

Detecting Satire in the News with Machine Learning

Stöckl, Andreas

Published: 01/10/2018

[Link to publication in pure](#)

Citation for published version (APA):

Stöckl, A. (2018). *Detecting Satire in the News with Machine Learning*.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain.
- You may freely distribute the URL identifying the publication in the public portal.

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Detecting Satire in the News with Machine Learning

Andreas Stöckl

andreas.stoeckl@fh-hagenberg.at
School of Informatics, Communications and Media
University of Applied Sciences Upper Austria
Softwarepark 11, 4232 Hagenberg, Austria

Abstract

We built models with Logistic Regression and linear Support Vector Machines on a large dataset consisting of regular news articles and news from satirical websites, and showed that such linear classifiers on a corpus with about 60,000 articles can perform with a precision of 98.7% and a recall of 95.2% on a random test set of the news.

On the other hand, when testing the classifier on “publication sources” which are completely unknown during training, only an accuracy of 88.2% and an F1-score of 76.3% are achieved.

As another result, we showed that the same algorithm can distinguish between news written by the news agency itself and paid articles from customers. Here the results had an accuracy of 99%.

1 Introduction

(Banko and Brill, 2001) determined that, with some learning curve experiments for typical natural language classification tasks, the performance of learners can benefit significantly from much larger training sets. They propose to direct efforts towards increasing the size of annotated training collections, while reducing the effort on comparing different learning techniques and features trained only on small training corpora.

(Halevy et al., 2009) proposed that the biggest successes in natural-language-related machine learning have been in domains in which a large training set is available. For the feature selection task, we should trust that human language has already evolved words for the most important concepts. Thus we can proceed by tying together the words that are already there, rather than by inventing new concepts.

In this paper, we experiment with methods for training a machine learning system to detect satirical or fake news by analyzing the text content

of the article. The amount of online published news articles has been rapidly growing over the last decades and is still steadily increasing. We assembled a new text corpus for training and testing from German news articles published over the last decade. The source for the non-fake articles are documents published by news agencies and newspapers on the web. As training data for the fake news, we use documents published by satirical news websites, and so they are declared as fake and can be used for supervised learning methods (see Section 3). The articles come from different domains such as politics, business, technology and others.

Sometimes it seems even difficult for humans to recognize the satirical or fake character of news articles. We endeavor to test which level confidence an algorithm can achieve by analyzing the pure text content without any information like links to other articles. For humans, to determine the trustworthiness of an article, it is very important to know who published the content. Humans learn to trust some news sources and not others. We will test how machine learning performs if the systems knows the sources in the sense that it was trained on texts published there. To see if the system can detect satirical content without knowing any articles from the publisher, we also test on a set of articles from publishers we did not use in any way for training.

2 Related work

There are numerous linguistic approaches for fake news detection which use simple methods of representing texts, such as the bag of words or n-grams or analyzing deeper language structures.

And there are network approaches which use the network structure of linked data or social network behavior. A good overview can be found in

(Conroy et al., 2015). The goal in the linguistic approaches is to find deception cues like frequencies and patterns of pronoun, conjunction, and negative emotion word usage. (Feng and Hirst, 2013) worked with n-grams features and deep syntax features to differentiate genuine and fake product reviews.

An example of a network approach can be found in (Tacchini et al., 2017) in which Facebook posts are classified as hoaxes or non-hoaxes on the basis of the users who liked them. They used logistic regression and other techniques on a dataset consisting of 15,500 Facebook posts and 909,236 users, and they obtained classification accuracies exceeding 99%.

(Wang, 2017) presented a dataset for fake news detection consisting of 12.8K manually labeled short statements in various contexts from PolitiFact.com. They implemented a hybrid approach with a convolutional neural network to integrate meta-data with text. (Amador et al., 2017) were characterizing political fake news on Twitter through the analysis of their meta-data.

(Rubin et al., 2016) analyzed a very similar setup as we do, by looking at news articles from the newspapers *The Toronto Star* and *The New York Times* and the satirical news sites *The Onion* and *The Beaverton*. They did so by looking only at a small set of articles (480 documents) and searching for handcrafted features which can be used for regression methods or support vector machines. They found a precision of 0.78 and a recall of 0.87 and therefore an F1-score of 0.82 for a baseline implementation using only tf-idf topic vectors. They tried out different additional features in their setup suggested from past literature, and found that they could improve performance slightly to an F1-score of 0.87. The features tested were combinations of things like the presence of new named entities at the end of the article, the punctuation, the percentage of linguistic categories in the text or or topic and sentiment classification.

The training and evaluation were done with support vector machines (SVMs) using 75% of the articles as training data with 10-fold cross validation and 25% as test set. The test set was a random sample from documents from a mixture of all the publishers which were used for training and testing.

A very similar study to ours was made by

URL	Date	No. Articles
de.reuters.com	2015 to 2018	32,984
www.pressestext.com	2008 to 2018	22,896
www.kleinezeitung.at	2018	3,361

Table 1: News agencies and newspapers

URL	Date	No. Articles
www.der-postillon.com	2008 to 2018	2,040
www.der-zeitspiegel.de	2010 to 2018	135
www.eine-zeitung.net	2010 to 2018	1,322
www.dietagespresse.at	2010 to 2018	1,130

Table 2: Satirical news sites

(Ahmed et al., 2017) with a dataset of news articles from reuters.com and a fake news dataset from PolitiFact.com collected by (Wang, 2017). They tested different supervised classification techniques and found best results for linear SVMs with an accuracy of 92%. (Granik and Mesyura, 2017) tried a simple naive Bayes classifier for fake news detection a data set of Facebook news posts and accuracy of approximately 74%.

(Pisarevskaya, 2017) used SVMs (linear/ rbf kernel) and Random Forest to classify a corpus of 174 Russian news reports (truthful and fake ones) They used frequencies of POS tags and frequencies of rhetorical relations types in texts and got F1-measure of 65% with SVMs and F1-measure of 54% with Random Forest Classifier.

3 Dataset and Data Collection

We collected datasets of news articles in German language from news agencies and newspapers via their websites (Table 1), and from satirical news sites (Table 2) for training and testing of the model.

After downloading some data cleaning, routines were performed on the data, deleting information such as marks of the publisher, author names and some footers or text-boxes - all pieces of text on which the machine learning could learn to detect the source without analyzing the content of the article. And all publishing dates were converted to a common format.

(Rubin et al., 2015) stated requirements for a fake news detection corpus based on reviewed practices and their own research. The “availability of both truthful and deceptive instances” and “verifiability of ground truth” was done by the choice of satirical news websites and normal news agencies and newspapers. For “digital textual format accessibility” we downloaded the documents to a

local SQL-Database.

“Homogeneity in lengths” was achieved in a preprocessing step by deleting articles with less than 500 characters or more than 10,000 characters in the body text. For “Homogeneity in writing matter and time” we stored the category in which the article was published and the date of publishing.

For each article we stored:

- The URL at which the article was published
- The title
- The body text
- A category that was given to the article by the publisher
- The date of publishing
- The publisher and therefore if satirical or not

In total we collected 63,868 articles in a local database, which then was used to select the data for training and testing.

4 Training and model selection

The text preprocessing pipeline was scripted in Python and used the *scikit-learn* open source machine learning package (Pedregosa et al., 2011; Géron, 2017; Raschka and Mirjalili, 2017). First the raw text of the articles and the title was imported to a Pandas¹ data frame (McKinney, 2012) and a pipeline for count-vectorization, tf*idf weighting, stop-word removal, uni-gram and bi-word tokenization was built. For classification we used Logistic Regression and Support Vector Machines.

First we split the whole dataset randomly into 80% training data and the rest for testing. We then started with standard Logistic Regression and regularisation parameter $C = 1$ and plotted the learning curves (Figure 1) with 10-fold cross validation. Since the number of fake examples in the dataset is much lower than the number of regular news, we use the F1-score as target measure. After a suitable number of training-examples the F1-score reaches a performance of approximately 85%.

¹ <https://pandas.pydata.org/>

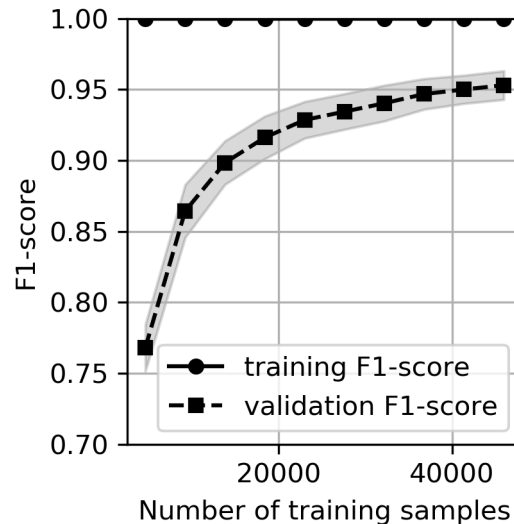


Figure 1: Learning curve of a 80% training set with Logistic Regression and $C=1$

Next we tried to optimize the regularisation parameter C to improve the performance. A grid-search shows that high values of C and therefore weak regularisation produces better results.

In the vectorization step, best results are achieved with uni-grams and bi-grams converted to lowercase. We use no classic stop-word filtering list, but terms that are used in more than 80% of the documents are filtered, and only terms that occur 20 or more times are used for the dictionary.

For Logistic Regression, the regularisation parameter C was set to 1,000 (Figure 2) and for Support Vector Machines (SVM) $C=100$. SVMs with linear kernel produces slightly better results with more computational cost than Logistic Regression. The nonlinear kernels “rbf” and “sigmoid” did not work in our setup due to overfitting. So we will use linear SVMs with the preprocessing and parameters described above for the results.

The vocabulary has a length of 125,623 terms consisting of uni-grams and bi-grams.

5 Results

With the model derived in section 4, we tested the classifier on the test set of randomly selected 20% of the corpus. On the set of 11,892 regular news and 882 satirical news, it classifies 11 regular news incorrectly and did not detect 42 of the fake news articles (Figure 3 shows the confusion matrix).

This gives a test accuracy of 0.996, a precision of 0.987, and a recall of 0.952 and so an F1-

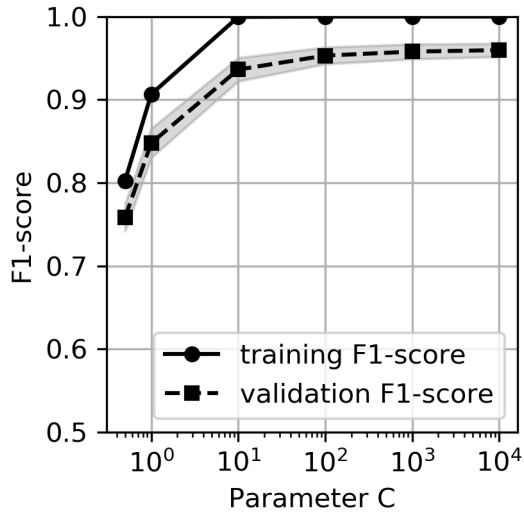


Figure 2: Validation curve for parameter C of Logistic Regression

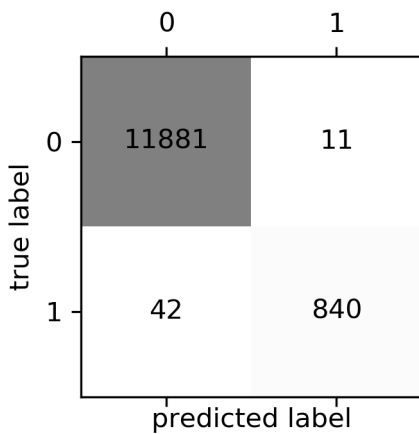


Figure 3: Confusion matrix of SVM with C=100 for a random test set

score of 0.969. This is an improvement against the F1-score of 0.87 of (Rubin et al., 2016) which was performed with 480 English news and satirical news articles and similar to the results from (Wang, 2017) with 92% accuracy but a more balanced dataset than ours. This shows the possibilities of a linear model with a lot of data without any handcrafted features.

But are we sure that the classifier detects the satire in the news? Or does it learn to identify the publisher of the article? For the training of a classifier for the “publication source” we use again a SVM with the same hyperparameter as in section 4. The six different classes of publishers were predicted fairly precisely on a 20% random test set

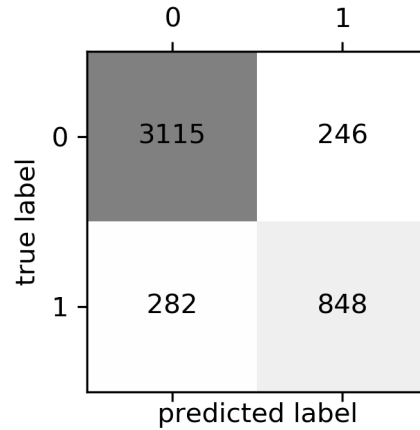


Figure 4: Confusion matrix of SVM with C=100 for “unknown” publishers

with an accuracy of 98.4%. So “knowing” articles from all publishers gives the algorithm clues for detecting satire by learning the features for detecting the publisher.

Thus, to check if the classifier can detect the satire without knowing examples from a publisher, we tried a different setup. We excluded two sources of news, one regular news site (“Kleine Zeitung”) and one satirical (“Die Tagespresse”) completely from the training dataset, and then used these articles as a test set. This provides a set of 3,361 regular and 1,130 satire test articles. The setup should be much harder for the classifier. And of course the results of a SVM classification with the same hyperparameters as configured in section 4 produces different results. The satire was detected with an accuracy of 88.2%, a precision of 77.5%, recall of 75% and therefore F1-score of 76.3%. (Figure 4 shows the confusion matrix)

Another interesting question arose while examining the data of the publisher “Presstext Austria”. There are different types of articles. Some are written by the news agency and some are paid by customers as PR text. This is marked in the online publication, so we can extract this information and use it for another classification. Is this type of news a sort of “lie”? Can the algorithm detect which article was paid?

There are 14,191 number of regular news from this publisher and 7,373 paid articles. On a 20% random test set, the two types of articles are classified with an accuracy of 99%.

6 Future Work

We think the presented methods can be used with other languages and we expect similar results as with the German corpus. Using non-linear methods for classification could improve the performance, but would need a much larger training set as we had. Our experiments with non-linear kernels in SVMs demonstrated bad results due to overfitting.

It would be interesting to use a classifier trained on satire fake news on a test set of fake news from non-satirical sources. A set of such news would be needed for this, and as indicated from the results on the untrained news sources, we expect satisfactory results only if enough articles from all sources are in the training set.

References

- Hadeer Ahmed, Issa Traore, and Sherif Saad. 2017. Detection of online fake news using n-gram analysis and machine learning techniques. In *International Conference on Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments*, pages 127–138. Springer.
- Julio Amador, Axel Oehmichen, and Miguel Molina-Solana. 2017. Characterizing political fake news in twitter by its meta-data. *arXiv preprint arXiv:1712.05999*.
- Michele Banko and Eric Brill. 2001. Scaling to very very large corpora for natural language disambiguation. In *Proceedings of the 39th annual meeting on association for computational linguistics*, pages 26–33. Association for Computational Linguistics.
- Niall J. Conroy, Victoria L. Rubin, and Yimin Chen. 2015. Automatic deception detection: Methods for finding fake news. *Proceedings of the Association for Information Science and Technology*, 52(1):1–4.
- Vanessa Wei Feng and Graeme Hirst. 2013. Detecting deceptive opinions with profile compatibility. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 338–346.
- Aurélien Géron. 2017. *Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems*. ”O’Reilly Media, Inc.”.
- M. Granik and V. Mesyura. 2017. Fake news detection using naive bayes classifier. In *2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON)*, pages 900–903.
- Alon Halevy, Peter Norvig, and Fernando Pereira. 2009. The unreasonable effectiveness of data. *IEEE Intelligent Systems*, 24(2):8–12.
- Wes McKinney. 2012. *Python for data analysis: Data wrangling with Pandas, NumPy, and IPython*. ”O’Reilly Media, Inc.”.
- Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*.
- Dina Pisarevskaya. 2017. Deception detection in news reports in the russian language: Lexics and discourse. In *Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism*, pages 74–79.
- Sebastian Raschka and Vahid Mirjalili. 2017. *Python Machine Learning*. Packt Publishing Ltd.
- Victoria Rubin, Niall Conroy, Yimin Chen, and Sarah Cornwell. 2016. Fake news or truth? using satirical cues to detect potentially misleading news. In *Proceedings of the Second Workshop on Computational Approaches to Deception Detection*, pages 7–17.
- Victoria L. Rubin, Yimin Chen, and Niall J. Conroy. 2015. Deception detection for news: Three types of fakes. *Proceedings of the Association for Information Science and Technology*, 52(1):1–4.
- Eugenio Tacchini, Gabriele Ballarin, Marco L. Della Vedova, Stefano Moret, and Luca de Alfaro. 2017. Some like it hoax: Automated fake news detection in social networks. *CoRR*, abs/1704.07506.
- William Yang Wang. 2017. ”liar, liar pants on fire”: A new benchmark dataset for fake news detection. *CoRR*, abs/1705.00648.