Public Opinion On News
Text Mining and Sentiment Analysis Project

Nicole Maria Formenti 941481 - DSE

June 2021

Public Opinion Analysis on the New York Times Comments Dataset

1 Introduction

This project focuses on analysing the *New York Times Comments dataset*, which contains several information about articles and their relative comments, along with the text of the comments itself. The general task is that of analysing the controversy of comments in order to understand how controverse is the article itself and its topic, contained in the variables `newDesk` and `sectionName` . The controversy is measured in terms of three variables, namely `editorSelection`, which indicates whether a comment has been considered worth promoting by a New York Times editor, `recommendations`, indicating the number of upvotes received by a comment from the other readers, and `replyCount`, which is the number of replies received by a comment.
Another aspect of interest for studying the topics of articles is the polarity of comments, which is the general mood or sentiment expressed. Therefore, methods from text mining are applied to process the textual data along with sentiment analysis and machine learning techniques to extract useful information and making predictions.

2 Research question and methodology

The goal of this analysis is threefold. First of all, a suitable supervised model for the classification of the most controversial comments is built. In this way it will be possible to predict if a comment will be controversial based on its characteristics. The second aim is to study the relationship between the controversy of comments and the topics of the articles to which the comments refer. Lastly, an alternative unsupervised lexicon-based approach is used to label the comments based on the polarity and intensity of the sentiment they express. The resulting labels will be explored in order to understand whether the sentiment of comments is a good predictor of their controversy, as done in [.] In addition, the polarity of comments will also be related to the topics in order to understand what is the general mood of the comments referring to a specific topic.

The model chosen for performing classification is the neural network, which works particularly well when a large dataset is available, such as in the case of the *New York Times Comments dataset*, which contains around 3 millions observations. The problem can be considered as a multi-label classification, where the same observation can have one or more positive labels, hence the chosen architecture is a multi-task neural network. Its main advantages are that of biasing the learning towards the most informative class or classes and that of reducing the noise from data by ignoring irrelevant information, hence generalising better. In order to make predictions by exploiting all the different data sources, a final multimodal neural network is used. This network concatenates together three pre-trained neural networks, namely a feed forward neural network for generic features, a convolutional neural network for the keywords of the articles and another convolutional neural network preceded by an embedding *doc2vec* layer for the articles' text. Learning from different sources allows the model to exploit possible synergies among different data about the same phenomenon.

The articles' text has been embedded through the *Doc2Vec* algorithm [4], which is based on neural network language models. This embedding method is based on another algorithm called *Word2Vec*, which is used to produce a representation of each word based on its context. The representation is provided by the weights of the first hidden layer of a neural network which has been trained to return a word from its context (*CBOW* model) or a context from a specific word *SkipGram*. This representation is a mapping of the original vectors onto a space where semantically similar words share a similar vector representation. Hence, they are close to each other.

The *Doc2Vec* model works by using as input for a neural network the vector of words plus an additional vector which uniquely represents the document and which will be trained along with the vectors of words. The task of the neural network either becomes that of predicting a center word by using the vectors of the context words and the unique document vector in the *PV-DM* (Distributed Memory version of Paragraph Vectors) or that of predicting the context using only the document vector in the *PV-DBOW* (Distributed Bag of Words version of Paragraph Vector). The vector representation of interest that is used to embed the documents is the trained document vector. In order to obtain the vector for a new unseen document, the neural network is trained again by adding a new document vector while keeping all the other vectors and weights fixed. After the training, the document vector is retrieved. The *Gensim* implementation allows for the use of a combination between the *PV-DM* and *PV-DBOW* models to learn the document vector, which, according to the authors of the paper, obtains a better performance. For this analysis the combined model is used.

The two main advantages of the *Doc2Vec* model are that of maintaining the semantic similarity among words through vector representation and that of taking into account the ordering of words in a small context, which is given by the size of the window used.

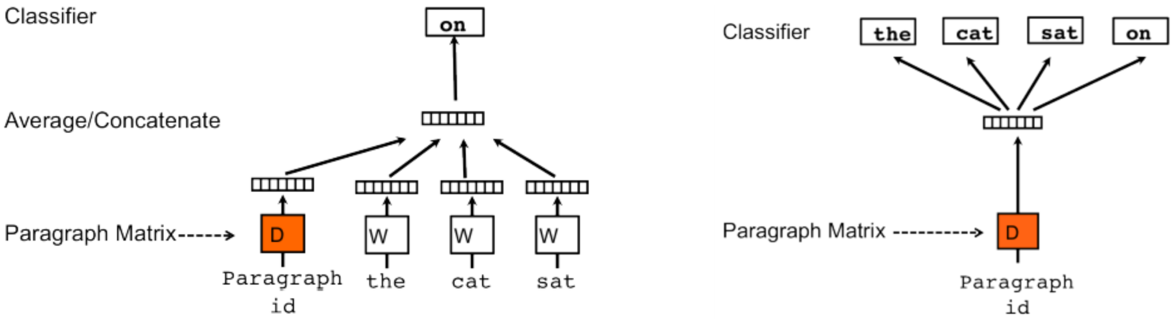


Figure 1: *PV-DM and PV-DBOW Doc2Vec model from Q. Lee and T. Mikolov paper* [4]

On the other hand, articles' keywords have been vectorised by weighting words through the *TF-IDF* score, which gives a larger weight to words that are both frequent in a specific document and infrequent in all the others, hence the more document-specific words.

The neural networks are multi-task networks with three different outputs, which are `editorSelection`, `recommendation` and `replyCount`. The output variables have been binarized. The target `Recommendations` has been divided in two classes in correspondence to the median value, which takes value 4, while the target `replyCount` can take values < 1 or ≥ 1. The last response variable `editorSelection` was already a boolean variable.
The first target is completely balanced due to its design, while the second one is somewhat unbalanced with a proportion of about 1 to 4 between positive and negative samples. The biggest problem is posed by the last variable which is heavily unbalanced, with a ratio of 1 to 49. For this reason, the metric used to assess the model is the *F1 score*, which balances the recall and precision metrics and it is particularly suitable for unbalanced target variables.

The training of the neural networks consists of two phases:

1. Hyperparameters tuning: a random search through a TPE (Tree-structured Parzen Estimator) sampler is performed to find the optimal optimiser between Adam and RMSprop, and the optimal learning rate. The tuning is carried out by leaving out a portion of the dataset that is used for testing in the second phase. The batch size is 100 for training samples and 200 for testing samples. Only a single architecture for each network and 5 trials of 2 epochs each are used in order to reduce the computational time required.

2. Model testing: the final performance of the optimised model is assessed on the held-out test set. A single architecture is used here too and the number of epochs is 20 with early stopping after 3 epochs of not improving test error.

Each single neural network is composed of one or more first common hidden layers followed by a single hidden layer for each task and a softmax layer for classification. The loss used for optimisation is the cross-entropy. The analysis has been carried out by first optimising and testing all the single neural networks. Afterwards, the weights of the common hidden layers have been frozen and loaded into the multimodal architecture, which concatenates the flattened representation of the three previous neural networks and then uses a further common hidden layer followed by the three specialised hidden layers for each target. The multimodal network is then optimised and tested by training only the last layers, so leaving out the first pre-trained layers. The procedure is illustrated below:

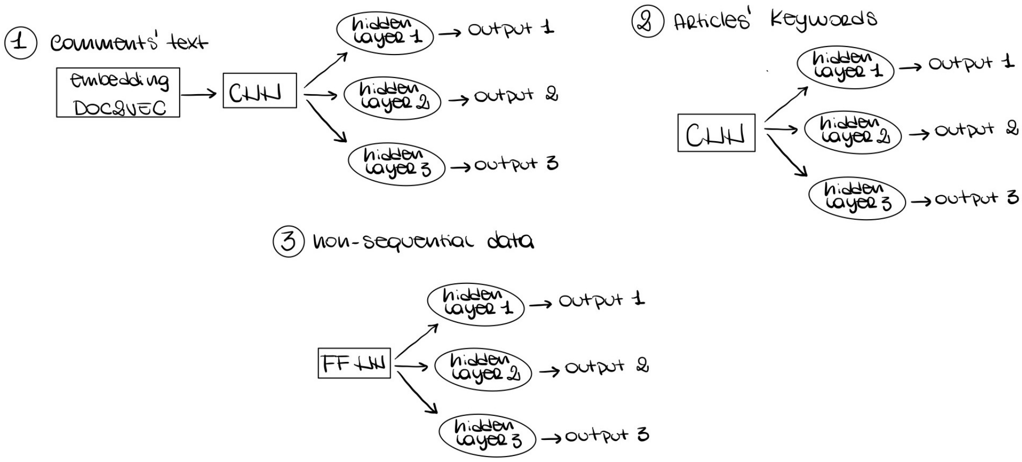


Figure 2: *step 1: train single multitask networks*

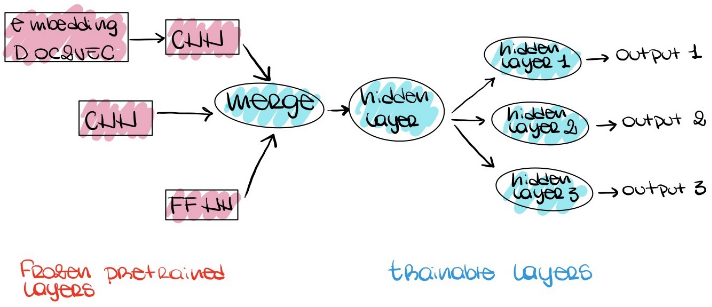


Figure 3: *Step 2: train a multimodal and multitask neural network*

The training and testing of the networks has been carried out through *Google Cloud AI Platform* and the models have been built with *PyTorch* library.

The second goal of the project is to analyse the controversy of the different topics of the articles under which the comments have been posted. The variables representing the topics of the articles are `newDesk` and `sectionName`.
First of all, two types of variables summarizing the 3 targets have been created. The first summary target is binary and takes values 1 if at least one of the three target is positive or 0 otherwise, while the second summary variable has four classes corresponding to the number of positive target variables, so it can take values 0, 1, 2 or 3. Hence, the second variable is a proxy of the intensity of the controversy of a comment. The summary targets have been grouped by topics according to their value and their relative frequency calculated. Only the topics with more than 60% of controversial comments have been considered. The results have been displayed by using a barplot for both the binary and multiclass summary target, in order

to show the composition of controversial comments in terms of intensity.

The last part of the project focuses on using an unsupervised approach for labelling comments based on lexicon-based methods. It will be explored whether using them as alternative labels is a viable alternative to the cases in which no labelled data are available. In fact, labelling data is not always feasible and it might be time and resource consuming, especially when the annotation has to be done manually.
Lexicon-based methods relies on some external dictionary or other sources which already contain annotations about the polarity and intensity of single words. They can be used to retrieve the general sentiment of a sentence or a whole document. In this case, they are applied to single comments. Two ready-to-use available libraries for performing unsupervised sentiment labelling have been used, namely *TextBlob* and *Vader*, both of them built on top of the *NLTK* library.
In order to assess the goodness of these unsupervised labels, their ability to discriminate between classes is tested, both for the binary and for the multiclass summary labels. The first method used is the point biserial correlation, which calculates the correlation between a binary independent variable and a continuous dependent variable, hence it is applied only for the binary target. However, since it is a parametric method it assumes that the variables are normally distributed, which might not be the case.
An additional analysis based on the application of a logistic regression is performed in order to study also the relation between the sentiment labels and the multiclass summary target. Moreover, the method doesn't assume any normality of the dependent variables. The procedure consists in the application of a logistic regression by considering as dependent variable the continuous one and as response variable the categorical one. The score calculated is the *area under the recall-precision curve*, which is particularly suitable in the case of unbalanced classes, and the baseline score is computed as the ratio between the observations in a class (positive observations) and the total observations. The baseline score is equivalent to the case in which the model classifies at random the observations. For the multiclass target case the score is calculated in a one-vs-rest fashion, since the *scikit learn* implementation is restricted to the binary case. The average score is retrieved by using a 10-folds cross validation procedure. In addition, the density distributions and the whiskers and box plots have been plotted for each sentiment variable according to the levels of the summary targets in order to visually compare them.
If these new unsupervised labels result being useful for discriminating among controversial and non-controversial comments, they could be used as alternative targets to train a new model.
In addition, these labels are used for studying the general polarity of comments within a certain topic, which might be thought as an approximation of whether comments are against or in favour of the article. The polarity of comments is explored by means of barplots where the percentage of positive, neutral and negative comments is showed.

Finally, a class in *Python* has been implemented in order to put together all the different parts of the analysis in an interactive way. The class takes as input the id of the comment and returns the following information:

- Original text of the comment
- Keywords of the article the comment refers to
- Prediction of the targets `editorSelection`, `recommendations` and `replyCount` given by the final optimised multimodal neural network model
- Correct values of the targets `editorSelection`, `recommendations` and `replyCount`

- Overall sentiment and scores of the comment according to *TextBlob* and *Vader*
- Barplots of the polarity of the topics of the article
- Barplots of the controversy of the topics of the article

3 Experimental results

All the three single neural networks obtained fairly good results, with a F1 score on the test set around 85%. The final multimodal architecture has been trained with the RMSprop optimiser and a learning rate of 0.0031872049332274466 for 15 epochs with early stopping. Its final F1 score on the test set is 0.8539, hence the precision and recall scores should be sufficiently balanced.

The performance of the final model is quite good, however it might be biased due to the large imbalance of some targets of the training set and due to the naïve implementation of the multimodal neural network. As a consequence, the model might not be particularly sensitive to positive observations occurring rarely in the dataset.

As regards the analysis of the controversy by topic, the number of topics having more than 60% of controversial comments are 19 for the variable `newDesk` and 20 for `sectionName`. Below the distribution of controversial comments is shown for the classes of both variables grouped by the summary binary and multiclass targets.

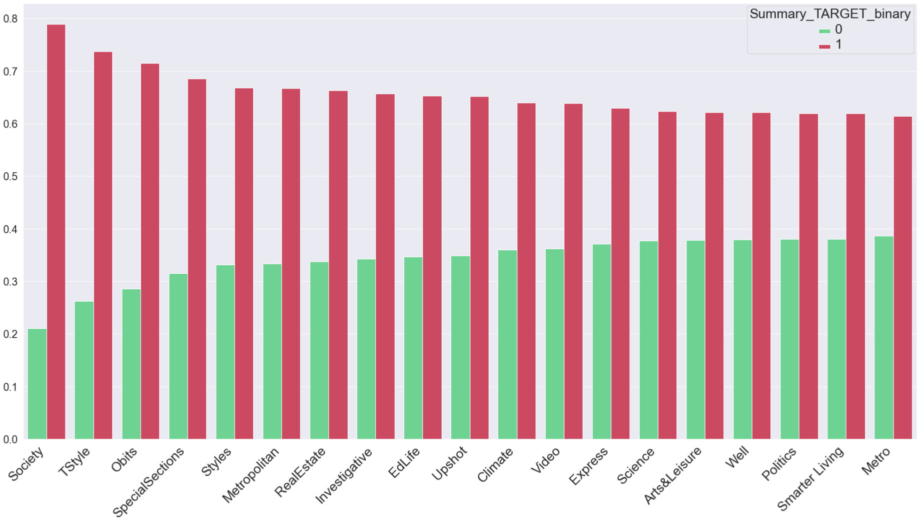


Figure 4: *Frequency barplot of newDesk (controversial vs. not controversial)*

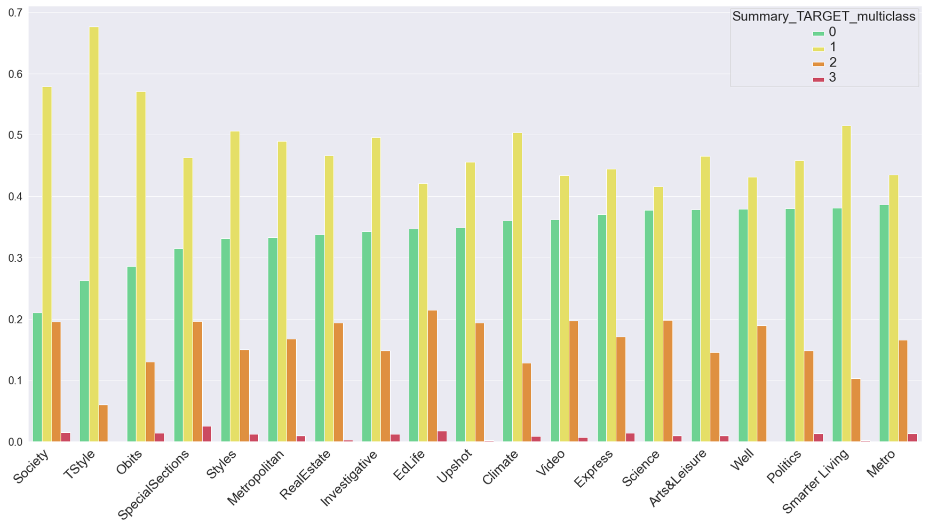


Figure 5: *Frequency barplot of newDesk (4 levels of controversy)*

The top-three controversial topics of `newDesk`, having more than 70% of controversial comments, are *Society*, including various news, *TStyle*, which includes articles about design and interiors, food, travel, fashion and beauty, entertainment and art, and *Obits*, which stands for obituaries and that comprehends articles about the life of famous or noteworthy people who passed away.

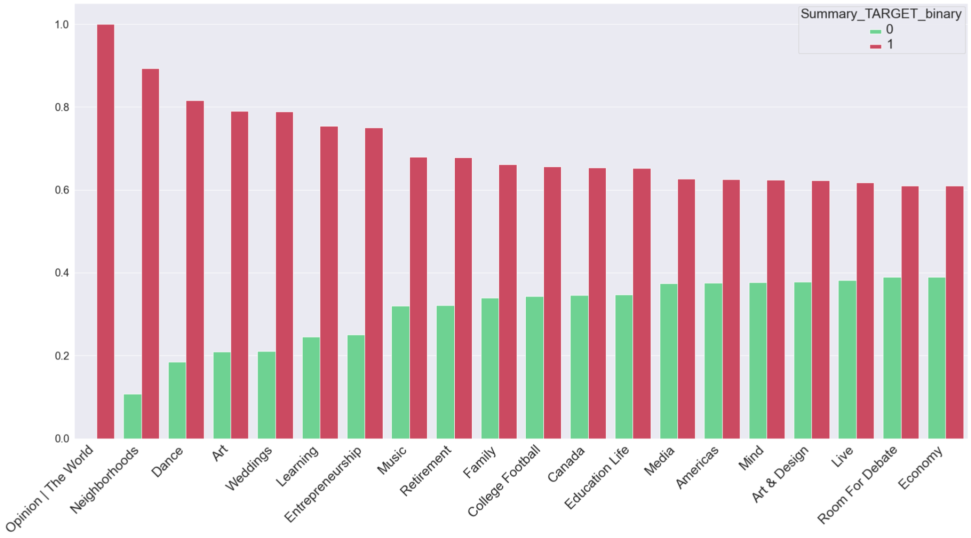


Figure 6: *Frequency barplot of sectionName (controversial vs. not controversial)*

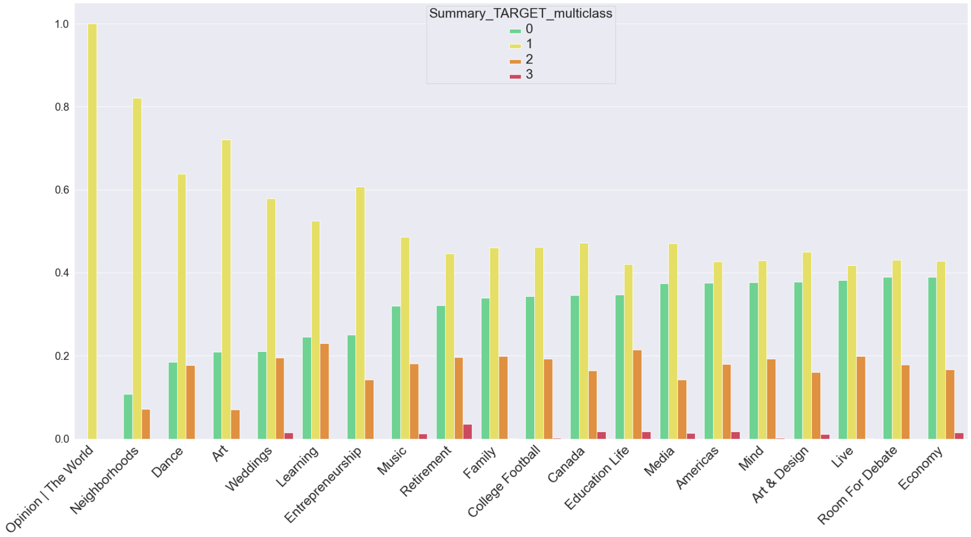


Figure 7: *Frequency barplot of sectionName (4 levels of controversy)*

On the other hand, `sectionName` contains topics with a greater percentage of controversial topics. The top-three topics whose amount of controversial comments surpass 80% of the total comments are *Opinion | The World*, which includes articles from columnists, editorials and guests that reflect their opinion, *Neighborhoods*, which is part of the Real Estate topic, and *Dance*, comprehending articles about everything related to dance, recitals and classical music. It's interesting that the topic *Opinion | The World* contains only controversial topics, however when looking at its composition it can be seen that they are all weakly controverse. Other topics having more than 70% of controversial comments are *Art*, *Weddings*, *Learning* and *Entrepreneurship*.
Most of the controversial topics for both variables are weakly controverse, while a smaller percentage, about one half or one third of the weakly controversial comments, are moderately controversial. Very few comments in each topic are highly controversial.

Regarding the last research question, which is whether unsupervised labels based on the sentiment and polarity of comments could be a proper alternative to the controversy of comments, the analysis clearly suggests a negative answer.
By visually inspecting the distributions of sentiment labels divided by the classes of the summary targets, it can be seen immediately that the distributions compared have almost an identical shape and they are centered around the same values. Below, an example of density distributions for the *TextBlob* application:

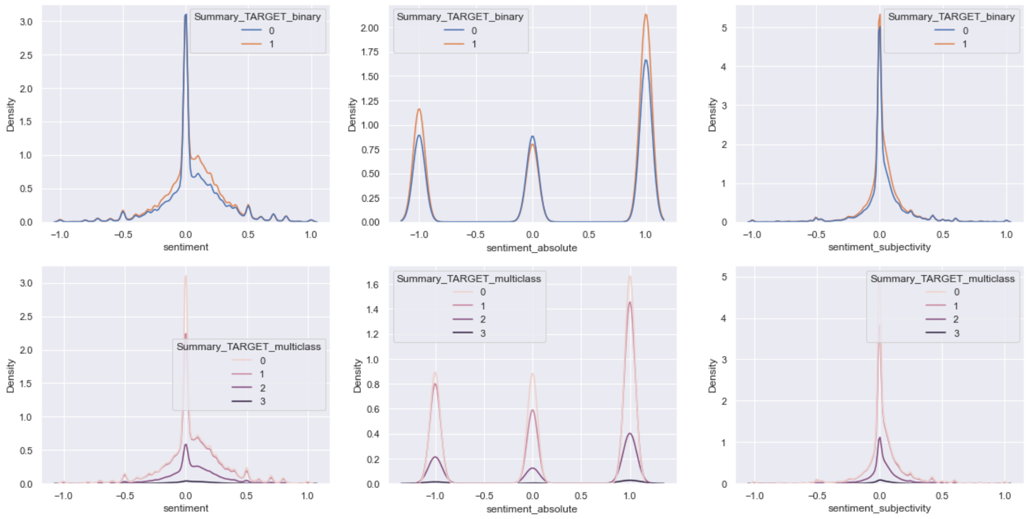


Figure 8: *Distribution of unsupervised sentiment labels divided by class for each summary variable (TextBlob library)*

The values of the *point biserial correlation* and of the *area under the precision-recall curve* from the application of the logistic regression confirm the findings of the visual exploration. The point biserial correlation between the sentiment labels and the binary summary target is about 0 for both libraries' application.

Regarding the second procedure, the *AUPRC* of each positive class is the same as its baseline value, which represents the case in which the classifier guesses totally at random the classes. Hence, the logistic regression is making classifications totally at random since the sentiment labels are completely useless for predicting the true controversy of comments.

Finally, the unsupervised sentiment labels obtained with *TextBlob* and *Vader* are used to study the distribution of the polarity of comments among different topics, for both variables `sectionName` and `newDesk`.

The results from *TextBlob* and *Vader* are reported altogether, however they return pretty similar results. According to `sectionName`, the top most positive topics are *Opinion | The World*, *Food*, *Entertainment*, *Neighborhoods* and *Auto Racing*, where almost all of the comments are positive, while according to `newDesk` they are *TStyle*, *Society*, *Photo*, *SpecialSections*, *EdLife*, *Obits* and *Travel*, where the percentage of positive comments is somewhat lower. On the other hand, the top five topics with comments having a negative polarity are, according to `sectionName`, *Baseball*, *Africa*, *Middle East*, *Opinion | Politics*, *Asia Pacific*, *Room For Debate* and *Canada*, whereas based on `newDesk`, they are *NYTNow*, *Editorial*, *Foreign*, *Washington*, *National*, *Express*, *Automobiles* and *Letters*.

It is worth noting that negative comments across the different topics are found in much smaller percentages with respect to positive ones, which often make up the majority of comments.

Below, the barplots from the *TextBlob* application are shown:

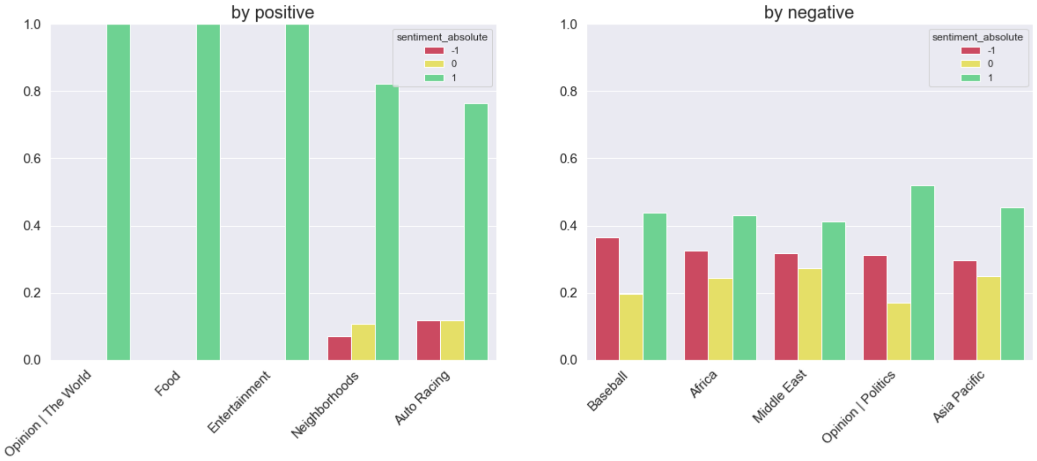


Figure 9: *Frequency barplot of the polarity of sectionName according to TextBlob*

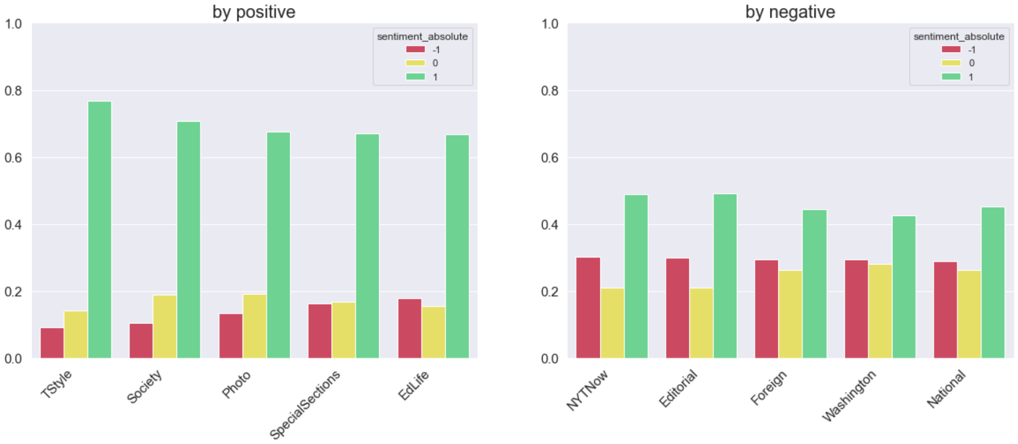
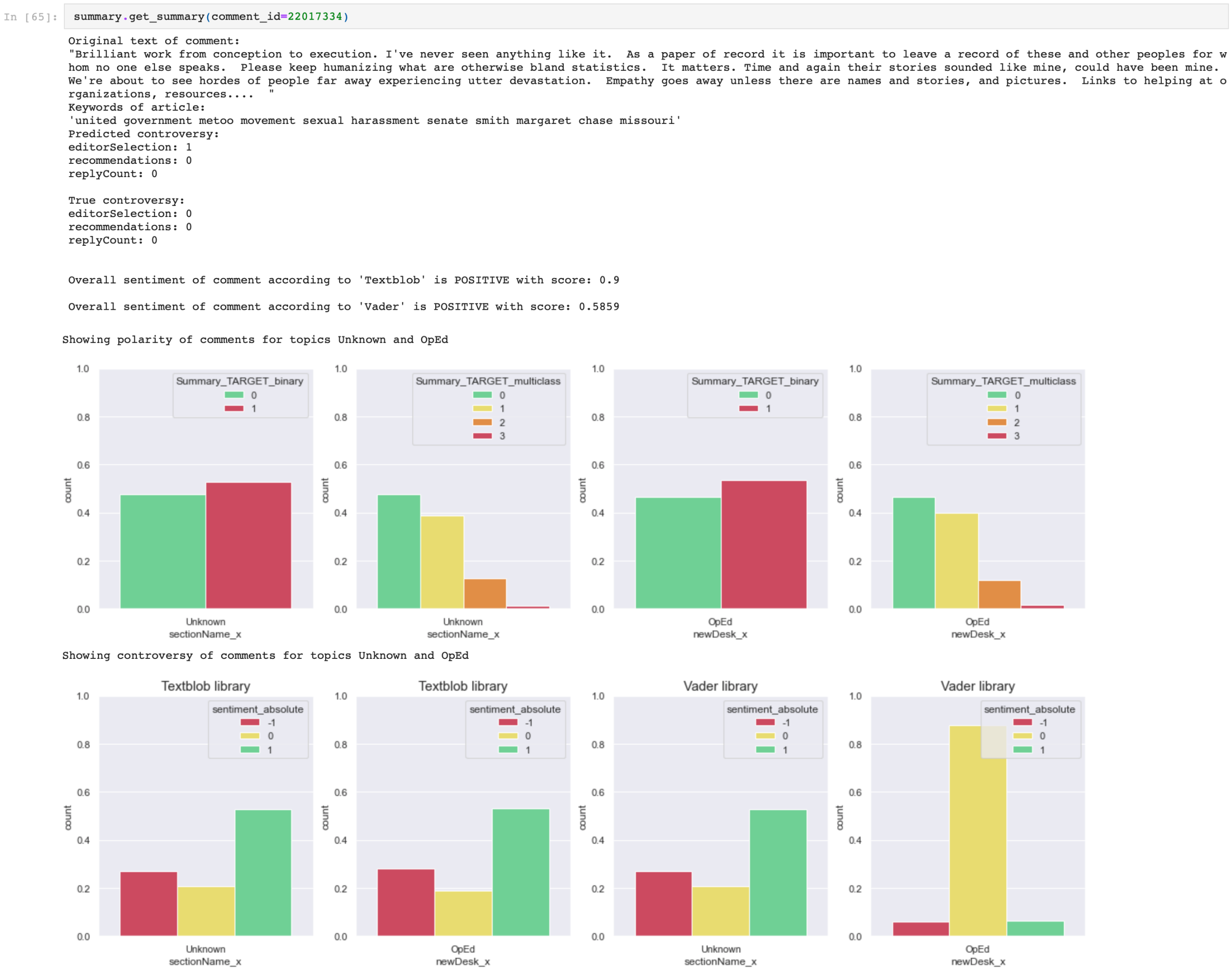


Figure 10: *Frequency barplot of the polarity of newDesk according to TextBlob*

It can be concluded that the alternative unsupervised labelling based on comments polarity and intensity cannot be used to train a new predictive model in this specific case. A possible explanation might be that the types of comments that can be found under the articles of a renowned newspaper, such as the *New York Times*, are probably mostly written with a moderate lexicon. Hence, labelling comments based on their polarity might work better in other types of environments, such as social networks, where comments are far more polarised.
However, these labels proved to be useful to study the polarity of topics, which are well distinguished. Polarity might also be thought as an approximation of whether a comment is against or in favour of the article which it refers to.

Below, the final result of the interactive implementation to retrieve information about single comments:



```
In [65]: summary.get_summary(comment_id=22017334)

Original text of comment:
"Brilliant work from conception to execution. I've never seen anything like it.  As a paper of record it is important to leave a record of these and other peoples for w
hom no one else speaks.  Please keep humanizing what are otherwise bland statistics.  It matters. Time and again their stories sounded like mine, could have been mine.
We're about to see hordes of people far away experiencing utter devastation.  Empathy goes away unless there are names and stories, and pictures.  Links to helping at o
rganizations, resources....  "
Keywords of article:
'united government metoo movement sexual harassment senate smith margaret chase missouri'
Predicted controversy:
editorSelection: 1
recommendations: 0
replyCount: 0

True controversy:
editorSelection: 0
recommendations: 0
replyCount: 0


Overall sentiment of comment according to 'Textblob' is POSITIVE with score: 0.9

Overall sentiment of comment according to 'Vader' is POSITIVE with score: 0.5859

Showing polarity of comments for topics Unknown and OpEd
```

```
Showing controversy of comments for topics Unknown and OpEd
```

Figure 11: *Implementation to return information about single comments*

4 Concluding remarks

The final performance obtained with the multimodal neural network might be enhanced through the refinement of the network architecture and of the data preprocessing. For instance, the training data with unbalanced labels might be balanced by using some undersampling and oversampling techniques, in order to have roughly the same number of positive and negative examples in each batch. It is however important to leave the test set unbalanced in order to better represent the true data. Other possible improvements are that of weighting in different ways the losses coming from different tasks during the training phase or that of using a more sophisticated method with respect to simple concatenation to merge the different neural networks within the multimodal network. Other models apart from neural networks could be explored too.

As regards the use of unsupervised sentiment labels as alternative targets, a possible improvement could be to include all the part of speeches while pre-preprocessing the comments and see whether the labels become informative or to use a more tailored lexicon-based approach to retrieve the sentiment of comments. It might also be the case that this method simply doesn't work for this specific dataset since there is no relation between the polarity of comments and their controversy.

Bibliography

[1] Addlight Mukwazvure, K.P Supreethi
A Hybrid Approach to Sentiment Analysis of News Comments (2015).
In 2015 4th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO)(Trends and Future Directions) (pp. 1-6). IEEE.

[2] Ziegele, M., Breiner, T., & Quiring, O.
What creates interactivity in online news discussions? An exploratory analysis of discussion factors in user comments on news items. (2014)
Journal of Communication, 64(6), 1111-1138.

[3] Shelke, N. M., Deshpande, S., & Thakre, V.
Survey of techniques for opinion mining (2012).

[4] Quoc Le, Tomas Mikolov
Distributed Representations of Sentences and Documents.

[5] Sebastian Ruder
An Overview of Multi-Task Learning in Deep Neural Networks.
AirXiv: 1706.05098v1

[6] Tadas Baltruŝaitis et al.
Multimodal Machine Learning: A survey and Taxonomy.
arXiv:1705.09406v2 International Journal of Computer Applications, 57(13), 0975-8887

[7] *medium.com.*