**Disaster Tweets : Real or Not ?**

**MODEL DESIGN DOCUMENT**

**Contents**

1. **Problem Definition**
2. **Software Requirements**
3. **Data Collection**
4. **Data Preprocessing**
5. **Data Analysis**
6. **Model Selection**
7. **Training**
8. **Evaluation**
9. **Fine Tuning Again**
10. **Submission**

**1. PROBLEM DEFINITION:**

[**Link to the Problem**](https://www.kaggle.com/c/nlp-getting-started)

**1.1. Problem Statement :**

**Predict which Tweets are about Real Disaster and which ones are not**

**1.2. Problem Analysis :**

**This is a Binary Classification problem with text data as the input, hence we’ll be using NLP.**

**2. SOFTWARE REQUIREMENTS:**

**[This is a good list for now. It may change a bit as we work through the project, though. And that’s okay.]**

**Programming language :**

* **Python 3**

**Data Manipulation :**

* **Pandas**
* **Numpy**
* **Spacy**

**Data Visualisation :**

* **Seaborn**
* **Plotly**

**Machine Learning:**

* **Tensorflow**
* **Tensorflow\_hub**
* **Keras**

**Evaluation:**

* **Scikit-Learn**

**3. DATA COLLECTION :**

[**Link to the the Dataset**](https://www.kaggle.com/c/nlp-getting-started/data)

**3 Files :**

* **train.csv**
* **test.csv**
* **sample\_submission.csv**

**4. DATA PREPROCESSING:**

**Train.csv Analysis from Kaggle**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **COLUMN** | **DATA TYPE** | **MISSING**  **( in %)** | **UNIQUE COUNT** | **FREQUENT** |
| **id** | **Key / int** | **0** | **all** | **all** |
| **keyword** | **String** | **1** | **221** | **fatalities** |
| **location** | **String** | **33** | **3341** | **USA** |
| **text** | **String** | **0** | **7503** | **11 year old** |
| **target** | **Integer** | **0** | **2** | **1** |

***Doubts*:**

1. **Can we use col with missing values(keyword, location) ? Is it necessary ? [Yes, we can and we should. We may be able to get some additional predictive power from these columns. The next question is how to use them. I’ll let you think about that rather than giving you possible answers.][Imputation and flags to specify that the value is missing]**
2. **Will using / not using cols with missing values affect the prediction ? [Yes, and if we do it right, this may improve the model.]**
3. **I am planning to use spacy ( a new library which I will be using or the first time). Is spacy or a good choice or any other effective solution to tokenize & text preprocessing ? [Since it’s so new, I don’t know much about it yet. It’s certainly worth exploring. Does it work well with BERT? Maybe do a few Google searches on that.] [Okay]**

**5. DATA ANALYSIS :**

***Doubts*:**

1. **Is it necessary to analyse data, as Kaggle has already given a detailed analysis [Only when we have a specific hypothesis in mind. Do you have any specific things you want to check? I think you should look at what Kaggle has done and then think of specific things you might want to analyze based on specific hypotheses. But if you don’t have any at this time, don’t worry. We can skip this step for now and then come back to it later with hypotheses. Specifically, we can do a first round of model building and then come back to this to study tweets that our initial model misclassifies.]**
2. **Does text data require a Graphical Analysis ? Will Visualization be of any use since the model will automatically learn from the text ? [Same answer as above. Only if we have specific hypotheses we want to check. It’s okay to come back to this in the second round of analyses.]**

**6. MODEL SELECTION :**

**BERT = Bidirectional Encoder Representation from Transformers**

***Thoughts* :**

1. **I have chosen BERT as it has been trained using Large Dataset. [Good choice.]**
2. **This is my first time learning & using BERT. Any tips for effective usage of BERT ? [I’m going to let you study BERT and figure it out for yourself.] [Sure]**

***Doubts*:**

1. **Using BERT is a kind of transfer learning, isnt it ? [Yes]**
2. **Is BERT a good choice or any other model suggestions ? [Very good choice]**
3. **There are so many BERT models** [**here**](https://tfhub.dev/s?q=bert)**. How to decide which one to use ? [Do internet research to figure this out.] [Sure]**

**7. TRAINING :**

**Planning to split into => Train size:0.8 , Test size:0.2**

**Will this be okay ? Your suggestions please… [I generally make my test sets about 30% of the data.]**

**8. EVALUATION :**

**Kaggle mentioned that they will use F1 Score to evaluate the model**

**I am planning to use SCIKIT-LEARN CLASSIFICATION REPORT to evaluate. Is it okay ? [Yes. I believe it returns the F1 score, so that should be good. Obviously, you should focus on the F1 score.]**

***Doubts* :**

1. **My guess is that a high Recall Score is necessary as identifying True Disaster is necessary take emergency actions to help people in disaster. Is my guess correct ? [Well, it depends on the situation. But it doesn’t matter here. See my next point below.]**
2. **Why Kaggle uses F1 score instead of Recall / Sensitivity to evaluate models ? [F1 is a balanced metric that evaluates how good the model is in pushing the “efficient frontier” of precision vs. recall. You can’t focus on just recall or just precision. You need the model to focus on both. Focusing on F1 is one way to focus on both in a balanced way. (As a metaphor, do you want a project to be fast or accurate? Usually you can go very fast and be less accurate or be very accurate and be slower. But you normally want both, with a reasonable tradeoff between them. This is just a metaphor, but do you get the point?)] [Yes]**

**9. FINE TUNING AGAIN :**

**After evaluation, I think we should fine tune certain things, like parameters (if any),**

**Improve text preprocessing even more if necessary [Definitely. And focus in on those tweets that are misclassified by the model to figure out why they are misclassified.]**

**10. SUBMISSION :**

**Your suggestions on the final checklist before submission. [Full review of the code and all intermediate data, multiple performance metrics, etc.]**

**[Thank You]**

**Predict which Tweets are about real disasters and which ones are not**