Utrecht University

Data Mining: Assignment 2

TEXT CLASSIFICATION FOR THE DETECTION OF OPINION SPAM

Ana Borovac
6584446

Argyro (Iro) Sfoungari
6528015

October, 2018

Contents

Abstract

1 Introduction

Nowadays email filters are able to classify spam and not spam emails quite successfully. Imagine now that you have a web site with hotel reviews and your goal is to offer the most truthful reviews but you can not check every single review and it is also hard to recognise when the review is not truethful. Therefore you would like to have a mechanism which is going to help you to achieve your goal.

In this assignment we tried to solve the above problem with 4 methods; Naive Bayes, logistic regression, classification tree and random forests. Before we started, we analysed the work that has been done already (section 2). Next, we did some data preprocessing, it is described in the section 3. Used methods are explained in the section 4 and analysis of the results (section 5) is in the section 6.

2 Related work

The authors of "Finding Deceptive Opinion Spam by Any Stretch of the Imagination" [8] compared truthful and deceptive positive reviews for hotels. They used Naive Bayes and Support Vectors Machine classifiers. For comparison they also had 3 human untrained judges which tried to predict if a review was real or fake. The results had shown that automated classifiers outperform human judges in almost every metric (precision, recall, F-score). They explain that with that untrained humans often focus on unreliable cues to deception. One of the results was also that models trained only on unigrams outperformed all non–text-categorizations approaches (genre identification and psycholingustic deception detection). Furthermore, the results were even better when bigrams were used.

In the article "Negative Deceptive Opinion Spam" [7] the authors created corpus of gold standard 400 reviews on 20 Chicago hotels and then used them to compare n-gram–based Support Vector Machine classifiers with untrained human judges. They concluded that the best detection performance was achived through automated classifiers.

3 Data

Our data consisted of 400 negative deceptive and 400 negative truthful hotel reviews that have been collected by Myble Ott and others ([7], [8]).

3.1 Data preprocessing

Before modeling we did some data preprocessing in order to get better models for predicting if a review is fake or real. First step in data preprocessing was

to remove punctuation marks, after that we made every letter lower case, we also removed stopwords, numbers and excess whitespace.

Next, we created a test and a training set. Because we wanted to use cross validation, we divided our data into 5 folds (each of size 160 samples - 80 fake reviews and 80 truthful reviews). So, 4 of the folds represented a training set, the remaining 5th fold was a test set.

At that moment every unique word from the training reviews was a feature. A number of features was large therefore we removed the words as features which occur less than 1 % of training documents.

After that we created a new training set which consisted all the features from the previous training set and bigrams. We again removed bigrams that occur in lass that 5 % of training documents. At the end we had two training sets ready to be used.

4 Methods

We used different classifiers to model our task:

- Naive Bayes (generative linear classifier) – subsection 4.1,
- logistic regression (discriminative linear classifier) – subsection 4.2,
- classification tree (flexible classifier) – subsection 4.3,
- random forests (flexible classifier) – subsection 4.4.

4.1 Naive Bayes

Naive Bayes is a probabilistic classifier. Class is predicted from features which has highest probability with independence assumption (features are independent within each class) [2]:

$$\hat{c} = \arg\max_{c \in C} P(c) \prod_{i=1}^{m} P(x_i|c)$$

For Naive Bayes we used the code from the lectures [1].

4.2 Logistic regression

Logistic regression is one of discriminative classification methods [4]. That means that modelling of probability, how likely are we to predict a class c with given input, is direct. In a binary case we predict class 1 if it holds:

$$\frac{P(Y=1|x)}{P(Y=0|x)} > 1; \quad P(Y=1|x) = \frac{1}{1+e^{-\beta^T x}}$$

otherwise we predict class 0.

For logistic regression model we used `cv.glmnet` function from `glmnet` library [5]. The function does k-fold cross-validation and as a result returns a value for `lambda`. We left the default value of k, which is set to 10. Since we that our model is binomial (`family = ''binomial''`), we were able to set `type.measure` to "class". This means that the loss which is used for cross-validation is misclassification error. When predicting the classes for the test samples, we first used largest lambda s.t. error is within 1 standard error of the minimum.

4.3 Classification tree

Classification trees are usually not the best models [3], but are easy to interpret and can handle both numerical and categorical attributes.

For growing the classification tree we used a function `rpart` from R library also called `rpart` [9]. We set `method` parameter to "class" and complexity parameter `cp` to 0. Library also contains a function for pruning the tree, `prune`. First, we used it with complexity parameter `cp` equals to 0,001.

4.4 Random forests

Random forests are improved classification trees, where the best split is chosen among k random features and not among all of them [6].

We grew a random forest of 200 trees with the function `randomForest` (R library `randomForest` [6]). At the begining we set `mtry` parameter to 6 which means that on each step the algorithm selects 6 random observed features.

5 Results

6 Analysis

7 Conclusion

References

[1] Ad Feelders. Code for naive bayes. `www.cs.uu.nl/docs/vakken/mdm/mnb.txt`. Online; accessed 24 October 2018.

[2] Ad Feelders. Text classification. `www.cs.uu.nl/docs/vakken/mdm/Slides/dm-text-naivebayes.pdf`, October 2017. Online; accessed 30 October 2018.

[3] Ad Feelders. Classification trees (1). `www.cs.uu.nl/docs/vakken/mdm/Slides/dm-classtrees-1-2018.pdf`, 2018. Online; accessed 30 October 2018.

[4] Ad Feelders. Logistic regression. `www.cs.uu.nl/docs/vakken/mdm/Slides/dm-logreg2018.pdf`, 2018. Online; accessed 30 October 2018.

[5] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22, 2010.

[6] Andy Liaw and Matthew Wiener. Classification and regression by randomforest. *R News*, 2(3):18–22, 2002.

[7] Myle Ott, Claire Cardie, and Jeffrey T. Hancock. Negative deceptive opinion spam. In *in HLT-NAACL*, 2013.

[8] Myle Ott, Yejin Choi, Claire Cardie, and Jeffrey T. Hancock. Finding deceptive opinion spam by any stretch of the imagination. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 309–319, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[9] Terry Therneau, Beth Atkinson, and Brian Ripley. *rpart: Recursive Partitioning and Regression Trees*, 2015. R package version 4.1-10.