

Introduction

This report will investigate the A Million News Headlines dataset published by Kaggle user Rohit Kulkarni (Kulkarni, 2017) and use it to build a model¹ that predicts the sentiment of a given headline. We will also compare the Sentiment Analysis models available through RapidMiner Studio's Extract Sentiment operator. We will adjust the parameters of the models in step with each other, and examine the resultant evaluation metrics. From this we can determine which is the best Sentiment Analysis model for this data.

Background

One analysis of this dataset by Kaggle user Ryan Cushen (Cushen, 2019) shows changing sentiments in these headlines over the last ten years, and in particular a downturn in positivity since 2016². Once we have picked a leading Sentiment Analysis model, we can look at sentiment trends over time in this data.

Sentiment Analysis itself is a hot topic in the technology industry. SA is used to automatically extract reviews and ratings for customer services, consumer products and news articles. A company can deploy an SA algorithm to check whether their news coverage is largely positive or negative, or see at a glance how their customers feel about their goods or services.

The data

The A Million News Headlines (AMNH) dataset contains just over a million article headlines from the Australian Broadcasting Corporation website (ABC) over a span of fifteen years. The only variables in the dataset itself are *publish_date* and *headline_text*. We intend to add *Sentiment* scores for each available model.

Assumptions

To perform SA on a dataset, we have to assume that the data contains sentiments. Good journalists will tell you that reportage should be as neutral as possible, but in this case we can expect that headlines will reflect the particular spin that a journalist intends to put on a piece of newsworthy information.

Constraints

This dataset is pulled from the ABC website, and as such is representative of news headlines in Australia. It should reflect world events over the last fifteen years, but will necessarily have a focus on Australian news and opinion. We therefore risk training models that are biased towards the Australian view of the world, whatever that may be.

¹ The final model is provided in SentimentAnalysis.RM.

² Figure 5 available in Appendix.

Costs

One million rows makes a large dataset. Analysing the text in these rows will involve creating a Document Vector, which will consist of thousands of columns. This will require a lot of processing power and development time. Therefore we will first use stratified sampling to look at only 5% of the data, then filter by date, only analysing headlines from the start of 2015 onwards. Then we will filter out neutral headlines.

The tools

Analysis of the SA models and classification model are carried out in the data mining software RapidMiner (Mierswa & Klinkenberg 2018). RapidMiner provides a GUI where a user can drag and drop individual operators into a data mining process. We use this tool in conjunction with learnings from Hoffmann and Klinkenberg (2014) and Data Science lectures at TUD (Leonard 2019).

The process

Process steps are described in more detail in later sections.

Steps to be carried out in RapidMiner:

1. Load dataset from Kaggle repository
2. Take a stratified sample of 5% of the data
3. Filter examples by date (after 31/12/2014)
4. Convert *headline_text* column to text
5. Run SA Model (A/B)
6. Filter out 'neutral' headlines where SA Score = 0
7. Convert SA score variable to generic positive / negative neutral variable *Sentiment*
8. Set *Sentiment* as label
9. Pre-processing steps (described below)
 - a. Create document vector
 - b. Calculate TF-IDF
10. Run SVM classification model on SA scores
 - a. Iteratively improve model performance by adjusting parameter values
11. Generate evaluation metrics for classification model for each SA score

Steps outside of RapidMiner:

1. Research into current SA / Opinion Mining techniques
2. Construct gold standard reporting set of manually annotated headlines
3. Compare output of two models to reporting set
4. Select best-performing classification model

Data Exploration and Descriptive Analytics

Data description

Kaggle user Rohit Kulkarni scraped this data from abc.net.au. The AMNH dataset contains 1,076,141 rows and two columns:

- *publish_date*
String of numbers in format “yyyymmdd”. We can process this as a date on import and use it to track change in headline sentiment over time.
- *headline_text*
String of entire headline. We will process this variable to extract sentiments.

Data exploration

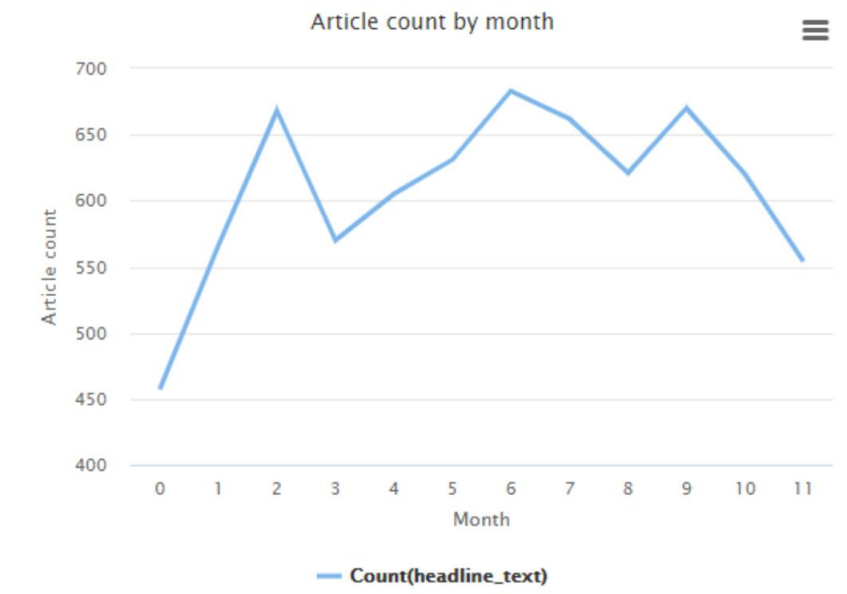
Initial data exploration is limited for this dataset since there are only two variables. We can however see that the one million articles are unevenly distributed over the years and months:

Figure 1: Count of headlines over life of dataset



As shown in Figure 1, our dataset contains more published articles from 2015 than for subsequent years. Within this trend is an uneven distribution over the months of each year, as seen in Figure 2.

Figure 2: Count of headlines over calendar months



We can do further data exploration once we have SA scores for this data.

Data preparation

No data preparation steps were taken ahead of importing the data into RapidMiner. The dataset contains only two columns, both of which are useful, so we include both in this analysis. There are no missing values. No generated records or merged data are added, though we will append SA scores from both models in RapidMiner. Headlines are reformatted as part of the pre-processing steps in RapidMiner.

The text pre-processing steps used are similar to those described by Rameshbhai and Paulose (2019):

1. Tokenise non-linguistic text
2. Tokenise linguistic text
3. Remove stop words
4. Filter tokens by length
5. Stemming (Porter)
6. Convert to lowercase
7. Generate n-grams
8. Calculate TF-IDF

Model iterations

RapidMiner includes a very useful operator called Optimize Parameters. We can use this to loop over the same model with different parameters as we optimise.

The models

The RapidMiner Extract Sentiment operator supports two different Sentiment Analysis models. This operator is available from the Operator Toolbox in the RapidMinder Marketplace (RapidMiner, 2019). Once we have SA scores, we will use further RapidMiner operators to tokenise and classify the headlines according to these scores.

VADER

Valence Aware Dictionary Sentiment Reasoner “is a lexicon and rule-based sentiment analysis tool” (Hutto & Gilbert, 2014). This model is largely trained on text from social media, which was independently scored by human annotators. It also has input from the New York Times and product review websites, expanding its reach to other domains.

SentiWordNet

“SentiWordNet is an opinion lexicon derived from the WordNet database” (Ohana & Tierney, 2009). SentiWordNet was constructed using Rocchio and SVM classifiers, and contains information about how positive and negative an individual word is (PN-polarity). SentiWordNet examines collections of synonyms that are analysed together (Esuli & Sebastiani 2006).

SVM

We will use Support Vector Machine and Term Frequency – Inverse Document Frequency techniques to classify this data. Rameshbhai and Paulose (2019) found that this combination of techniques performed best in their own sentiment analysis study. TF-IDF converts a document into a list of scores for each of its terms. SVMs are a widely-used classification model, and since our data is separated into just two categories (positive sentiment / negative sentiment), this approach fits the task.

Test design

A human annotator annotated a subset of 100 headlines for each model, and these scores were compared to those predicted by the models.

The SVM model can be evaluated using standard evaluation methods, but again we are evaluating each against the sentiments given by the two SA models, which are themselves not 100% reliable. Our test design is as follows:

1. Manually construct reporting set
2. Check output of each SA model against reporting set
3. Train SVM on headlines plus *Sentiment* scores for each model (A/B)
4. Report accuracy for each set of parameters
5. Adjust parameters, re-train, report
6. Look at best-performing models in detail: precision, recall, *Sentiment* scores (A/B)
7. Compare sentiment over time to existing literature, eg Cushen 2019

Model iterations

As discussed above, we looked at the SA classifications from two different models:

- SentiWordNet
- VADER

The parameters which we manipulated for SVMs are as follows:

- Kernel type
- Kernel gamma
- Convergence epsilon

We looked at which SA model provided the most robust results, and tweaked the parameters of the SVM to optimise accuracy, precision and recall. Other model parameters, such as C and Kernel degree, were found to have no discernible effect on model accuracy. We kept these at their default settings. We kept train/test split at the default setting of 0.7/0.3, since this is also standard across most of the literature (Raschka, 2018). We kept a random local seed to keep samples constant over many iterations.

SA models

The SA models investigated here produced different classifications for the same headlines, and therefore had different accuracy ratings against the reporting set. SentiWordNet tended to assign more headlines to the positive and negative classes, while VADER assigned more to the neutral class. The results of this manual investigation are shown in Table 1.

Table 1: Results of manual annotation investigation into two SA models³

| | VADER | SentiWordNet |
|----------|-------|--------------|
| Accuracy | 0.77 | 0.53 |

VADER has a higher overall accuracy rate, but further investigation shows that its predictions are skewed towards the negative class. A comparison of the headlines that appeared in both reporting sets showed that the models agreed on classifications 74% of the time.

Kernel type

The most commonly-used SVM kernel function is Radial Basis Function (RBF) (Dataflair, 2018), which looks for a circular decision boundary to separate classes. In our data we're looking for a linear boundary, so these models performed poorly. We also tested Dot product, Polynomial and Multiquadratic kernels with different gamma and epsilon settings, with poor results (Table 3). ANOVA kernel functions performed best across the board, with both SentiWordNet and VADER-classified headlines.

Table 3: Best performing parameter combinations for each model⁴

³ Further details are available in Tables 7 and 8 in the Appendix.

⁴ Further results are available in Table 9 in the Appendix.

| Kernel type | Kernel gamma | Convergence epsilon | Kernel degree | Accuracy (SentiWordNet) | Accuracy (VADER) |
|----------------|--------------|---------------------|---------------|-------------------------|------------------|
| Dot product | NA | 0.01 | NA | 0.548 | 0.774 |
| Polynomial | NA | 0.001 | 2 | 0.547 | 0.602 |
| Multiquadratic | NA | 0.001 | NA | 0.548 | 0.600 |
| RBF | 1 | 0.001 | NA | 0.547 | 0.600 |
| ANOVA | 1 | 0.01 | 2 | 0.580 | 0.858 |

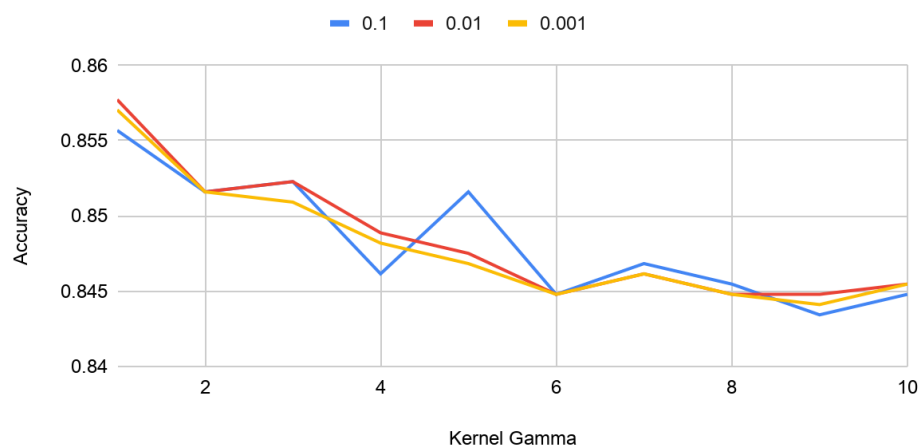
Kernel gamma

We found an optimal setting with an ANOVA function and a kernel gamma setting of 1. Kernel gamma defines the complexity of the decision boundary: a boundary more fitted to the training set allows for more detail in classification decisions, but also risks over-fitting (Bhattacharyya, 2018). Figure 3 shows the change in accuracy rates as we changed the kernel gamma setting for the VADER data. Different values for convergence epsilon are shown as coloured lines on the same graph, and the accuracy of each model declines as the gamma value is increased.

Figure 3: Accuracy scores and Kernel Gamma changes for ANOVA model

Kernel Gamma and Accuracy

For different values of epsilon, C = 0.7



Convergence epsilon

Convergence epsilon defines how the model recognises datapoints as errors. A smaller value allows for a smaller distance (error) between prediction x and true label y , while a larger value is more permissive. As seen above, we found that a medium value convergence epsilon (0.01) allowed our model to perform slightly better than it did with smaller or bigger values.

Evaluation

First, we got a baseline accuracy for both the VADER and SentiWordNet SA models by comparing their predictions to the reporting set. In this we found that VADER performs better overall, but it is skewed towards negative classifications.

Secondly, we used output from both these SA models to train a series of SVMs with different parameters. Our final model and process are presented in SentimentAnalysis.RM. This contains an SVM model with the following parameters:

Table 4: Final model parameters

| Sample size | Train/test split | Kernel type | Kernel gamma | Convergence epsilon | C | Kernel degree |
|-------------|------------------|-------------|--------------|---------------------|-----|---------------|
| 5% | 0.7/0.3 | ANOVA | 1 | 0.01 | 0.7 | 2 |

This model performed well on data tagged by both SA models, but particularly with data tagged by the VADER SA model. It achieved accuracy of 85.4% and an AUC of 0.933⁵.

Table 5: Confusion matrix for final model trained on VADER

| | Predicted positive | Predicted negative | Class precision | |
|---------------|--------------------|--------------------|-----------------|-------|
| True positive | 796 | 130 | 0.86 | |
| True negative | 85 | 458 | 0.844 | |
| Class recall | 0.904 | 0.779 | Accuracy | 0.854 |

Process Review

Newspaper headlines are written for brevity and to be eye-catching: they don't need to be technically grammatically correct or even contain complete words. We therefore may need an SA model that is specifically trained on newspaper headlines and optimised for this kind of content. The best SA model available in RapidMiner for this purpose was VADER, which is partially trained on New York Times text.

As noted above, we only carried out manual annotation on a small subset of headlines, and given more resources this subset would be larger. Having this reporting set was a useful tool, however, and helped to explain the difference in performance of models trained with the VADER SA scores as opposed to the SentiWordNet scores.

We filtered our sample by date and neutrality so that we could reduce the size of the sample and focus in on our main question. Our end goal is to compare this data to that reported by Cushen (2019), so it's only necessary to look at headlines around the reported downturn of sentiment- from 2015 to 2017. Filtering out neutral headlines allowed us to focus on the positive/negative sentiment split, and visualise more easily the change in mood.

This research only discovered the Optimize Parameters operator quite late in this process. Aside from annoying the researcher, this meant that the overall time taken running models for the project could have been vastly reduced.

⁵Figure 6 available in Appendix.

Next Steps

We found that the VADER SA model performs better with this data: both compared to the reporting set and when used to classify the data set as a whole.

This initial investigation looked into binary classification of headlines, but the output of both SA models allows for many different kinds of questions. The *Score* is a real number, so we could train a Linear Regression model on this data, too.

Analysis

SA models

We expected SentiWordNet to perform better than VADER, since the former has a more complex understanding of word usage- grouping terms together into synsets of similar words and assigning different sentiment scores to each. VADER's performance however can be explained by its training data containing Social Media and New York Times data. A look in detail at the output for the SA models shows us that they saw the same headlines very differently:

Table 6: Selected headlines and analysis from SA models

| Headline | Reporting set class | VADER class | SentiWordNet class | VADER terms | SentiWordNet terms |
|---|---------------------|-------------|--------------------|--|---|
| riots mar new year's eve celebrations in wa | FALSE | FALSE | TRUE | riots (-0.59) | mar (-0.02) new (0.05) |
| qld fraud squad failed to act on palmer adviser fraud claim | FALSE | FALSE | TRUE | fraud (-0.72) failed (-0.59) fraud (-0.72) | fraud (-0.03) act (0.05) on (0.02) fraud (-0.03) claim (0.03) |
| federer recovers from early nerves to defeat bolelli | TRUE | FALSE | TRUE | nerves (-0.10) defeat (-0.51) | early (0.02) nerves (0.03) |

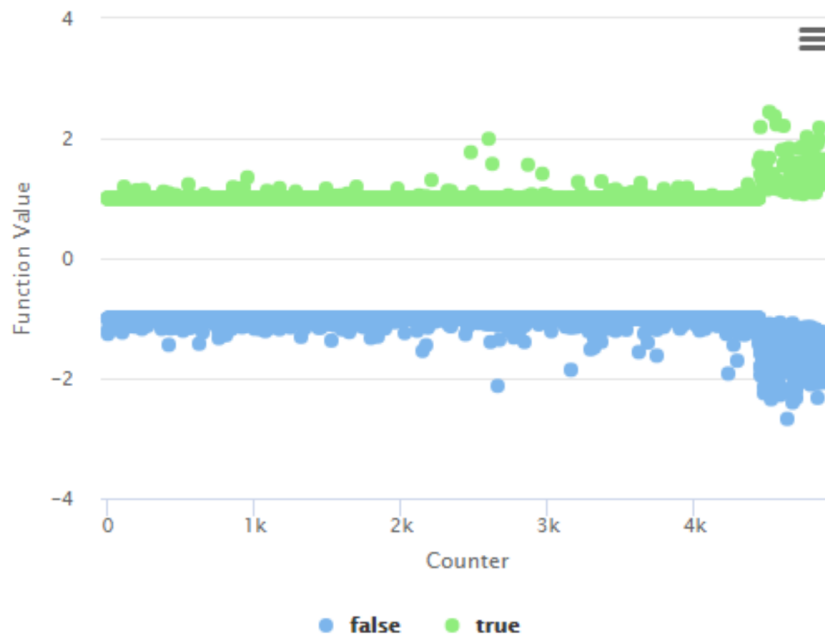
In the first example, SentiWordNet missed the term 'riots' completely, while the reporting set and VADER both took this and its negative connotations into account. In the second example, both SA models identified the term 'fraud', but gave it different weightings: VADER gives this a sentiment score of -0.72, while SentiWordNet gives it only -0.03. VADER therefore considers 'fraud' *more* negative a term than SentiWordNet does. Also, SentiWordNet considers 'on' a positive term, which is odd, and raises questions about the order of operators in our process. Finally, the third example in Table 6 shows another difficulty in SA: whose sentiment are we supposed to consider? In this example, a Federer fan will find this headline largely positive, while a Bolelli fan would disagree. Our models also disagreed: VADER classed Federer's 'nerves' and Bolelli's 'defeat' both as negative, while SentiWordNet thought the 'nerves' and their 'early'-ness were both positive. In this case, SentiWordNet agrees with the reporting set, though it's possible this is just by chance. We'd expect an SA model to pay attention to a word like 'defeat', even if it leads to an incorrect classification.

TF-IDF and SVM

The approved model contains 4898 support vectors and has a bias of -8.73, suggesting it is biased towards negative Sentiment scores, much like VADER is itself. With so many features

and support vectors, the weight for each one is absolutely tiny. A future piece of research could filter out some terms or employ Principle Component Analysis to reduce the number of features.

Figure 4: final classified data

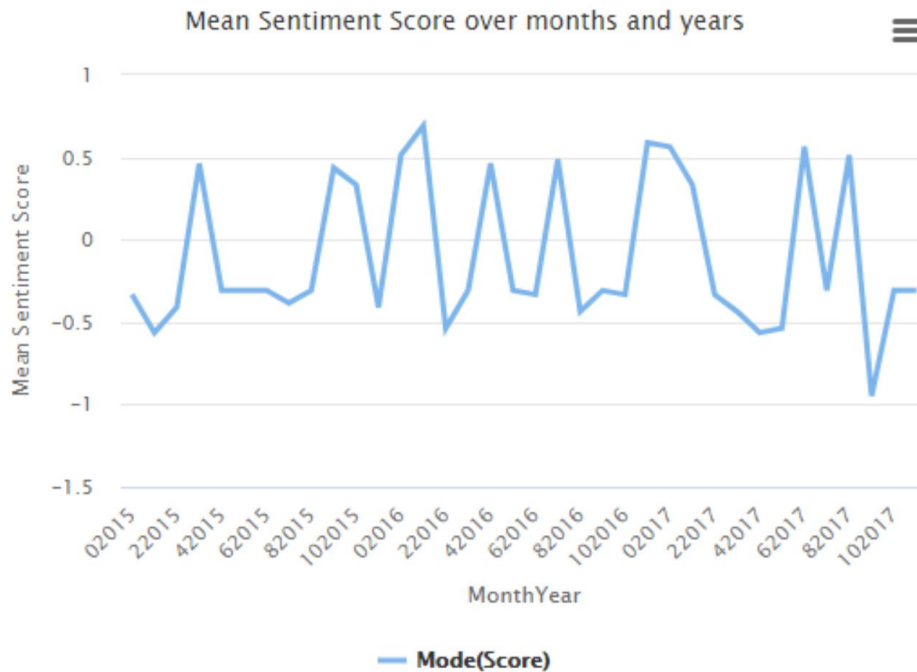


We used the standard SVM operator in order to compare different kernel types. A future study could also use the Linear SVM operator to evaluate differences in performance.

Available literature

When comparing our model output to Cushen (2019) (Figure 5 available in Appendix), we can note differences in the direction of sentiment over the years.

Figure 5: Mean Sentiment Score over months and years



Our data analysis doesn't show the same trend as Cushen: there isn't a visible downward trend in sentiment scores around the year 2016. This could be due to a number of factors: differences in tools, techniques or filtering could remove the most telling instances from our dataset.

An Australian view of the world

We can use this analysis to investigate the individual terms and phrases that our model picked up. Below, we can see that the dataset contained many thousands of mentions of the terms 'aborigin' (a word stem that would include terms like 'aboriginal') and 'abbott' (likely to refer to Tony Abbott, Australian PM from 2013-2015). Here we can see that opinion is divided among our headline writers about these two topics.

Figure 6: 'aborigin' word stem, mentions and sentiment (colour)

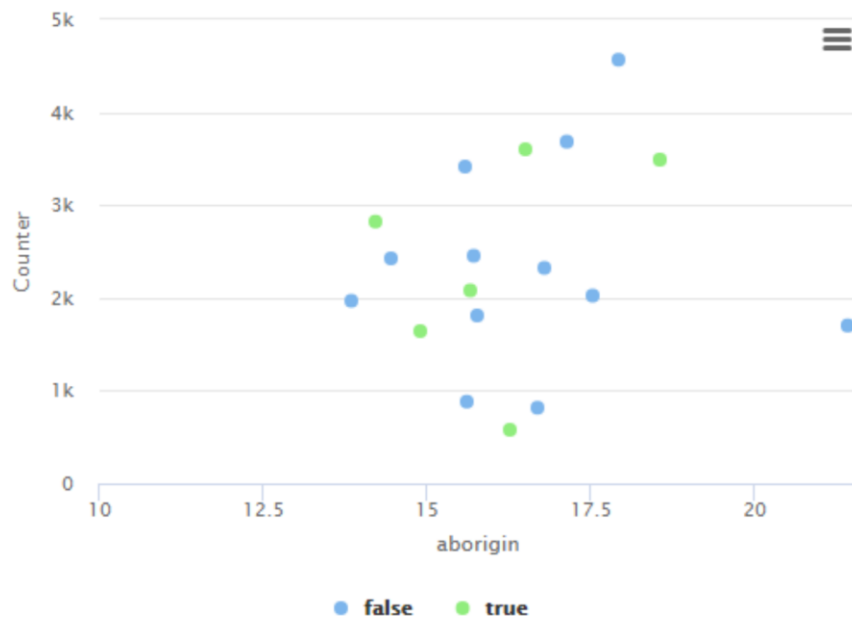
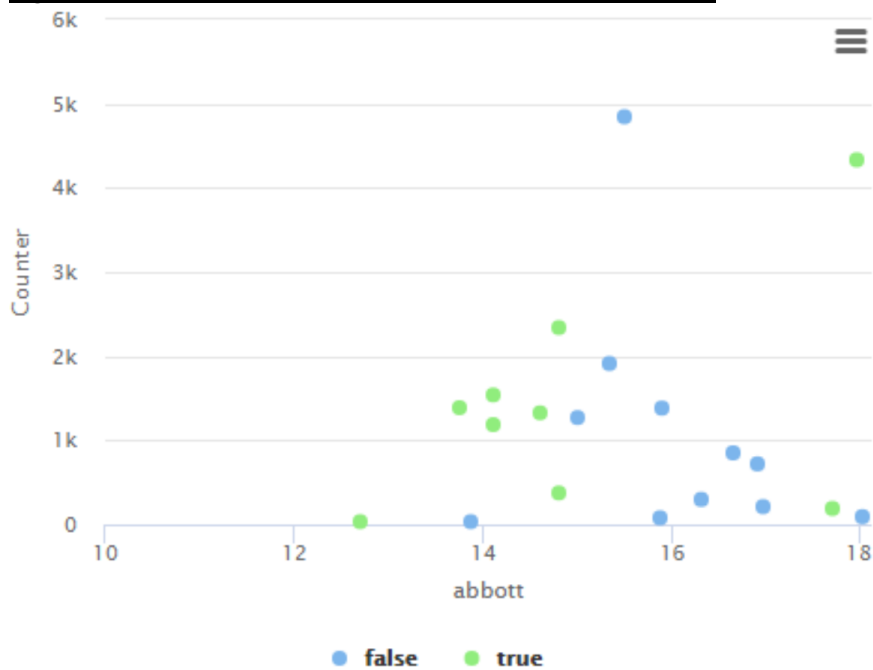


Figure 7: 'abbott' word stem, mentions and sentiment (colour)



Other terms, such as 'justic' and 'fatal' were unanimously positive or negative, and didn't produce any interesting graphs at all.

Appendix

Figure 8: Mean sentiment polarity scores from Cushen (2019)

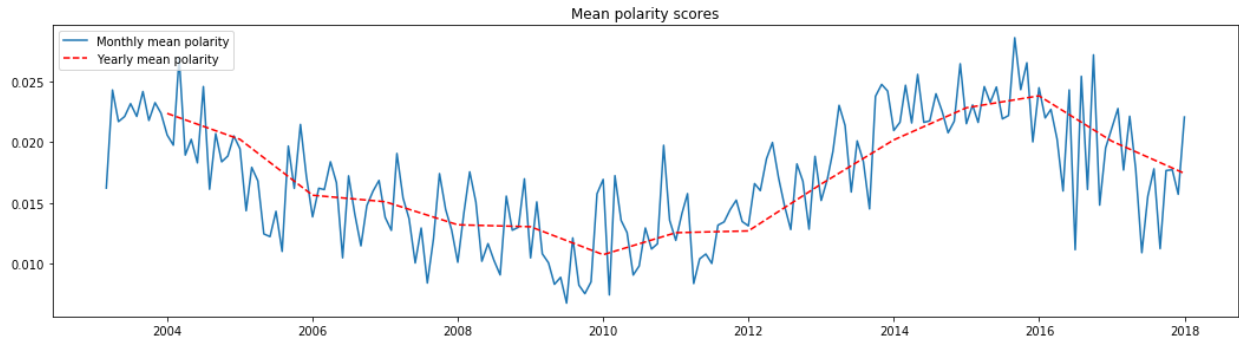


Table 7: Results of manual annotation investigation into VADER SA model

| VADER | Predicted Positive | Predicted Negative | Predicted Neutral | Class precision | |
|---------------|--------------------|--------------------|-------------------|-----------------|------|
| True positive | 26 | 9 | 0 | 0.703 | |
| True negative | 6 | 51 | 0 | 0.81 | |
| True neutral | 5 | 3 | 0 | 0 | |
| Class recall | 0.684 | 0.895 | 0 | Accuracy | 0.77 |

Table 8: Results of manual annotation investigation into SentiWordNet SA model

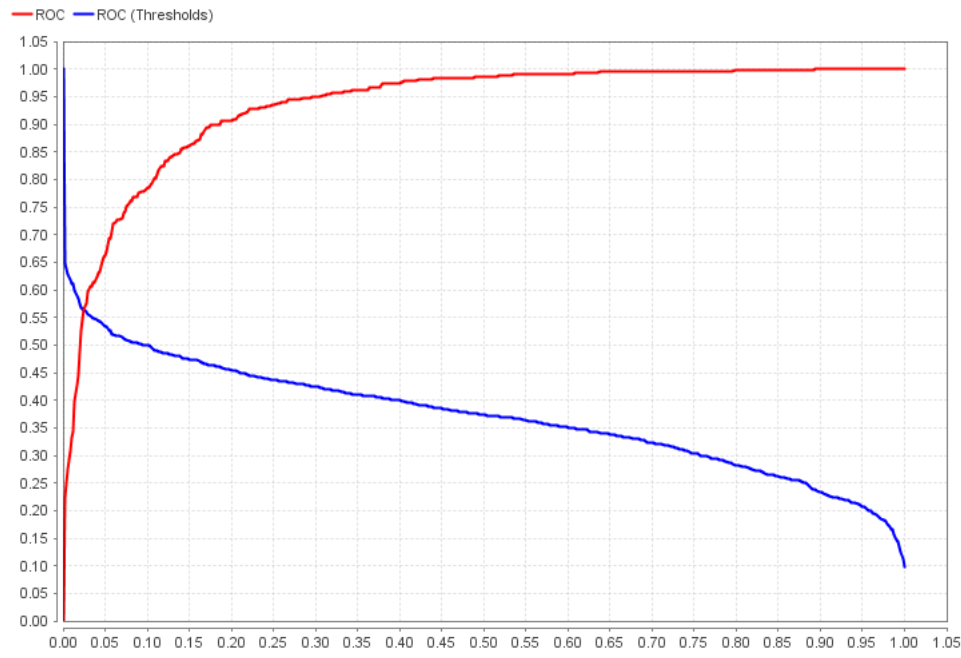
| SentiWordNet | Predicted Positive | Predicted Negative | Predicted Neutral | Class precision | |
|---------------|--------------------|--------------------|-------------------|-----------------|------|
| True positive | 33 | 12 | 0 | 0.611 | |
| True negative | 16 | 20 | 0 | 0.488 | |
| True neutral | 10 | 9 | 0 | 0 | |
| Class recall | 0.559 | 0.556 | 0 | Accuracy | 0.53 |

Table 9: Extra results table for all model types

| Kernel type | Kernel gamma | Convergence epsilon | Accuracy (SentiWordNet) | Accuracy (VADER) |
|----------------|--------------|---------------------|-------------------------|------------------|
| Dot product | NA | 0.01 | 0.548 | 0.774 |
| Polynomial | NA | 0.01 | 0.547 | 0.602 |
| Multiquadratic | NA | 0.01 | 0.548 | 0.6 |
| RBF | 1 | 0.01 | 0.547 | 0.6 |

| | | | | |
|-------|----|-------|--------------|-------|
| | 10 | 0.01 | 0.547 | 0.6 |
| ANOVA | 5 | 0.01 | 0.6090328467 | 0.848 |
| | 10 | 0.001 | 0.4520985401 | 0.845 |

Figure 9: ROC curve for final model



References

Australian Broadcasting Corporation. <https://abc.net.au>

Bhattacharyya, S. (2018). Support Vector Machine: Kernel Trick; Mercer's Theorem. December 2018, Accessed November 2019. <https://towardsdatascience.com/understanding-support-vector-machine-part-2-kernel-trick-mercers-theorem-e1e6848c6c4d>

Cushen, R. (2019). "A Basic Implementation of Sentiment Analysis." Kaggle March 2019, accessed December 2019. <https://www.kaggle.com/rcushen/a-basic-implementation-of-sentiment-analysis>

Dataflair (2018). Kernel Functions- Introduction to SVM Kernel & Examples. Data-Flair.Training, November 2018, accessed November 2019. <https://data-flair.training/blogs/svm-kernel-functions>

Esuli A, Sebastiani F. (2006). SentiWordNet: A Publicly Available Lexical Resource for Opinion Mining. Proceedings from International Conference on Language Resources and Evaluation (LREC), Genoa, 2006. http://www.lrec-conf.org/proceedings/lrec2006/pdf/384_pdf.pdf

Hoffmann, R. & Klinkenberg, R. (eds.) (2014). RapidMiner: Data Mining Use Cases and Business Analytics Applications. CRC Press.

Hutto, C.J. & Gilbert, E.E. (2014). VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. Eighth International Conference on Weblogs and Social Media (ICWSM-14). Ann Arbor, MI, June 2014.
<https://github.com/cjhutto/vaderSentiment>

Kulkarni, R. (2017). "A Million News Headlines | Kaggle", Kaggle, July 2017, Accessed November 2019, <https://www.kaggle.com/therohk/million-headlines>

Leonard, D. (2019). Data Mining, Data Science Postgraduate Certificate, Part-Time, TU069, Technical University of Dublin, Dublin, Winter 2019.

Mierswa, I., & Klinkenberg, R. (2018). RapidMiner Studio (9.1) [Data science, machine learning, predictive analytics]. Retrieved from <https://rapidminer.com/>

Ohana, B. & Tierney, B. (2009). Sentiment classification of reviews using SentiWordNet. School of Computing 9th IT & T Conference. Dublin Institute of Technology, Dublin, 2009.
https://www.scss.tcd.ie/Khurshid.Ahmad/Research/Sentiments/K_Teams_Buchraest/viewcontent.pdf

Rameshbhai, C.J., Paulose, J. (2019), Opinion mining on newspaper headlines using SVM and NLP. International Journal of Electrical and Computer Engineering (IJECE), Dept. of Computer Science, Christ University, India, June 2019.
https://pdfs.semanticscholar.org/befe/6606f71935f3cd60dde8951f22fe865762ce.pdf?_ga=2.7608114.1413312263.1574857627-1947983167.1573751867

Raschka, S. (2018). Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning. University of Wisconsin–Madison Department of Statistics, November 2018. Accessed December 2019 <https://arxiv.org/pdf/1811.12808.pdf>

RapidMiner, (2019), "RapidMiner Marketplace", Operator Toolbox v2.2.0 September 2019, Accessed November 2019,
https://marketplace.rapidminer.com/UpdateServer/faces/product_details.xhtml?productId=rmx_operator_toolbox