Natural Language Processing with Disaster Tweets

Jiaqi Liu[1], Gang Li[2]

[1] Jilin University, China
[2] Deakin University, Australia

Introduction

Twitter has become an important communication channel in times of emergency. But, it's not always clear whether a person's words are actually announcing a disaster, which is quite clear to human, but not to computers. This may result from the following reasons:

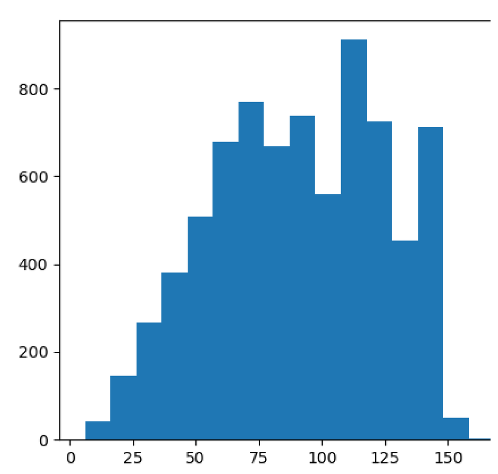
Disaster not mentioned directly When a disaster happened, people use to discuss it assuming that others are already informed. Thus, we can hardly find tweets that directly mentioned the disaster itself.

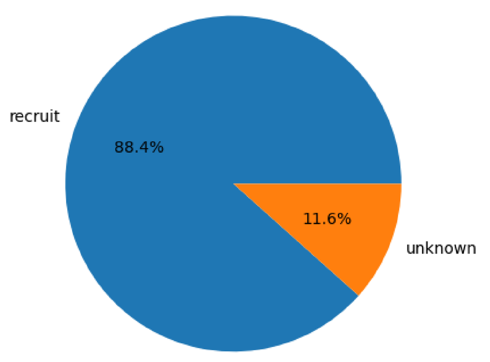
Key words understood with context As the example given by kaggle, a tweet containing word 'ablaze' may not be reporting a fire attack. This occasion is quite normal considering language habits of mankind. And identifying them is a non-trivial work.

In this work, we tend to dig the deeper features of tweet texts to make predictions beyond word itself. Thus , the above problems can be solved.

Data Process

- *Word Spliting*
  In online media, texts may contain various seperations. The vital task of spliting words by regex matching is to list all the possible seperations. And some specific forms, such as abbreviation, deformation and links need to be matched separately and primarily.
- *Sentence Padding*
  Pad all sentences to max length 209 to ensure they can be processed equally by the CNN net.
- *Word Encoding*
  Encode the words mentioned by indexing them in the dictionary. Thus, the following embedding operations can be done by finding the indexed vectors.





Word Emedding

- *GloVe*
  Word vector pretrained by *Stanford*.

Probability and Ratio	$k = solid$	$k = gas$	$k = water$	$k = fashion$
$P(k\|ice)$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(k\|steam)$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$P(k\|ice)/P(k\|steam)$	8.9	8.5×10^{-2}	1.36	0.96

- Based on the co-occurence matrix and represent the correlation of words by calculating the ratio.

Ratio	word j,k related	word j,k non-related
word i,k related	close to 1	very big
word i,k non-related	very small	close to 1

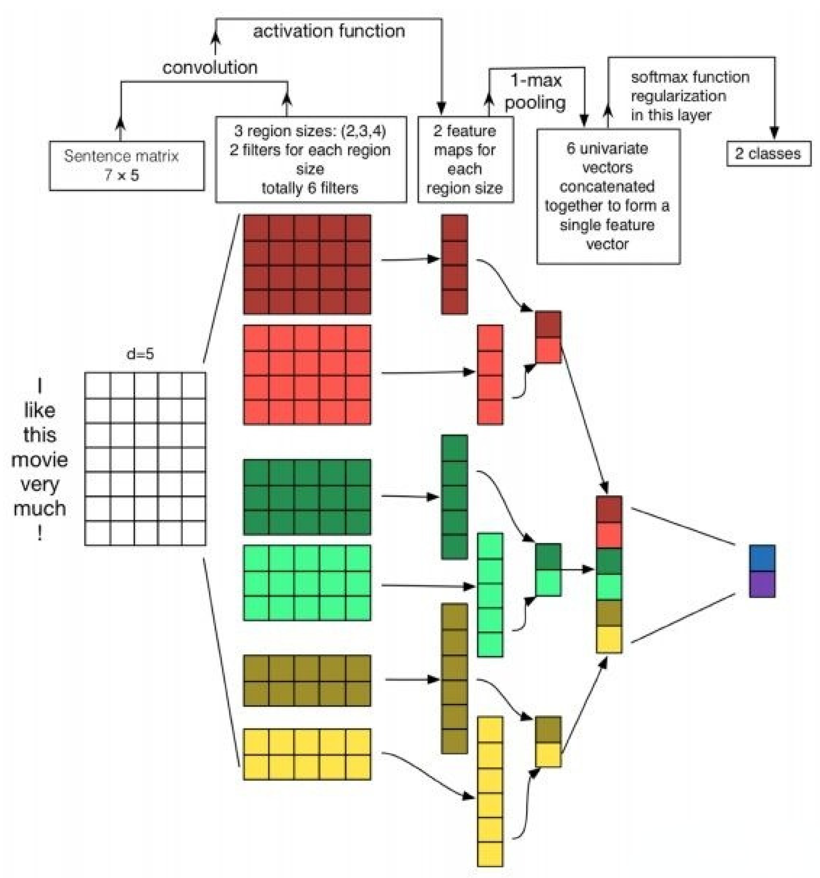
- Construct the target function which:

$$F(w_i, w_j, w_k) = \frac{P_{ik}}{P_{jk}}$$

Both sides of the equation represents correlation between words.

TextCNN

Suppose we view an 100 word vector as signals on 100 channels. We can extract text features by using 1D convolution kernal over the channels.

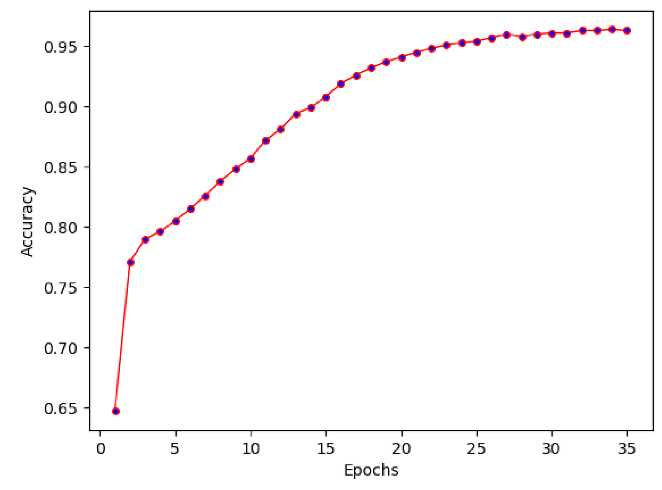


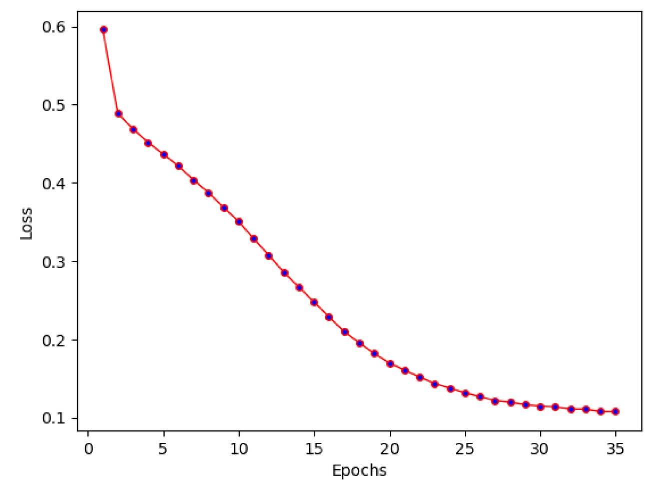
Model Training

Training Parameters are listed as followed:

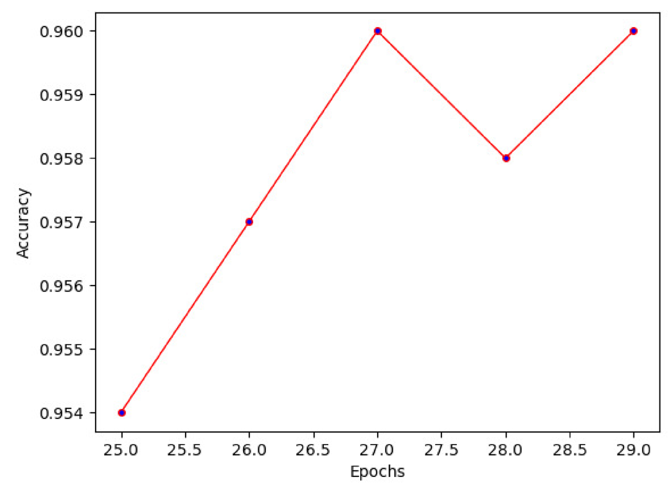
- $batch\_size = 10$
- $learning\_rate = 0.005$

Accuracy & Loss during the training process is as the figures show:





An interesting phonomenon happened near $epochs = 27$, as the figure shows.



- By testing epochs near 27, we find the best $epochs = 27$

Result

Final accuracy reahces 0.78455 with 27 epochs.

Rank :2349/3625

Acknowledgement
- Team for Universal Learning and Intelligent Processing