Machine Learning Engineer Nanodegree Capstone

Natural Language Processing with Disaster Tweets

Domain background

Natural Language Processing is a complex field which is hypothesised to be part of AI-complete set of problems, implying that the difficulty of these computational problems is equivalent to that of solving the central artificial intelligence problem making computers as intelligent as people.[1]

Every year more and more data is produced and a great proportion of it corresponds to human generated unstructured texts, so the need to advance in the field of Natural Language processing is even more evident. It is important to notice that a lot of that data is generated in real time so the efficiency of text processing is also representing an important challenge.

This project focuses on the analysis of text generated messages in social media. In this particular case, we are going to analyse tweet messages generated in the Twitter social network with the aim to determine if a tweet has some content related to a natural disaster.

Problem statement

I have chosen this Kaggle competition because I don't have much experience with NLP projects and I wanted to get started into this field. This project seems like a good opportunity to start applying text processing techniques and get fluency.

From Kaggle[2] competition page:

Twitter has become an important communication channel in times of emergency.
The ubiquitousness of smartphones enables people to announce an emergency they're observing in real-time. Because of this, more agencies are interested in programatically monitoring Twitter (i.e. disaster relief organizations and news agencies).

But, it's not always clear whether a person's words are actually announcing a disaster.

---

[1] "Natural language processing - Wikipedia." https://en.wikipedia.org/wiki/Natural_language_processing.
[2] https://www.kaggle.com/c/nlp-getting-started/overview

In this competition, you're challenged to build a machine learning model that predicts which Tweets are about real disasters and which one's aren't. You'll have access to a dataset of 10,000 tweets that were hand classified.

Datasets and input

The datasets used are the included in Kaggle competition page. Three datasets are provided:

- train.csv
- test.csv
- sample_submission.csv

For the implementation in this capstone only train and test datasets will be used. Each of the samples of train and test datasets has the following information:

- **text**: The text of a tweet
- **keyword**: A keyword from that tweet (although this may be blank)
- **location**: The location the tweet was sent from (may also be blank)

NOTE: About the datasets in Kaggle competition page it is mentioned a disclaimer about that dataset can contain text that may be considered profane, vulgar, or offensive.

Solution statement

In this case it is a binary classification problem in which tweet messages should be classified as disaster(1) or non disaster(0). For the solution I have planned to implement a model using a LSTM neural network to perform the classification.

The whole solution is going to include the following phases:

- Exploratory data analysis to explore data in depth.
- Preprocessing and cleaning data.
- Feature generation.
- Implementation of classification model.

Benchmark model

The model implemented will be trained using the train dataset and will be benchmarked against a test dataset evaluating the result using the metrics described in next section.

Evaluation metric

In order to evaluate the performance of the model the metrics used are: accuracy, precision, recall and f1-score. These metrics will give us a good intuition about how well the model is classifying or misclassifying between disaster or non disaster. Additionally, I am going to use RMSprop as the optimizer inside the LSTM network.

Project design

Next, the steps to solve the project will be detailed:

Getting data

The data was downloaded from Kaggle competition and included into the project repository in the “input/tweets_data” folder.

Exploring data

In order to get data insights and get patterns in data, I will do an exhaustive exploration of the data downloaded in the previous section.

Preprocessing data

Before training the LSTM network it is necessary to preprocess data in order to prepare the model input. Some actions that will be performed in this section will be removing stop words or other invalid characters, tokenize, etc...

Training model

I am planning to build a LSTM neural network. The optimizer used will be RMSProp and Binary Crossentropy as loss function.

Evaluate model metrics

Evaluate model performance using test dataset to do a complete reporting using metrics mentioned in Evaluation Metric section.