Logistic + RF	**0.80294**
TF-IDF	Logistic + RF	0.80049
Word2Vec	Logistic + RF	0.77811

Table: Performance Comparison of Text Classification Models

Discussion

Limitations

- **Location Variable**
- **Diverse Preprocessing**
- **Advanced Methods**: like *BERT* with *LSTM* (0.80508) or *1D-CNN* (0.80661)

The End!