Text Mining Project Handout 2025

Stock Sentiment

Predicting market behavior from tweets

1. Project Objective

"Over time, major indexes go up and down based on internal and external factors. Performance like that excites investors, but typically in opposite ways. Constant gains lead some investors to expect more of the same. Others worry the good times are surely about to end. The former sentiment is sometimes called "bullish," while the latter is referred to as "bearish."[1]

The goal of this project is to develop an NLP model capable of predicting Market sentiment based on tweets. In summary, with the NLP techniques you have learned during class, you must implement a classification model that receives tweets as inputs and is able to predict, for each tweet, if it describes a Bearish (0), Bullish (1), or Neutral (2) attitude.

The project should be developed using python 3 and libraries such as NLTK and Scikit Learn and Huggingface. Also, the project is simple and can be solved in various ways, which means there is no exact correct solution. Students should not use code from each other!

2. Group Rules

The project is to be developed in groups of **two to five (2-5)** students. The group will receive a **2.5-point penalty** for each person below or above the expected group size.

3. Corpora

The data is divided in a file for training "train.csv", and another file for testing "test.csv":

- **Train** (9543 lines): Presents the tweets ("text") and the sentiment label ("label"). Each tweet can have one of the following labels: Bearish (0), Bullish (1), or Neutral (2). You can divide this set in Train/Validation.
- **Test** (299 lines): The structure of these dataset is the same as the train set, except that it does not contain the "label" column. You are expected to provide the predicted status (0, 1 or 2) for each tweet in this set and **we will compare your predictions with the actual (true) labels**.

[1] https://smartasset.com/financial-advisor/bullish-vs-bearish

4. Solution Requirements and Evaluation Criteria:

Your solution should present the following points:

1. **Data Exploration**: Here you should analyze the corpora and provide some conclusions and visual information (bar charts, word clouds, etc.) that contextualize the data.
2. **Corpus split**: You must apply some method to split your training corpus into train/validation sets to evaluate the performance of your model. You can also resort to K-Fold cross validation.
3. **Data Preprocessing**: You must correctly implement at least four (4) of the data preprocessing techniques shown in class (stop words, regular expressions, lemmatization, stemming, etc.).
4. **Feature Engineering**: You must correctly implement and experiment at least one variation of the following feature engineering techniques seen in class: BoW, word2vec, Transformer (encoder).
5. **Classification Models**: You must correctly implement and test at least one variation of the following classification methods seen in class including KNN, LSTM, Transformer (encoder).
6. **Evaluation**: You must correctly evaluate and compare your models resorting, at least, to Recall, Precision, Accuracy and F1-Score, and explain what the evaluation means in the context of the problem.

Moreover, the development of extra work (techniques not shown in class that are more advanced than the ones in point 5 above) is highly recommended and will account for a maximum of **3 points** divided as follows:

1. **Feature Engineering – 0.5 point** for using other advanced embedding methods (maximum of 2 extra methods). For example, resorting to another transformer embedding method, other than the mandatory one.
2. **Classification Models – 1 point** for using other transformer encoder and decoder models for classification (1 extra method of each).

5. Delivery Guide

In terms of the solutions developed (see **delivery template folder**), you must deliver:

1. One .pynb file (notebook), named tm_tests_xx (xx stands for the group number), following the structure in section 4 of this handout and containing the techniques you experimented and their evaluation.
2. Another .pynb file (notebook) named tm_final_xx with only your ready-to-run final solution. This solution should include a single pipeline with a single classification model.
3. A .csv file, named “pred_xx”, with only two columns - the id of the test set and your predicted labels for the test set.

Additionally, you **must submit a PDF report** named “report_XX”, documenting your work, with the following structure (other structures are also accepted):

1. **Data Exploration** – data presentation and explanation of the main finding from the exploratory analysis (accounts for **50%** of criteria **4.1**).

2. **Data Preprocessing** – explanation of the different preprocessing methods developed (accounts for **25%** of criteria **4.2** and **4.3**).
3. **Feature Engineering** – description and explanation of the feature engineering methods applied (accounts for **30%** of criteria **4.4**)
4. **Classification Models** – description and explanation of the models implemented (accounts for **30%** of criteria **4.5**)
5. **Evaluation and Results** – description of the performance of the models and main conclusions (accounts for **50%** of criteria **4.6**)

Final Notice:

- The PDF report should have a maximum of 10 pages describing the previous points. Exceeding this number will incur a **0.5-point penalty** for each extra page.
- Any **extra work** developed ***must be clearly defined as such in the PDF report***, or else it will not be considered for evaluation as extra work!
- All files should be saved in a folder named "group_xx". This folder (zip it if you need) must be submitted through Moodle's project submission section until **23h:59 of the 15th of June (Sunday)**.
- Failure to deliver on time will incur a **1.0-point penalty** for each half-day late.
- Failure to comply with the delivery guide above will meet with up to **1.0-point penalty**.

Extra Challenge

We will compare your predictions with the actual Label from the test set ("test.csv").
The three (3) groups with the highest performance will receive points as follows:

- **2.00 points** for the group with the best model
- **1.00 points** for the group with the 2nd best model
- **0.50 points** for the group with the 3rd best model

Students may be **randomly selected for an oral defense** to access their knowledge.

Good luck with your project!