Text Mining - Final assignment
Bias influence on sentiment analysis

Maud Megret, Dekel Zeldov, group 11

5 January 2024

Contents

1 Introduction

For this project, we wanted to focus on the sentiment analysis field, which constitutes an important part of all of the various text mining uses. Thanks to social media like Tweeter, where people can express themselves and give their opinion about everything, the number of data sets containing opinion texts available to study is really important. One of the possible uses can be to analyse some of those tweets and to aggregate either positive, negative, or neutral opinions on them thanks to some models pre-trained on existing labelled data.

The data collected and labeled for training are not always well balanced. Some properties of the data set are known to have an effect on the resulting module. [2] researched the effect of such an imbalance on the bias of the trained model. They found out... . If such an effect is prominent, it is important to know its extent in order to take proper precautions, such as balancing the dataset before using it for training.

In this project, we want to conduct an additional investigation on the impact of an unbalanced dataset on the bias of the resulting trained module. We will train the same model a few times, each time with a dataset that has a different balance between the amount of positive and negative labels. The data we will use is from the dataset *SemEval-2017 Twitter* . We will enforce the imbalance between positive tweets and negative tweets in our dataset by under-sampling it.

2 Background

We wanted to check the influence on our model predictions when we train on more positive or negative than the other labels. It is quite clear that if we train our model with only positive tweets, it will predict only positive opinions, but we do not really know how the influence on predictions would evolve with the proportion of positive and negative tweets in the training set.

To make our data set more or less unbalanced, we had the choice between two methods as presented in [2].

Method 1 : Over-sampling

It consists of getting more data by increasing the number of minority class members in the training set. To do that, it is possible to duplicate some of our existing data.

Method 2 : Under-sampling

This method consists in removing some of the data. The goal is to reduce the number of majority class enteries in the training set.

Because our data set is large enough, we decided to use the second method. By removing some of our data, we will still have a very large data set. That way we can also avoid the consequences of duplicated data that would result from over-sampling.

3 Methods

3.1 Model

In this work, we use Huggingface's distilbert-base-uncased model. It is a smaller and faster version of the BERT base model, pre-train on the Masked Language Modeling task. Since the goal of our work is to study the impact of training the model for sentiment analysis with an unbalanced dataset, it was important that we use a model that was not yet trained in sentiment analysis.

3.2 Data set

We used the*SemEval-2017 Twitter* dataset [3]. The dataset consists of tweets, each with a label of *positive*, *negative*, or *neutral*. For training, we used the accumulated data from all previous years, containing ** tweets, of which ** are labeled *positive*, ** are labeled *negative*, and ** are labeled *neutral*. For evaluation, we used 2017 test data that contain ** tweets, of which ** are labeled *positive*, ** are labeled *negative*, and ** are labeled *neutral*.

We deleted all the duplicates and we are wondering if we should also remove the neutral tweets. Our data set will still be large enough and on all the articles that we read

3.3 Training the Modules

To decide which parameters to use for the training, we experimented with a small sample of training data. We tried a few different values for batch size and learning rate. On the basis of that, we chose to set the values to **.

For the baseline, we trained our model with the full training dataset.

3.4 Training with an unbalanced dataset

Questions for the draft report:

- Should we discuss the Over-sampling vs. under-sample in the Methods or in the Intro?
- We plan to go with under-sampeling the data, in order to avoid duplicates. we feel it is a better representation of the real cases where the trainig data is unbalnced. Is that a good choice?
- We were thinking to train only with positive and negative labels for the unbalnced training, in order to focus on the ration between those two and not mix in a third label. Do you think it is a good idea?
- Should the ration in the test data set be 1:1?

In order to study the effects of an unbalanced dataset, we did the same experiment 5 times with a different ration between the labels in the training data each time.

The rations between the number of positive and negative tweets we expiramented on where: $p \in 1:9, 3:7, 5:5, 7:3, 9;2$. The dataset for each retio was constracted by randomly sampeling the needed amount of tweets with each label. Then, for each data set we traind a new mudole.

All modules were evaluated on the same testing dataset, so that we would be able to compare the final results of each of the models. The testing dataset was (is the original 2017 test set) / (that was constructed by random sampling such that it will have the same amount of positive and negative tweets).

4 Results

The results we obtained for the baseline and each of the five different values of p are described in *table ref*. For each module, we recorded the recall and precision and F1 scores for each label, and the $g_{performance}$, as suggested by [2]. In figure *fig ref*, a plot is shown for each of the scores, over all of the models.

** table for the baseline with the score for each of the 3 labels.

** A plot with a line for each score, yAxes being the score value, xAxis being the diffrent modules by order of ration between the labels**

	F1		Recall		Precision		Accuracy		$g_{performance}$
	P	N	P	N	P	N	P	N	
0.1 - 0.9									
0.3 - 0.7									
0.5 - 0.5									
0.7 - 0.3									
0.9 - 0.1									

Table 1: Performance Metrics at Different Thresholds

5 Conclusion

Comparing our baseline with [1], we can assert that our modules are valid.

What is the influence of bias on our results ?

Is it a good idea to make the data sets well balanced before training any model to do sentiment analysis ?

References

[1] Mathieu Cliche. Bb_twtr at semeval-2017 task 4: Twitter sentiment analysis with cnns and lstms. *CoRR*, abs/1704.06125, 2017.

[2] Asmaa Mountassir, Houda Benbrahim, and Ilham Berrada. An empirical study to address the problem of unbalanced data sets in sentiment classification. In *2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 3298–3303, 2012.

[3] Sara Rosenthal, Noura Farra, and Preslav Nakov. SemEval-2017 task 4: Sentiment analysis in Twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation*, SemEval '17, Vancouver, Canada, August 2017. Association for Computational Linguistics.