Disaster Tweets

Applying Natural Language Processing (NLP) techniques

Gaurav Dembla

Nov 7, 2020

Problem Statement

Predict which Tweets are about real disasters and which ones aren't

- Users observe or experience a disaster -> post "tweets" in real-time on Twitter
- Person's words clear to a human right away, but not so much to a machine
- E.g. "On plus side LOOK AT THE SKY LAST NIGHT IT WAS ABLAZE". Here, the author explicitly uses the word "ABLAZE", but means it metaphorically.
- San Jose Police Department (SJPD) Communications Divisions responds to 911 calls
- SJPD to leverage Twitter in real-time to discover and manage incidents not-yet-reported through 911 calls, thereby lowering the services dispatch time to the events

Source Data

- **Source**: Kaggle
- **Size**: 7613 x 5
- *Target* value of 1 denotes tweet as a disaster
- **Success Criteria**: ROC-AUC (Area Under the Curve of Receiver Operating Characteristic graph) on unseen data (hold-out/test set)

id	keyword	location	text	target
1			Our Deeds are the Reason of this #earthquake May ALLAH Forgive us all	1
4			Forest fire near La Ronge Sask. Canada	1
5			All residents asked to 'shelter in place' are being notified by officers. No other evacuation or shelter in place orders are expec	1
48	ablaze	Birmingham	@bbcmtd Wholesale Markets ablaze http://t.co/lHYXEOHY6C	1
49	ablaze	Est. September 2012 - Bristol	We always try to bring the heavy. #metal #RT http://t.co/YAo1e0xngw	0
50	ablaze	AFRICA	#AFRICANBAZE: Breaking news:Nigeria flag set ablaze in Aba. http://t.co/2nndBGwyEi	1
112	accident	San Mateo County, CA	Traffic accident N CABRILLO HWY/MAGELLAN AV MIR (08/06/15 11:03:58)	1
119	accident		Can wait to see how pissed Donnie is when I tell him I was in ANOTHER accident??	0

Exploratory Data Analysis

- **Data Balance**: Nearly balanced -> No data balancing required

Target	% of records
0	57.0%
1	43.0%

- **Missing Values**
  - Location with 1/3 values -> Drop the attribute
  - Keyword is redundant information -> Drop the attribute

Attribute	% Nulls
Keyword	0.8%
Location	33.2%
Text	0.0%

- **URLs (http://)** and **mentions (@someone)** are of no use whatsoever
- **Emoticons** could be useful in predictions, as humans tend to use them in casual language to express their feelings and state of mind

id	keyword	location	text	target
1			Our Deeds are the Reason of this #earthquake May ALLAH Forgive us all	1
4			Forest fire near La Ronge Sask. Canada	1
40			Cooool :)	0
48	ablaze	Birmingham	@bbcmtd Wholesale Markets ablaze http://t.co/lHYXEOHY6C	1
49	ablaze	Est. September 2012 - Bristol	We always try to bring the heavy. #metal #RT http://t.co/YAo1e0xngw	0
50	ablaze	AFRICA	#AFRICANBAZE: Breaking news:Nigeria flag set ablaze in Aba. http://t.co/2nndBGwyEi	1
112	accident	San Mateo County, CA	Traffic accident N CABRILLO HWY/MAGELLAN AV MIR (08/06/15 11:03:58)	1
119	accident		Can wait to see how pissed Donnie is when I tell him I was in ANOTHER accident??	0

Data Cleaning

Remove the following items:

- ✓ URLs
- ✓ Mentions (@)
- ✓ Emoticons
- ✓ Punctuations (! # $? @ % & ( ) * ^_ - + / \ < = > [ ] { } | ~ . “ ` ‘ , : ;). Retain hashtag text (string following # character)
- ✓ Tabs and line breaks
- ✓ Numeric digits
- ✓ Stop words using nltk library. E.g. “the”, “is”, “in”, “for”, “where”, “when”, “to”, “at” etc.
- ✓ Non-ascii characters. E.g. convert CarolinaåÊAblaze to CarolinaAblaze.

Apply following transformations:

- ✓ Lower case the characters - “Fire” and “fire”.
- ✓ Lemmatize using spacy library

Word	Lemma
seen/saw/seeing/see	see
drove/drive/driving	drive
better/good	good
playing/played/play	play

Vectorization/Statistical Modeling

Simple

i. TF-IDF vectorization + Logistic Regression

ii. Average word embeddings + Logistic Regression/Random Forest

iii. LSTM (Long Short-Term Memory) based model

Complex

Loss function: Log-loss (aka binary cross-entropy) to optimize the model

TF-IDF

Vectorization

Clean Documents

Pre-processing

Remove –

High-freq. words (present in more than 80% of documents)

Low-freq. words (present in less than 5 documents)

- d_1: *"sky blue*
- d_2: *"sun bright today"*
- d_3: *"sun sky bright"*
- d_4: *"can see shining sun bright sun"*

$f_{t,d}$

	blue	bright	can	see	shining	sky	sun	today
1	1	0	0	0	0	1	0	0
2	0	1	0	0	0	0	1	1
3	0	1	0	0	0	1	1	0
4	0	1	1	1	1	0	2	0

$$\texttt{tf}(t,d) = \frac{f_{t,d}}{\sum_{t'} f_{t',d}}$$

	blue	bright	can	see	shining	sky	sun	today
1	1/2	0	0	0	0	1/2	0	0
2	0	1/3	0	0	0	0	1/3	1/3
3	0	1/3	0	0	0	1/3	1/3	0
4	0	1/6	1/6	1/6	1/6	0	1/3	0

N=4

$$\texttt{idf}(t,D) = \log_{10}\frac{N}{n_t}$$

	blue	bright	can	see	shining	sky	sun	today
	0.602	0.125	0.602	0.602	0.602	0.301	0.125	0.602

$\log_{10}\frac{4}{1} = 0.602$ $\log_{10}\frac{4}{3} = 0.125$

$$\texttt{tfidf}(t,d,D) = \texttt{tf}(t,d) \cdot \texttt{idf}(t,D)$$

	blue	bright	can	see	shining	sky	sun	today
1	**0.301**	0	0	0	0	0.151	0	0
2	0	0.0417	0	0	0	0	0.0417	**0.201**
3	0	0.0417	0	0	0	**0.100**	0.0417	0
4	0	0.0209	**0.100**	**0.100**	**0.100**	0	0.0417	0

Source: https://sci2lab.github.io/ml_tutorial/tfidf/

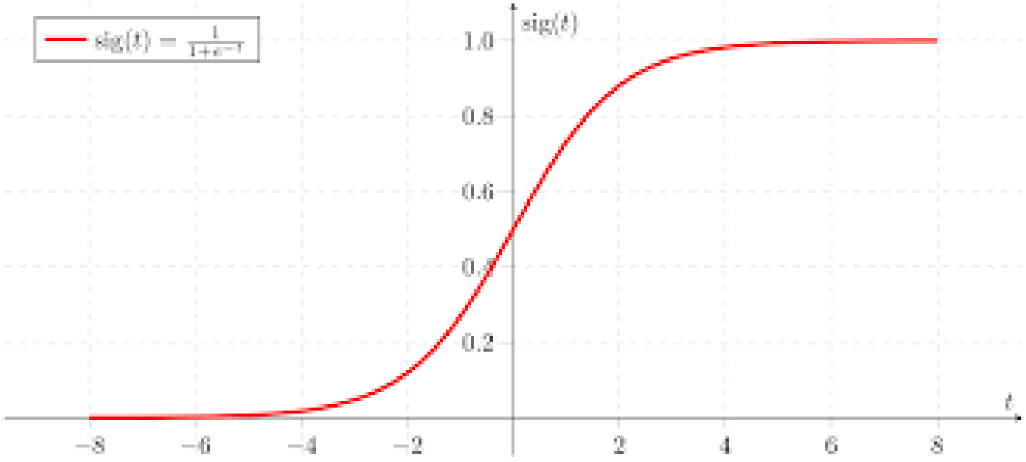
TF-IDF

Statistical Modeling

$$\mathtt{tfidf}(t, d, D) = \mathtt{tf}(t, d) \cdot \mathtt{idf}(t, D)$$

	blue	bright	can	see	shining	sky	sun	today
1	**0.301**	0	0	0	0	0.151	0	0
2	0	0.0417	0	0	0	0	0.0417	**0.201**
3	0	0.0417	0	0	0	**0.100**	0.0417	0
4	0	0.0209	**0.100**	**0.100**	**0.100**	0	0.0417	0

Sparse vector of 1783 length



Logistic Regression

Source: https://sci2lab.github.io/ml_tutorial/tfidf/

Average Word Embedding

Vectorization

Clean Documents

Spacy word2vec

Dense Vector

Average word embeddings

Illustration:

Document: “there is fire on the hill”

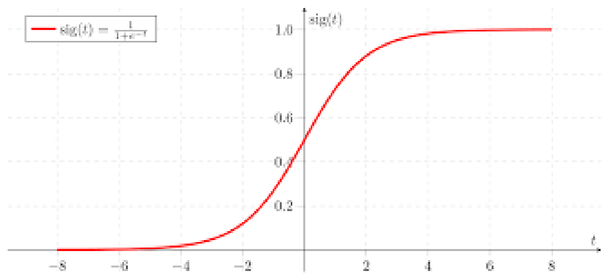
Remaining tokens after removing stop-words: [fire, hill]

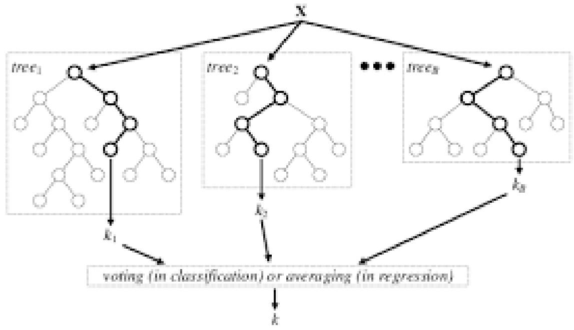
List of word embeddings (spacy): [[3, 6, 2], [9, 4, 6]]

Average embedding of the document: [(3+9)/2, (6+4)/2, (2+6)/2] = [6, 5, 4]

Statistical Modeling

Logistic Regression





Random Forest

LSTM

Vectorization

6300 features | 6300 features | 6300 features

~7600 samples

mountain wild fire escalate	[0, 1, 0, … 0, 0, 0]	[0, 0, 1, 0, … 0]	[0, … 0, 1, 0, 0, 0]	[1, 0, … 0, 0]	…	[0, … 0]	[0, … 0]	[0, … 0]	[0, … 0]	[0, … 0]
…										
see hurricane guillermo	[0, 0, 0, 1, 0, … 0]	[0, … 0, 1, 0, 0]	[0, 0, … 0, 0, 0, 1]	[0, … 0]	…	[0, … 0]	[0, … 0]	[0, … 0]	[0, … 0]	[0, … 0]

26 timesteps

Sparse vectorization

Document (clean text)	**0**	**1**	**2**	**3**	**...**	**20**	**21**	**22**	**23**	**24**	**25**
mountain wild fire escalate	406	2253	2254	1577	...	0	0	0	0	0	0

Fixed-length sequencing

Document (clean text)	**Sequence of Indexes**
mountain wild fire escalate	[406, 2253, 2254, 1577]

(Variable-length) Sequencing

[('mountain', 406), ('wild', 2253), ('fire', 2254), ('escalate', 1577), ('sky', 213), ('region', 24)]

Build vocabulary (word, index pairs)

Corpus (clean documents)

Statistical Modeling

Training/Test Data

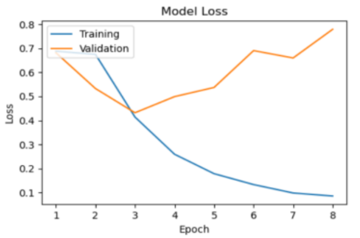
Embedding Layer	input: (?, 26, 6300)	output: (?, 26, 30)
LSTM	input: (?, 26, 30)	output: (?, 20)
Dense Layer 1 (Activation: ReLU)	input: (?, 20)	output: (?, 256)
Dense Layer 2 (Activation: Sigmoid)	input: (?, 256)	output: (?, 1)

Binary Classification Predictions

Batch Size: 64

 # Epochs: 20

(with early-stopping criteria)



Conclusion

Vectorization Model	TF-IDF	Document-level Word Embedding		LSTM
Statistical Model	Logistic Regression	Logistic Regression	Random Forest	
ROC AUC	0.847	0.853	0.858	0.851
Log-loss	0.4685	0.4676	0.4713	0.4579
Accuracy	78.7%	79.1%	81.0%	79.6%
Training Effort (in minutes)	1	1	17	1

✓ **Document-level word embedding + Random Forest**

- Simple to comprehend
- Best results of the lot
- ○ Training effort large relative to others

Future Scope

Transfer learning: Existing LSTM-based model – Embedding layer + Spacy word2vec

Useful Links

[Project Repository]

[Detailed Report]

Code: *Disaster Tweets.ipynb* (Jupyter Notebook)

Intuition behind Log-loss score

Intuition behind ROC-AUC score