Binary Classification on DiasterTweets Texts

designed by:
Lanshi FU
Pierre QIU
Zhe huang

Hands on Natural Language Process

Faculté de Science,Université Paris-Saclay
2023-02-23

Outline

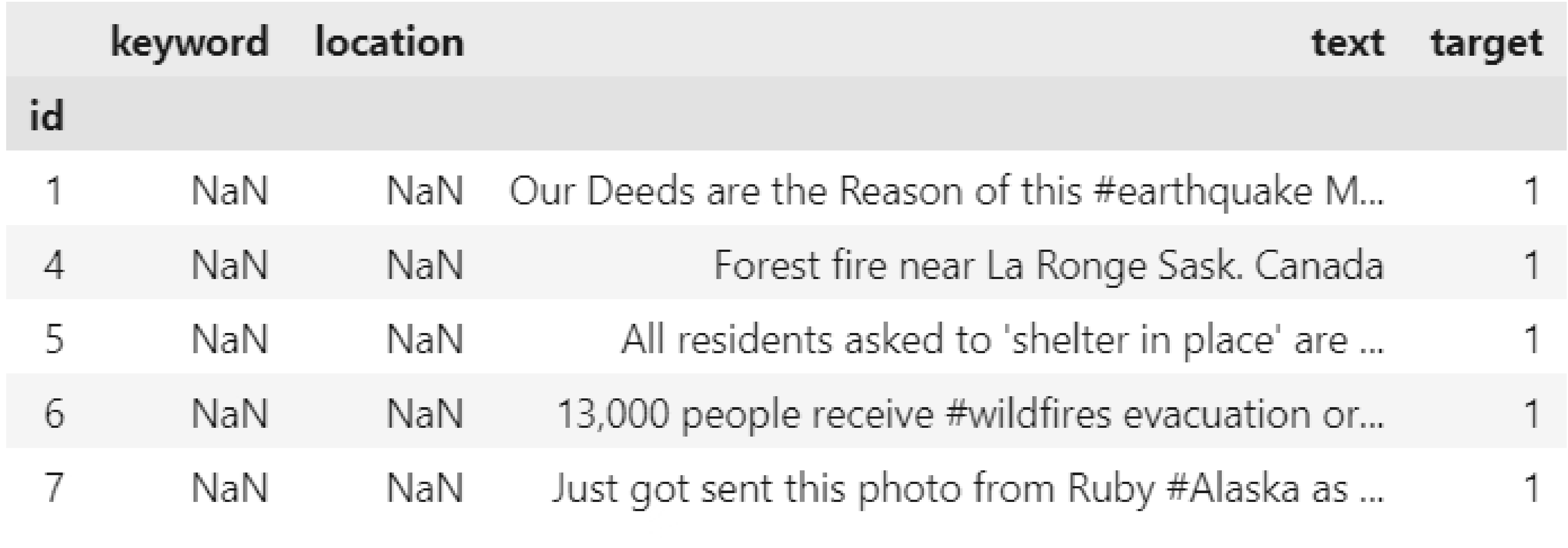
- Introduction of problem
- Have a look on dataset
- Data preprocessing
- Basic approach: Word2vec
- Approach of best performance: Fasttext
- Advanced approach: Bert

Introduction of Problem

Have a look on the dataset

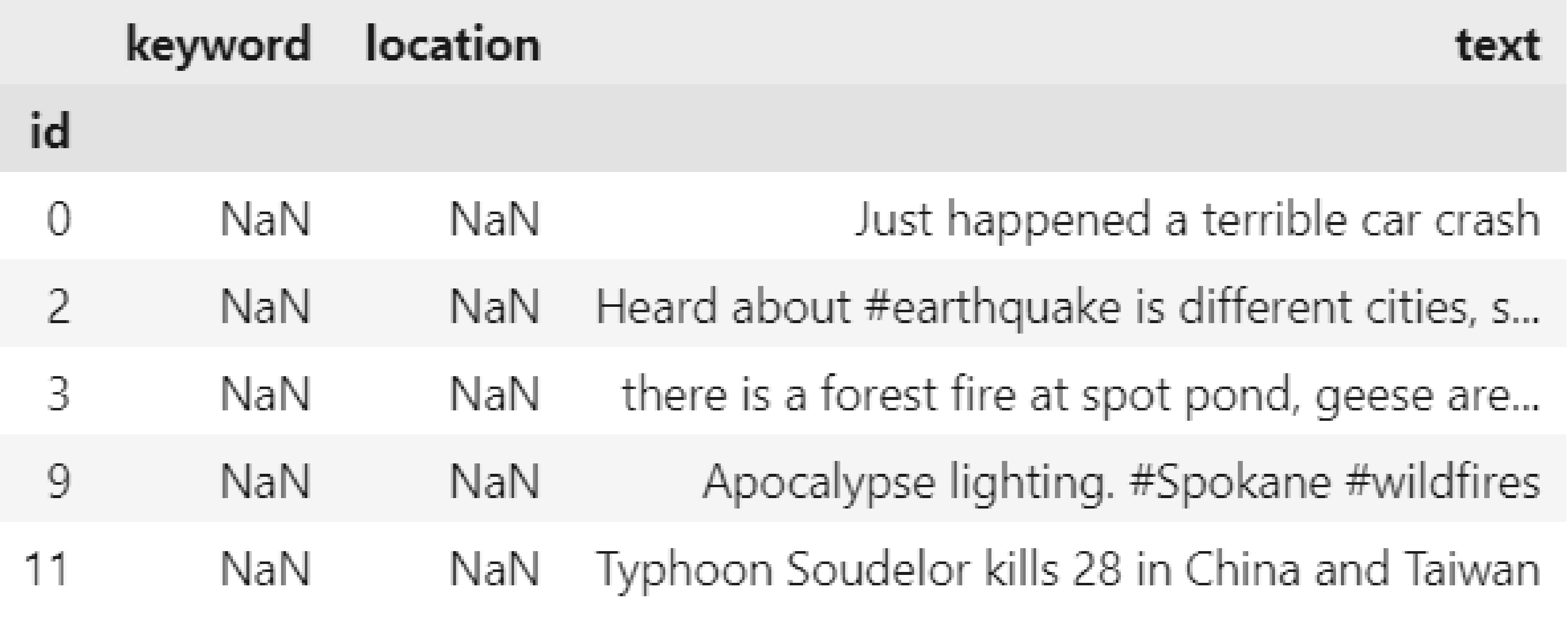
Have a look on the dataset

- id - a unique identifier for each tweet
- texts- 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 - in train.csv only, this denotes whether a tweet is about a real disaster (1) or not (0)



	keyword	location	text	target
id				
1	NaN	NaN	Our Deeds are the Reason of this #earthquake M...	1
4	NaN	NaN	Forest fire near La Ronge Sask. Canada	1
5	NaN	NaN	All residents asked to 'shelter in place' are ...	1
6	NaN	NaN	13,000 people receive #wildfires evacuation or...	1
7	NaN	NaN	Just got sent this photo from Ruby #Alaska as ...	1

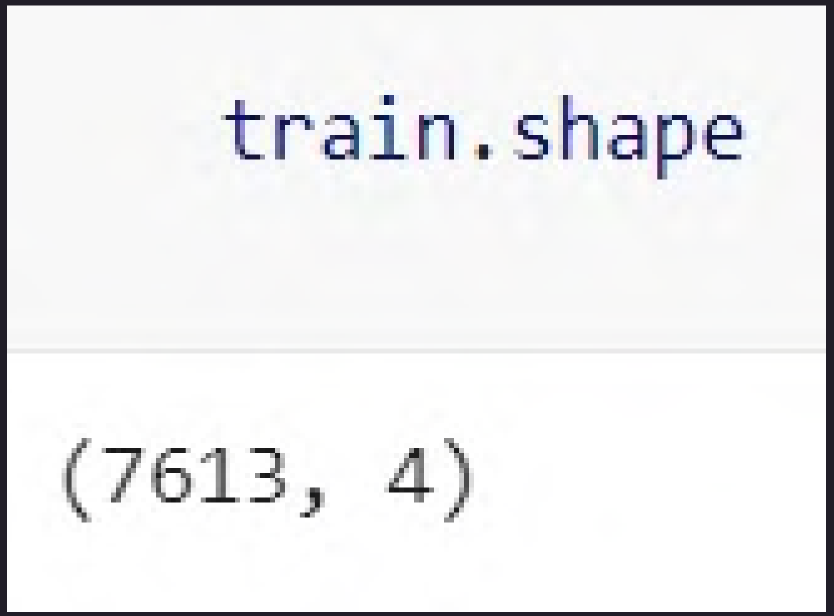
Train set



	keyword	location	text
id			
0	NaN	NaN	Just happened a terrible car crash
2	NaN	NaN	Heard about #earthquake is different cities, s...
3	NaN	NaN	there is a forest fire at spot pond, geese are...
9	NaN	NaN	Apocalypse lighting. #Spokane #wildfires
11	NaN	NaN	Typhoon Soudelor kills 28 in China and Taiwan

Test set

Drop duplicate rows in train set

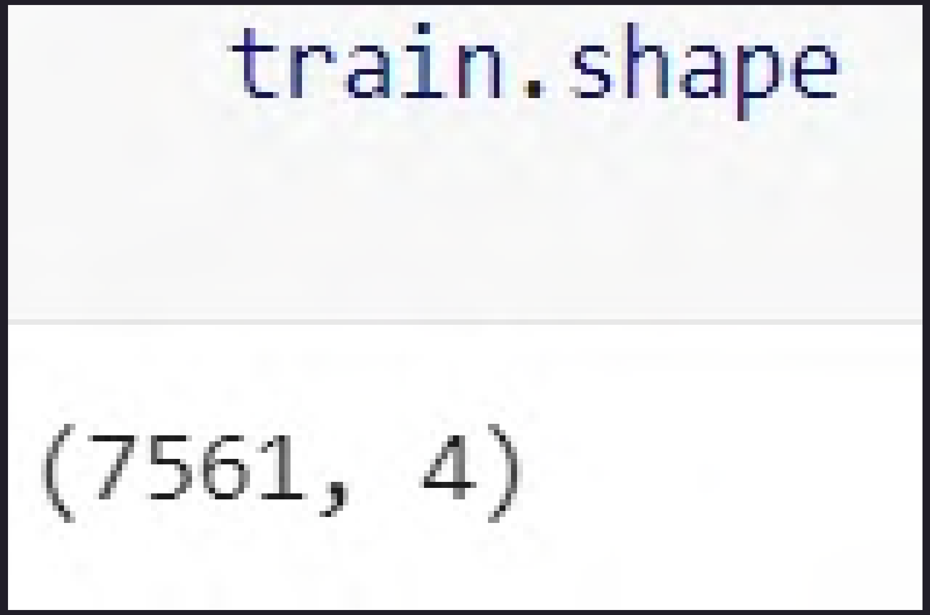


```
train.shape
```

```
(7613, 4)
```

drop_duplicates()

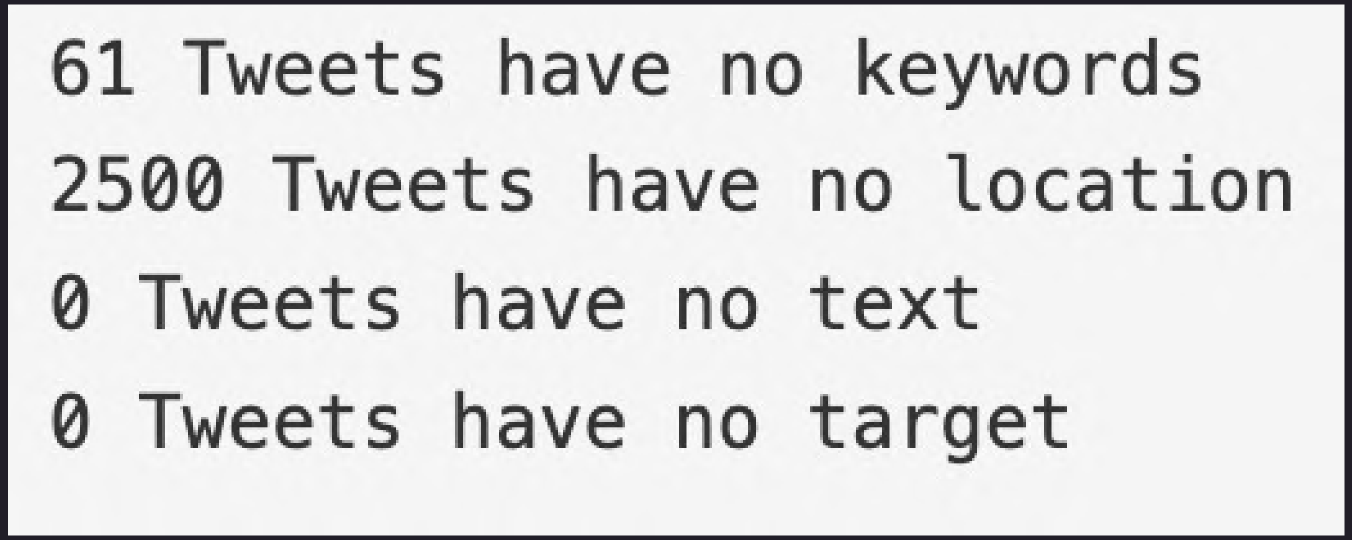
52 duplicates founded



```
train.shape
```

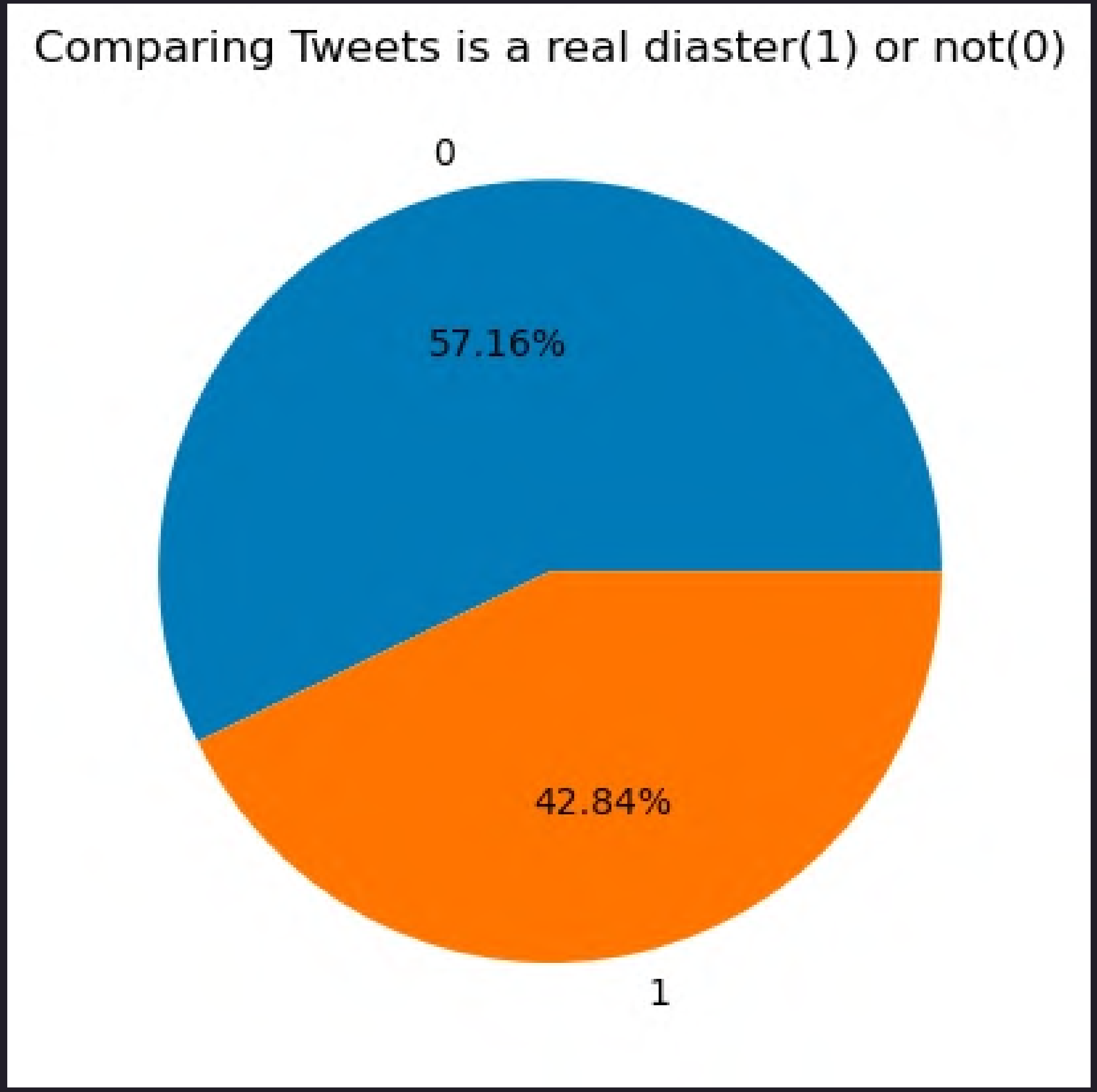
```
(7561, 4)
```

- **Check for completeness of training data**



```
61 Tweets have no keywords
2500 Tweets have no location
0 Tweets have no text
0 Tweets have no target
```

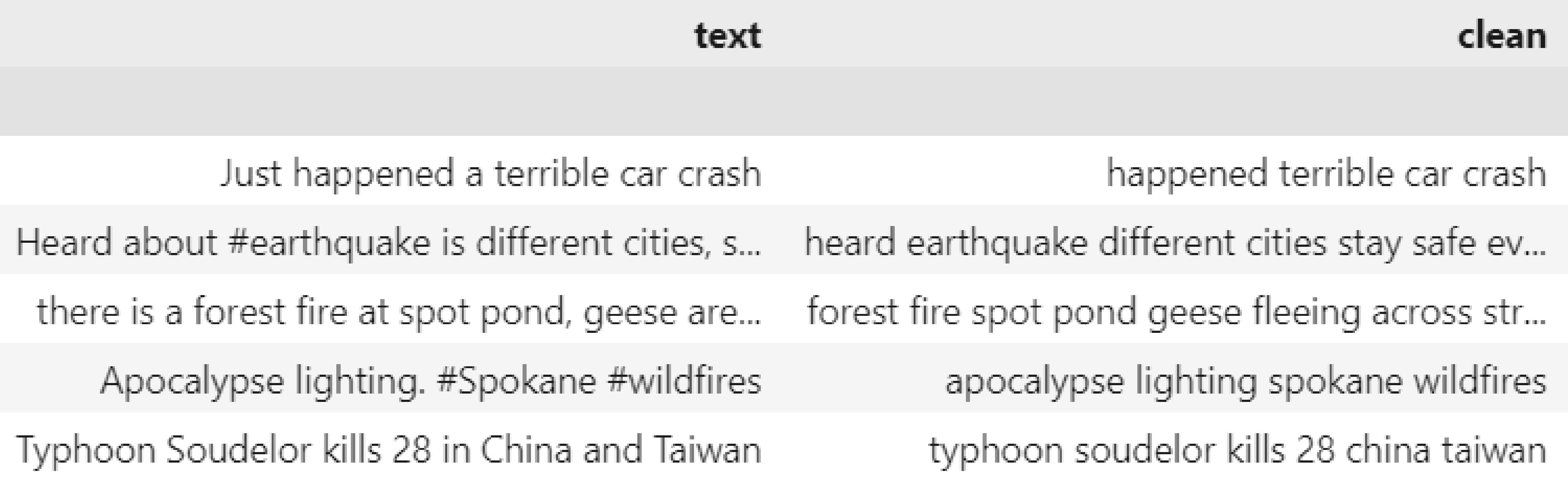
- **Check for balance of targets**



Data Preprocessing

Text Cleaning

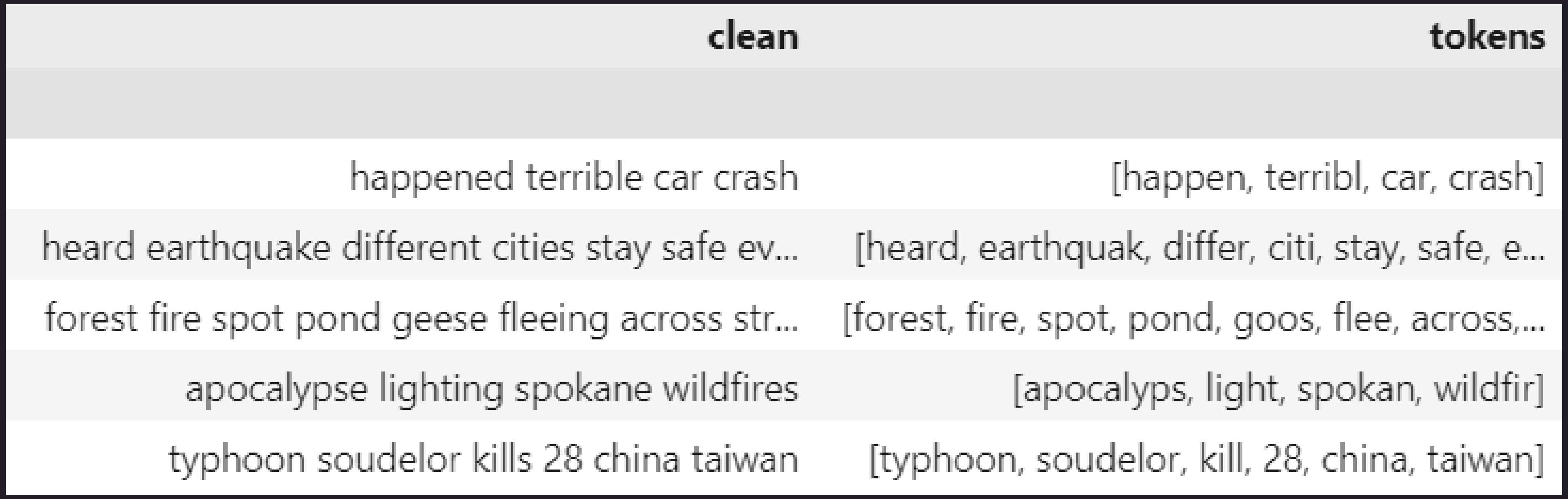
- Remove punctuation; http links; special symbols
- lower text;
- Remove Stopwords(using nltk)



text	clean
Just happened a terrible car crash	happened terrible car crash
Heard about #earthquake is different cities, s...	heard earthquake different cities stay safe ev...
there is a forest fire at spot pond, geese are...	forest fire spot pond geese fleeing across str...
Apocalypse lighting. #Spokane #wildfires	apocalypse lighting spokane wildfires
Typhoon Soudelor kills 28 in China and Taiwan	typhoon soudelor kills 28 china taiwan

Lemmatization and Stemming

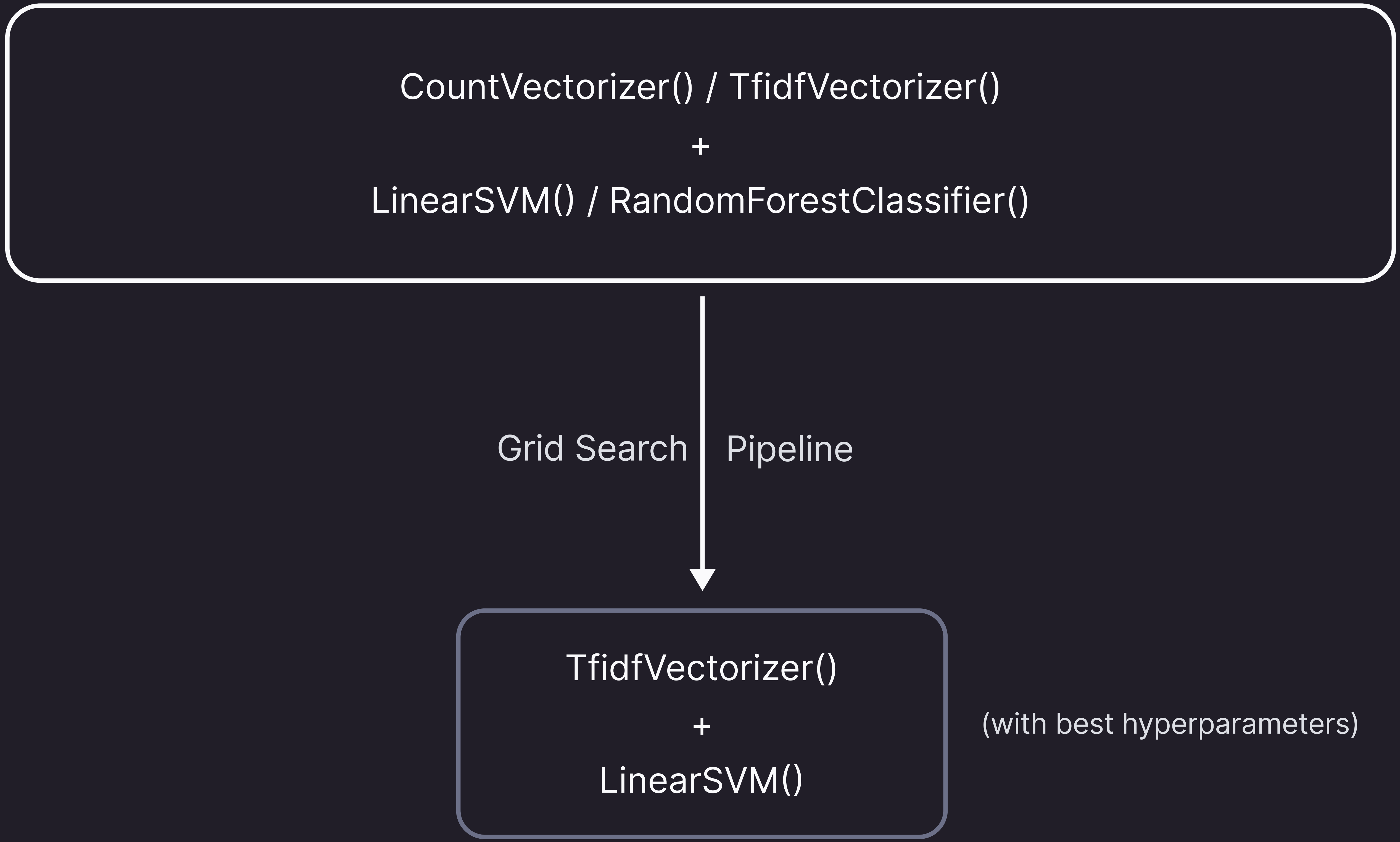
- Lemmatization: mainly from cars → cat
- Stemmer: mainly talked → talk



clean	tokens
happened terrible car crash	[happen, terribl, car, crash]
heard earthquake different cities stay safe ev...	[heard, earthquak, differ, citi, stay, safe, e...
forest fire spot pond geese fleeing across str...	[forest, fire, spot, pond, goos, flee, across,...
apocalypse lighting spokane wildfires	[apocalyps, light, spokan, wildfir]
typhoon soudelor kills 28 china taiwan	[typhoon, soudelor, kill, 28, china, taiwan]

Basic approach: Word2vec

Grid Search



CountVectorizer() / TfidfVectorizer()

+

LinearSVM() / RandomForestClassifier()

Grid Search | Pipeline

TfidfVectorizer()

+

LinearSVM()

(with best hyperparameters)

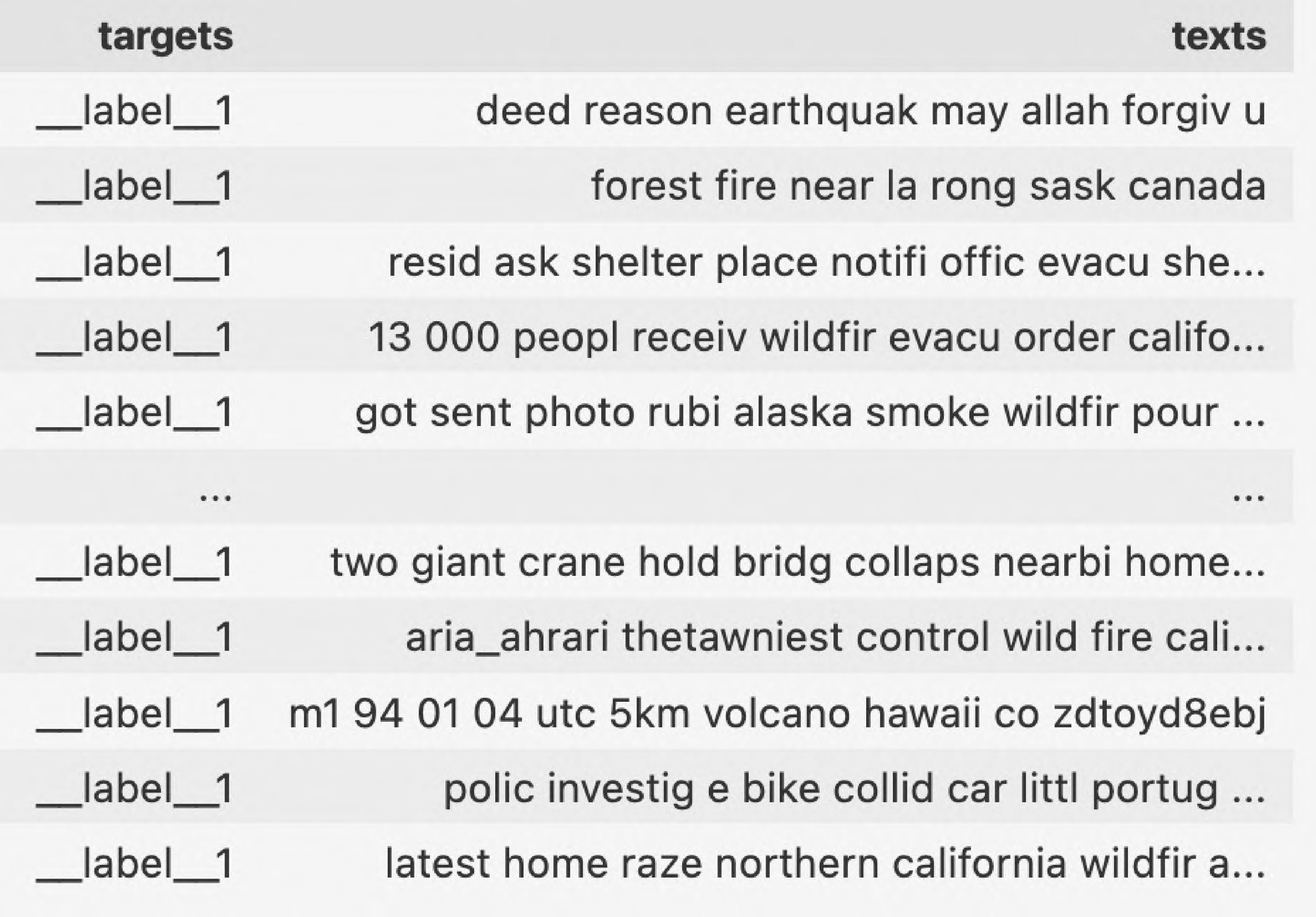
Results in Kaggle

With the best model in grid search: TfidfVectorizer()+LinearSVM(),
We do the predictions for the Tweets in test set.
And we got a score of **0.78424** on Kaggle.

Approach of best performance: Fasttext

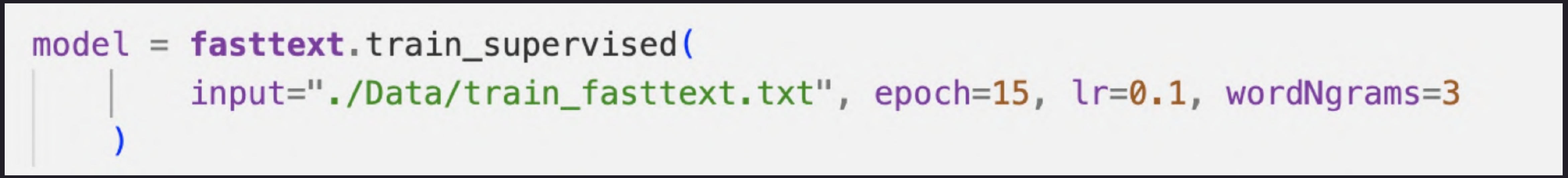
Input for Fasttext

- Texts here are preprocessed already, not the original ones
- Labels should be form of "__label__1/0" (because of binary classification)



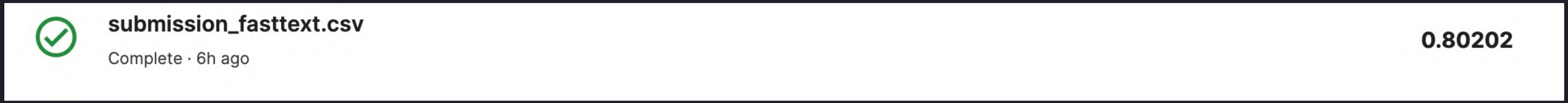
targets	texts
__label__1	deed reason earthquak may allah forgiv u
__label__1	forest fire near la rong sask canada
__label__1	resid ask shelter place notifi offic evacu she...
__label__1	13 000 peopl receiv wildfir evacu order califo...
__label__1	got sent photo rubi alaska smoke wildfir pour ...
...	...
__label__1	two giant crane hold bridg collaps nearbi home...
__label__1	aria_ahrari thetawniest control wild fire cali...
__label__1	m1 94 01 04 utc 5km volcano hawaii co zdtoyd8ebj
__label__1	polic investig e bike collid car littl portug ...
__label__1	latest home raze northern california wildfir a...

- **We do Grid Search for the best hyper parameters**



```
model = fasttext.train_supervised(
        input="./Data/train_fasttext.txt", epoch=15, lr=0.1, wordNgrams=3
    )
```

- **Accuracy of predictions for test set on Kaggle**



✓ **submission_fasttext.csv** **0.80202**

Complete · 6h ago

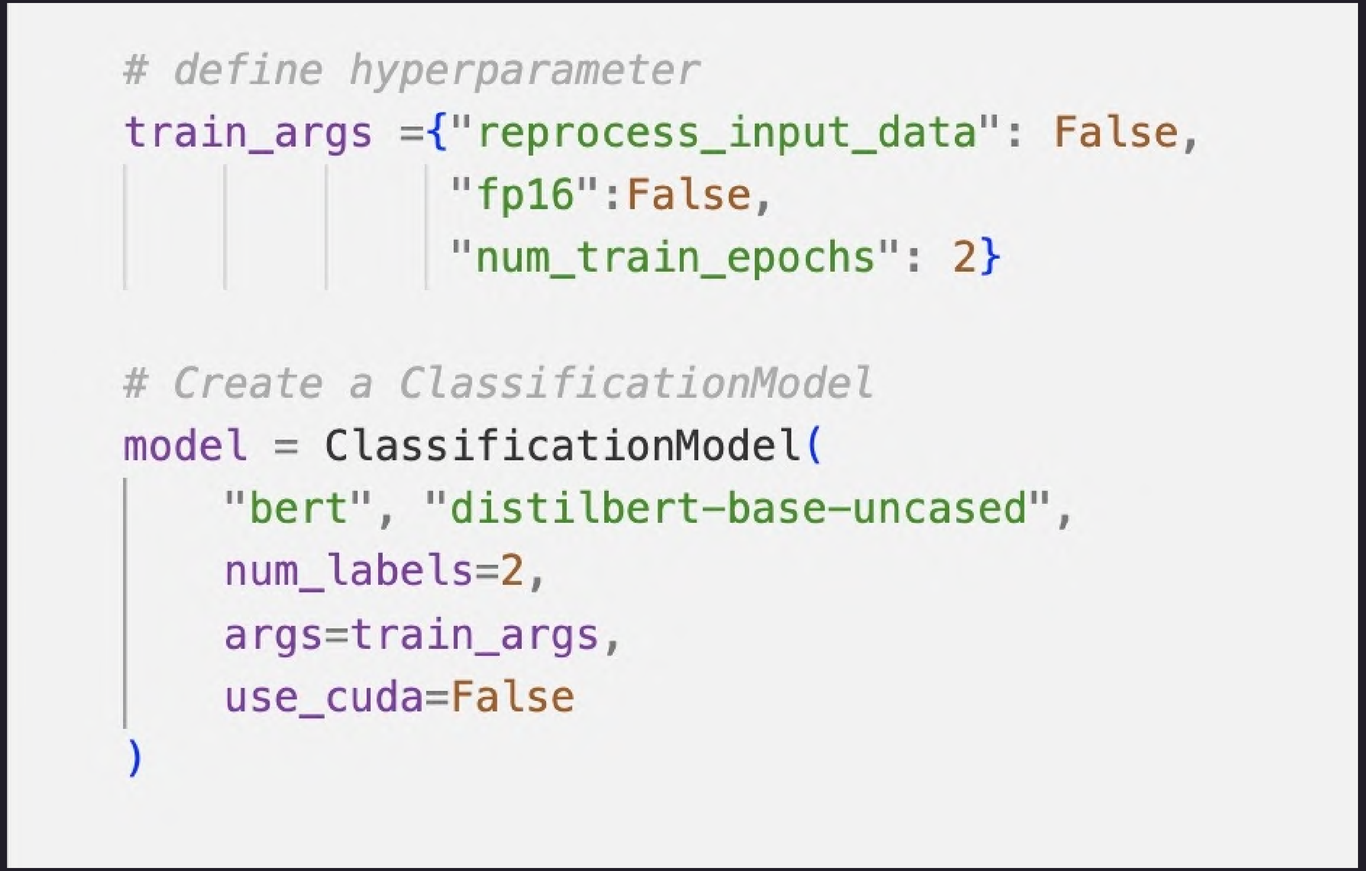
This is our best result ever!

Advanced approach: Bert

Bert model based on pretrained transformer model

- At the very beginning, We actually tried to create our own bert model with the pretrained huggingface model, which means we have to fine tune the first layer of the model. Since this is a little hard for our current level, we gave it up.

- In the final version, we use directly model "distilbert-base-uncased"(where uncase means the model is not case sensitive)

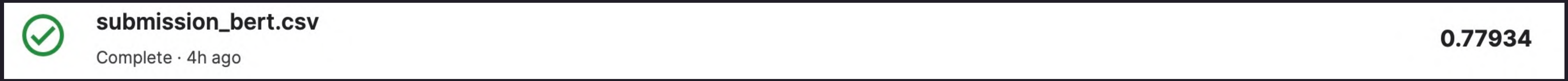


```python
# define hyperparameter
train_args ={"reprocess_input_data": False,
             "fp16":False,
             "num_train_epochs": 2}

# Create a ClassificationModel
model = ClassificationModel(
    "bert", "distilbert-base-uncased",
    num_labels=2,
    args=train_args,
    use_cuda=False
)
```

- **We haven't finish the Grid Search here**
  because training of Bert is too slow(each training takes at least 5 hours)

- **Accuracy of predictions for test set on Kaggle(using current model)**



submission_bert.csv

Complete · 4h ago

0.77934

We believe that this result will be better than the results of first two approaches
if we find the best hyperpameters of **simpletransformers.classification.ClassificationModel()**

Thank you for listening!

Questions?