Weakly Supervised Learning for Fake News Detection on Twitter

Stefan Helmstetter
Data and Web Science Group
University of Mannheim
Mannheim, Germany
stefanhelmstetter@web.de

Heiko Paulheim
Data and Web Science Group
University of Mannheim
Mannheim, Germany
heiko@informatik.uni-mannheim.de

***Abstract*—The problem of automatic detection of fake news in social media, e.g., on Twitter, has recently drawn some attention. Although, from a technical perspective, it can be regarded as a straight-forward, binary classification problem, the major challenge is the collection of large enough training corpora, since manual annotation of tweets as fake or non-fake news is an expensive and tedious endeavor. In this paper, we discuss a weakly supervised approach, which automatically collects a large-scale, but very noisy training dataset comprising hundreds of thousands of tweets. During collection, we automatically label tweets by their source, i.e., trustworthy or untrustworthy *source*, and train a classifier on this dataset. We then use that classifier for a different classification target, i.e., the classification of fake and non-fake *tweets*. Although the labels are not accurate according to the new classification target (not all tweets by an untrustworthy source need to be fake news, and vice versa), we show that despite this unclean inaccurate dataset, it is possible to detect fake news with an F1 score of up to 0.9.**

***Index Terms*—Fake News, Twitter, Classification, Weak Supervision, Machine Learning**

I. Introduction

In the recent years, fake news shared on social media have become a much recognized topic [1]–[3]. Hence, methods for automatically identifying fake news is a topic which has gained some attention.

The identification of a news tweet into fake or non-fake news is a straight forward binary classification problem. Classification of tweets has been used for different use cases, most prominently sentiment analysis, but also by type (e.g., news, meme, etc.), or relevance for a given topic.

In all of those cases, the quality of the classification model strongly depends on the amount and quality of training data. Thus, gathering a suitable amount of training examples is the actually challenging task. While sentiment or topic can be more easily labeled, also by less experienced crowd workers [4], [5], labeling a news tweet as fake or non-fake news requires a lot more research, and may be a non-trivial task. For example, web sites like *Politifact*[1], which report fake news, employ a number of professional journalists for this task.

In this paper, we follow a different approach. Instead of aiming at a small-scale hand-labeled dataset with high-quality labels, we collect a large-scale dataset with low-quality labels. More precisely, we use different labels – trustworthiness of the source instead of the tweet itself – as a noisy proxy for the actual labels. This may introduce false positives (since untrustworthy sources usually spread a mix of real and fake news), as well as occasional false negatives (false information spread by trustworthy sources, e.g., by accident), although we assume that the latter case is rather unlikely. We show that the scarcity of hand-labeled data can be overcome by collecting such a dataset, which can be done almost automatically.

In other words: we build a large scale training dataset for a slightly different task, i.e., predicting the trustworthiness of a tweet's source, rather than the truth of the tweet itself. Here, we follow the notion of *weakly supervised learning*, more specifically, learning with *inaccurate supervision*, as introduced in [6]. We show that a classifier trained on that dataset (which, strictly speaking, classifies tweets as coming from a trustworthy or a non-trustworthy source) also achieves high-quality results on the task of classifying a tweet as fake or non-fake, i.e., an F1 score of up to 0.9.

II. Related Work

Although fake news in social media is an up-to-date topic, not too much research has been conducted on the automatic detection of fake news. There are, however, some works which focus on a related question, i.e., assessing the *credibility* of tweets, e.g., [7]–[10]. Most of these approaches share the same characteristics:

1) they use datasets that are fairly small,
2) they use datasets related to only a few events, and
3) they rely on crowd sourcing for acquiring ground truth.

The first characteristic may be problematic when using machine learning methods that require larger bodies of training data. The second and the third characteristic may make it difficult to update training datasets to new events, concept drift, shifts in language use on Twitter (e.g., possibly changes caused by switching from 140 to 280 characters), etc.

In contrast, the approach discussed in this paper acquires a dataset for the task of fake news detection by an automatic process, requiring only a few lists of trustworthy and non-trustworthy sources. Therefore, the process of acquiring the

[1]http://www.politifact.com/

IEEE/ACM ASONAM 2018, August 28-31, 2018, Barcelona, Spain
978-1-5386-6051-5/18/$31.00 © 2018 IEEE

dataset can be repeated, gathering a large-scale, up-to-date dataset at any time.

III. Datasets

For training a machine learning model to detect fake news, we collect a large-scale dataset from Twitter. Furthermore, we collect a small, hand-labeled dataset for evaluation purposes.

A. Large-scale Training Dataset

We create our training dataset by first collecting trustworthy and untrustworthy sources. Then, for each of the sources, we collect tweets using the Twitter API. Each tweet from a trustworthy source is labeled as real news, each tweet from an untrustworthy source is labeled as fake news.

While this labeling can be done automatically at large scale, it is far from perfect. Most untrustworthy sources spread a mix of fake and real news. The reverse (i.e., a trustworthy source spreading fake news, e.g., by accident) may also occur, but we assume that this case is far less likely.

For collecting fake news sources, we use lists from different Web pages[2,3,4,5,6,7,8], as well as the Web catalogue *opensources*[9]. In total, we collected 65 sources of fake news.

For collecting trustworthy news sources, we used a copy of the recently shut down DMOZ catalog[10], as well as those news sites listed as trustworthy in *opensources*, and filtered the sites to those which feature an active Twitter channel. In order to arrive at a balanced dataset, we collected 46 trustworthy news sites.[11]

In the next step, we used the Twitter API[12] to retrieve tweets for the sources. The dataset was collected between February and June 2017. Since the Twitter API only returns the most recent 3,200 tweets for each account[13], the majority of tweets in our dataset is from the year 2017 – e.g., for an active twitter account with 20 tweets per day, that limitation Twitter API allows us retrieve tweets for the past 160 days.

In total, we collected 401,414 examples, out of which 110,787 (27.6%) are labeled as fake news (i.e., they come from fake news sources), while 290,627 (24,4%) are labeled as real news (i.e., they come from trustworthy sources). Fig. 1 shows the distribution of tweets by their tweet time.

As discussed above, we expect the real news class to contain only a neglectable amount of noise, but we inspected a sample

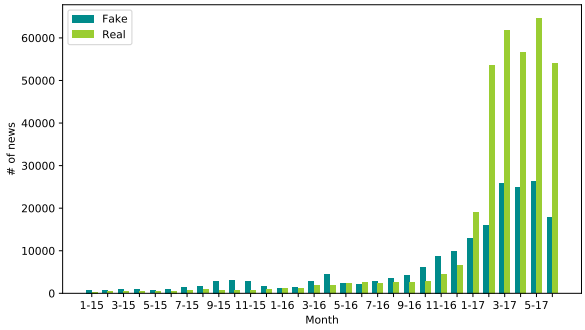

Fig. 1. Distribution of tweets labeled as real and fake news in the training dataset

of the fake news class manually. We found that the fake news class is comprised of only 15% fake news tweets, 40% real news tweets, whereas the rest is either no news or undecidable. However, since the sample contains both real and fake news tweets from the same period of time, we can assume that for real news, those will also appear in the class labeled as non-fake, and since the real news class is larger by a factor of three, the classifier will more likely label them as real news.

B. Small-scale Evaluation Dataset

For creating a hand-labeled gold standard, we used 116 tweets from the *politifact* web site that were classified as fake news by expert journalists (see above). Those were used as positive examples for fake news tweets. Note that the sources of those tweets are not sources that have been used in the training set. For generating negative examples, awe picked those 116 tweets which were the closest to the fake news tweets in the positive class according to TF-IDF and cosine similarity, and removed those 116 tweets from the training dataset before training our classification models.

C. Evaluation Scenarios

We consider two different evaluation scenarios. Scenario 1 only considers the tweet as such, whereas scenario 2 also includes information about the user account from which the tweet was sent. The rationale is that while including as much information as possible will likely improve the results, we also want to be able to apply the approach in a setting where a tweet is sent from a new, unknown user account which neither known to be credible or non-credible.

IV. Approach

We model the problem as a two-class classification problem. Our approach is trained on the large-scale, noisy dataset, using different machine learning algorithms. All of those methods expect the representation of a tweet as a vector of features. Therefore, we use different methods of extracting features from a tweet. We consider five different groups of features: user-level features, tweet-level features, text features, topic features, and sentiment features. For the feature engineering,

[2]http://mashable.com/2012/04/20/twitter-parodies-news/#IdNx6sIG.Zqm

[3]https://www.dailydot.com/layer8/fake-news-sites-list-facebook/

[4]https://www.cbsnews.com/pictures/dont-get-fooled-by-these-fake-news-sites/

[5]http://fakenewswatch.com/

[6]https://www.snopes.com/2016/01/14/fake-news-sites/

[7]https://www.thoughtco.com/guide-to-fake-news-websites-3298824

[8]https://newrepublic.com/article/118013/satire-news-websites-are-cashing-gullible-outraged-readers

[9]http://www.opensources.co/

[10]http://dmoztools.net/

[11]That number is incidentally chosen lower than that of fake news sources, since we could collect more tweets from the trustworthy sites.

[12]https://developer.twitter.com/en/docs

[13]https://developer.twitter.com/en/docs/tweets/timelines/api-reference/get-statuses-user_timeline.html

TABLE I
RESULTS SUMMARY, DEPICTING THE BEST F1 SCORE ACHIEVED FOR EACH TASK (FAKE NEWS SOURCE AND FAKE NEWS TWEET DETECTION), AS WELL AS FOR EACH FEATURE GROUP (TWEET LEVEL FEATURES ONLY AND TWEET AND SOURCE FEATURES)

	Identifying fake news...	
	Sources	Tweets
Tweet features only	0.7758	0.7699
Tweet and source features	0.9360	0.8996

we draw from previous works that extract features from tweets for various purposes [7], [9]–[15].

A. User-level Features

For the user, we first collect all features that the Twitter API[14] directly returns for a user, e.g., the number of followers. Furthermore, we use the API to create additional statistics modeling the user's behavior on Twitter, e.g., the frequency of tweets or the ratio of retweets. In total, we create 53 user-level features.

B. Tweet-level Features

For tweet-level features, we again use the Twitter API to first collect all information directly available (e.g., number of retweets), and add meta information (e.g., weekday and time) as well as statistical information on the contents (e.g., word count, ratio of question and exclamation marks). In total, we create 69 tweet-level features.

In order to make the approach applicable in real-time scenarios and be able to immediately classify new tweets, we remove time-dependent attributes (i.e., number of retweets and number of favorites).

C. Text Features

The features above do not consider the actual text of the tweet. For representing the textual contents of the tweet, we explored two alternatives: a bag of words (BOW) model using TF-IDF vectors, and a neural Doc2vec model [16] trained on the corpus. For the latter, we use *gensim*[15], and train models with 100, 200, and 300 dimensions, both with DM and DBOW.

D. Topic Features

Since the topic of a tweet may have a direct influence on the fake news classification, as some topics are likely more prone to fake news than others, we also apply topic modeling for creating features from the tweets. We trained both a Latent Dirichlet Allocation model (LDA) [17] on the whole dataset, varying the number of topics between 10 and 200 in steps of 10, as well as a Hierarchical Dirichlet Process (HDP) model [18], which does not require the selection of a number of topics.

[14]https://developer.twitter.com/en/docs/api-reference-index

[15]https://radimrehurek.com/gensim/

E. Sentiment Features

In addition to content representation, we used *SentiWordNet* [19] to compute the polarity of tweets in terms of ratio of positive, negative, and neutral words. Furthermore, we use the *TextBlob*[16] library to compute the subjectivity of a tweet, as introduced in [20]. Polarity and subjectivity scores account for eight additional features.

F. Feature Scaling and Selection

The resulting feature set combining all of the above strategies is fairly large, hence, we expect performance gains from dimensionality reduction or feature selection. We explored three different options here: setting a minimum variance threshold, recursive elimination using the Gini index as an importance criterion, and recursive elimination using mutual information [21].

G. Learning Algorithms and Parameter Optimization

As learning algorithms, we use Naive Bayes, Decision Trees, Support Vector Machines (SVM), and Neural Networks as basic classifiers. Moreover, we use two ensemble methods known to usually work well, i.e., Random Forest and XGBoost, using parameter optimization on all of those approaches (i.e., C and γ for SVM, number of hidden neurons, learning rate, activation function and regularization penalty for neural network, number of trees for Random Forest and XGBoost).

V. EVALUATION

We evaluate our approach in different settings. First, we perform cross-validation on our noisy training set; second, and more importantly, we train models on the training set and validate them against a manually created gold standard.[17] Moreover, we evaluate two variants, i.e., including and excluding user features. The rationale of the latter is to simulate two use cases: assessing a tweet from a known user account, and assessing a tweet from a new user account.

Since the original training set was labeled by source, not by tweet, the first setting evaluates how well the approach performs on the task of identifying fake news *sources*, whereas the second setting evaluates how well the approach performs on the task of identifying fake news *tweets* – which was the overall goal of this work. It is important to note that the fake news tweets come from sources that have not been used in the training dataset. Table I summarizes the best achieved results for each of the four settings.

A. Setting 1: Cross-validation on Training Dataset

To analyze the capabilities of the predictive models trained, we first perform cross-validation on the training dataset. Due to the noisy nature of the dataset, the test dataset also carries noisy labels with the same characteristics as the training dataset, and thus, we expect the results to *over-estimate* the actual performance on a correctly labeled training dataset.

[16]http://textblob.readthedocs.io/en/dev/index.html

[17]Both datasets are available at http://dws.informatik.uni-mannheim.de/en/research/twitter-fake-news-detection

Hence, the results depict an *upper bound* of our proposed weak supervision method. Not much surprisingly, adding information on the user clearly improves the results. We can observe that the best results are achieved using XGBoost, leading to an F1 score on the fake news class of 0.78 and 0.94, respectively.

B. Setting 2: Validation against Gold Standard

As discussed above, the more important setting validates the approach using a manually annotated gold standard. Since that gold standard dataset was collected independently from the training dataset, and is never used for training, feature selection, or parameter optimization, we can safely state that our approach is not overfit to that dataset.

For the feature sets, feature selection methods, and parameter settings, we used the setups that worked best in the cross validation settings. In contrast to the results in cross validation, the neural network learner performs best in that scenario.

VI. Conclusion and Outlook

In this work, we have shown a practical approach for treating the identification of fake news on Twitter as a binary machine learning problem. While that translation to a machine learning problem is rather straight forward, the main challenge is to gather a training dataset of suitable size. Here, we have shown that, instead of creating a small, but accurate hand-labeled dataset, using a large-scale dataset with inaccurate labels yields very good results as well.

We have shown that our approach yields very good results, achieving an F1 score of 0.77 when only taking into account a tweet as such, and up to 0.9 when also including information about the user account. It is particularly remarkable that the results are not much worse than those achieved for classifying trustworthy and untrustworthy *sources* (which is actually reflected in the labels for the tweets): with tweet features only, the best F1 score achieved is 0.78, with both tweet and user features, the best F1 score 0.94. In summary, we have shown that the problem of acquiring large-scale training datasets for fake news classification can be circumvented when accepting a certain amount of label noise, which still can yield well performing classifiers.

References

[1] J. Ratkiewicz, M. Conover, M. R. Meiss, B. Gonçalves, A. Flammini, and F. Menczer, "Detecting and tracking political abuse in social media." *ICWSM*, vol. 11, pp. 297–304, 2011.

[2] E. Ferrara, O. Varol, C. Davis, F. Menczer, and A. Flammini, "The rise of social bots," *Communications of the ACM*, vol. 59, no. 7, pp. 96–104, 2016.

[3] K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu, "Fake news detection on social media: A data mining perspective," *ACM SIGKDD Explorations Newsletter*, vol. 19, no. 1, pp. 22–36, 2017.

[4] S. Rosenthal, P. Nakov, S. Kiritchenko, S. Mohammad, A. Ritter, and V. Stoyanov, "Semeval-2015 task 10: Sentiment analysis in twitter." in *SemEval at NAACL-HLT*, 2015, pp. 451–463.

[5] S.-H. Yang, A. Kolcz, A. Schlaikjer, and P. Gupta, "Large-scale high-precision topic modeling on twitter," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2014, pp. 1907–1916.

[6] Z.-H. Zhou, "A brief introduction to weakly supervised learning," *National Science Review*, 2017.

[7] A. Gupta and P. Kumaraguru, "Credibility ranking of tweets during high impact events," in *Proceedings of the 1st Workshop on Privacy and Security in Online Social Media*, ser. PSOSM '12. New York, NY, USA: ACM, 2012, pp. 2:2–2:8. [Online]. Available: http://doi.acm.org/10.1145/2185354.2185356

[8] S. Mohd Shariff, X. Zhang, and M. Sanderson, "User perception of information credibility of news on twitter," in *Proceedings of the 36th European Conference on IR Research on Advances in Information Retrieval - Volume 8416*, ser. ECIR 2014. New York, NY, USA: Springer-Verlag New York, Inc., 2014, pp. 513–518. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-06028-6_50

[9] S. Sikdar, S. Adali, M. Amin, T. Abdelzaher, K. Chan, J. H. Cho, B. Kang, and J. O'Donovan, "Finding true and credible information on twitter," in *17th International Conference on Information Fusion (FUSION)*, July 2014, pp. 1–8.

[10] C. Castillo, M. Mendoza, and B. Poblete, "Information credibility on twitter," in *Proceedings of the 20th International Conference on World Wide Web*, ser. WWW '11. New York, NY, USA: ACM, 2011, pp. 675–684. [Online]. Available: http://doi.acm.org/10.1145/1963405.1963500

[11] A. E. Azab, A. M. Idrees, M. A. Mahmoud, and H. Hefny, "Fake account detection in twitter based on minimum weighted feature set," *World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering*, vol. 10, p. 13–18, 2015, n/a.

[12] F. Benevenuto, G. Magno, T. Rodrigues, and V. Almeida, "Detecting spammers on twitter," in *In Collaboration, Electronic messaging, Anti-Abuse and Spam Conference (CEAS)*, 2010.

[13] A. Deshwal and S. K. Sharma, "Twitter sentiment analysis using various classification algorithms," in *2016 5th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, Sept 2016, pp. 251–257.

[14] S. Gurajala, J. S. White, B. Hudson, and J. N. Matthews, "Fake twitter accounts: Profile characteristics obtained using an activity-based pattern detection approach," in *Proceedings of the 2015 International Conference on Social Media & Society*, ser. SMSociety '15. New York, NY, USA: ACM, 2015, pp. 9:1–9:7. [Online]. Available: http://doi.acm.org/10.1145/2789187.2789206

[15] G. Stringhini, C. Kruegel, and G. Vigna, "Detecting spammers on social networks," in *Proceedings of the 26th Annual Computer Security Applications Conference*, ser. ACSAC '10. New York, NY, USA: ACM, 2010, pp. 1–9. [Online]. Available: http://doi.acm.org/10.1145/1920261.1920263

[16] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proceedings of the 31st International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, E. P. Xing and T. Jebara, Eds., vol. 32, no. 2. Bejing, China: PMLR, 22–24 Jun 2014, pp. 1188–1196. [Online]. Available: http://proceedings.mlr.press/v32/le14.html

[17] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003. [Online]. Available: http://dl.acm.org/citation.cfm?id=944919.944937

[18] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Hierarchical dirichlet processes," *Journal of the American Statistical Association*, vol. 101, no. 476, pp. 1566–1581, 2006.

[19] S. Baccianella, A. Esuli, and F. Sebastiani, "Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining," in *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, N. C. C. Chair), K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, and D. Tapias, Eds. Valletta, Malta: European Language Resources Association (ELRA), may 2010.

[20] T. D. Smedt and W. Daelemans, ""vreselijk mooi!" (terribly beautiful): A subjectivity lexicon for dutch adjectives." in *LREC*, N. Calzolari, K. Choukri, T. Declerck, M. U. Dogan, B. Maegaard, J. Mariani, J. Odijk, and S. Piperidis, Eds. European Language Resources Association (ELRA), 2012, pp. 3568–3572. [Online]. Available: http://dblp.uni-trier.de/db/conf/lrec/lrec2012.html#SmedtD12

[21] L. C. Molina, L. Belanche, and À. Nebot, "Feature selection algorithms: A survey and experimental evaluation," in *Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on*. IEEE, 2002, pp. 306–313.