



# Fake News Detection by Weakly Supervised Learning Based on Content Features

Özlem Özgöbek<sup>1(✉)</sup>, Benjamin Kille<sup>1</sup>, Anja Rosvold From<sup>2</sup>,  
and Ingvild Unander Netland<sup>3</sup>

<sup>1</sup> Norwegian University of Science and Technology, Trondheim, Norway  
{ozlem.ozgobek,benjamin.u.kille}@ntnu.no

<sup>2</sup> Bredvid AS, Oslo, Norway

<sup>3</sup> Kantega AS, Oslo, Norway

**Abstract.** Fake news, defined as the publication of false information, either unintentional or with the intent to deceive or harm, is one of the important issues that affects today's digital society significantly. All around the world, journalists and fact checking organizations are trying to fight this problem manually. However, fighting fake news is a time-sensitive task. Once leaked, fake news spreads fast and its impact on society increases. Because of the complex and dynamic nature of news, applying artificial intelligence methods to address the automatic detection of fake news is a challenging task. This work explores the use of weak supervised learning for fake news detection by using only the content of news articles. This is particularly important when the contextual information is not available or difficult to obtain quickly. To our knowledge, this is the first work which uses a content-based approach in weak supervised learning without the use of any contextual information for fake news detection. We propose an architecture that generates weak labels. We explore the effect of using weak labels for fake news detection with five different machine learning models. We demonstrate that weakly supervised learning is an effective approach to the automated detection of fake news in the absence of high quality labels.

**Keywords:** Fake news detection · Disinformation · Weakly supervised learning · Content features

## 1 Introduction

The spread of fake news is not a new problem. However, with the advancement of the internet and social media, it has become a growing problem [32]. Fast and uncontrolled spread of fake news can affect society in many ways, including ideological polarization [24] and psychological bias [28]. During the Covid-19 pandemic, we have experienced how problematic the situation can be globally [7, 20]. Despite the ongoing efforts for developing automated fake news detection

systems, most of the work is still being done by professional journalists in fact checking organizations all around the world<sup>1</sup>.

As machine learning is one of the promising techniques for automated fake news detection, one of the obstacles is the amount of accurately labeled training data that are available. Unfortunately, there are very few labeled datasets of sufficient size and quality for supervised learning in this domain. This is due to the scarce resources for manual fact checking and labeling efforts. In addition, because of the new events are introduced continuously, the content and topic of news articles are time-dependent and varied [4]. Weakly supervised learning, a new machine learning paradigm, has been developed to work with low-quality labels called weak labels [19].

In this paper, we present a fake news detection system that uses weakly supervised learning based only on content features. We have chosen to work with full news articles instead of social media posts or shares. Even though the social media is seen as the primary source of the spread of fake news, recent research [27] points out the importance of the coverage of fake news in mainstream media.

Weakly supervised systems can utilize content and contextual features. Previous work using this approach for fake news detection has given promising results [10]. However, the contextual features used in these efforts (e.g. likes, comments, shares) are time-dependent (they change over time), take time to accumulate, and are unavailable for some articles. Therefore, our approach is solely based on the content-based features extracted from the title and content of the articles.

To the best of our knowledge, this is the first work that uses weak supervision for fake news detection by using only content features. Our contributions are three-fold: We introduce a probabilistic weak labeling system that relies only on content features. We collect and present a test dataset from a set of fact-checking organizations including Snopes<sup>2</sup> and PolitiFact<sup>3</sup>. The dataset has been made publicly available on Github<sup>4</sup>. We apply five machine learning classifiers for fake news detection with and without the weak labels to investigate the efficiency of using weak supervised learning with content features.

The rest of the paper is organized as follows: Sect. 2 discusses the state of the art and existing work on weakly supervised methods for fake news detection. Section 3 presents the dataset for the experiments. Section 4 outlines the proposed architecture and experimental design, and presents the findings. Section 5 concludes and gives an outlook to future research directions.

## 2 Related Work

Even though the fact checking tasks still rely on the professional journalists, the efforts of developing automated fake news detection systems has been in focus of

<sup>1</sup> <https://reporterslab.org/fact-checking/>.

<sup>2</sup> [www.snopes.com](http://www.snopes.com).

<sup>3</sup> [www.politifact.com](http://www.politifact.com).

<sup>4</sup> <https://github.com/piingz/fake-news-detection-test-set>.

the researchers for the last couple of years. Within these research there is a wide variety of approaches. Crowd-sourcing has been proposed to obtain the labels for fake news [6, 17]. However, human annotators can process a limited number of articles. [6] had 90 articles annotated, whereas [17] acquired labels for 240 articles. Crowd sourcing suffers from high costs and doubts in annotations’ quality. [35] argues that with a large enough population of fact-checkers, indicating the credibility of articles remains feasible. How to attract and motivate a large enough population remains to be seen. Research on fully automated methods follows different approaches such as content-based, user-based, network-based, and hybrid methods which use the combination of other methods [15]. Content-based methods focus on the analysis of text and non-text content, such as video or sound. For instance, Shrestha et al. [21] combine textual features, sentiment, writing style and psycho-linguistics to identify fake news. User-based methods look at user behaviour and comments to identify fake news [26]. Wang et al. [31] combines a weakly supervised approach with user reports. Network-based methods monitor network activity, which can help to detect bots and investigate the spread patterns. Conversely, Shu et al. [23] reports that humans spread more fake news than bots. In this case, finding out if a user is a real human may help to increase the clues a system collects. Moreover, [23] shows that fake news is more likely to be spread by fake accounts. The computational methods used in the automated detection of fake news is varied. Castelo et al. [4] proposes a topic-agnostic approach to the classification of fake news by using web-markup in addition to LIWC (Linguistic Inquiry and Word Count), and stylistic features. With this approach, they focus on identifying the non-credible web pages spreading fake news instead of detecting individual fake news articles.

Weakly supervised learning (mainly together with contextual features) has been used for fake news detection by many researchers. Helmstetter and Paulheim [10] apply weakly supervised learning to microblogs for detecting fake news in social media and obtain an accuracy of approximately ninety percent. Wang et al. [30] uses reinforcement learning for fake news detection with the use of crowd-sourced labels. Yuan et al. [34] combine weakly supervised learning with a structure-aware multi-head attention network to identify fake news. Weakly supervised learning has been used with content features for tasks such as learning discourse structures in dialogues [2] and building a text classifier in combination with transfer learning [25].

### 3 Dataset

To choose the best suited dataset for this task, we have reviewed 14 datasets. Table 1 presents an overview of these datasets. Our evaluation considered four properties: size, features, class balance, and labeling method. As a result, we have decided to use the NELA-GT-2019 [9] dataset<sup>5</sup>. The chosen dataset has a large amount of data for all classes. Thus, it supports creating class-balanced subsets. The dataset’s features include title and content. The documentation

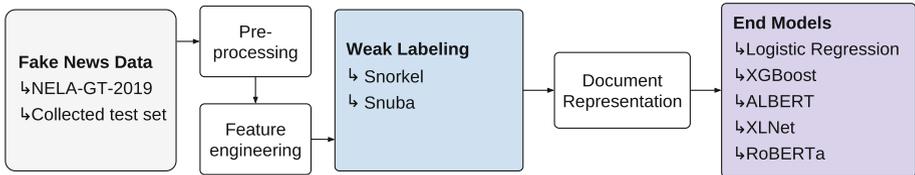
<sup>5</sup> At the time this work started NELA-GT-2020 dataset was unavailable.

of the dataset is excellent. NELA-GT-2019 comprises 1.12M news articles from 260 mainstream and alternative news sources. It has been collected between 01 January 2019 and 31 December 2019. There are four different labels: *reliable*, *mixed*, *unreliable*, and *unknown*. In this work, we consider articles labelled *reliable* as credible news and *unreliable* as fake news. We discard the labels *mixed* and *unknown*. The labels in the dataset have been assigned based on the credibility of news sources which does not guarantee the correctness of information itself.

**Dataset Collection.** For more realistic assessment of the developed models, we have collected an independent dataset which consists of manually fact-checked news articles. So the labels of the news articles in this test set are not based only on the sources they were published on, but on the decision of professional fact checkers. In addition, we payed attention to the publishing date of articles to avoid testing our models on the same news items as were included in the training dataset. Therefore the articles collected for this dataset were published in a different period than NELA-GT-2019. During the collection of this dataset we have used entries from FakeNewsNet<sup>6</sup> [22] and MisInfoText [1] datasets as well as manual collection of articles from Snopes fact-checking archives<sup>7</sup>. The collected dataset includes 434 news articles where half of them is fake and the other half is real news articles. This dataset is available on Github<sup>8</sup>.

## 4 Architecture, Experiments and Results

The proposed system consists of two main components: The weak labeling system and the classification models that use weakly supervised learning. For each of these main components we ran a series of experiments in order to find the best performing models. Then we combine these in our proposed architecture. Figure 1 shows the overall architecture of the proposed system.



**Fig. 1.** Overall system architecture

First we apply pre-processing and feature engineering to the raw data. Then the output is passed to the weak labeling system which generates weak

<sup>6</sup> <https://github.com/KaiDMML/FakeNewsNet>.

<sup>7</sup> <https://www.snopes.com/fact-check>.

<sup>8</sup> <https://github.com/piingz/fake-news-detection-test-set>.

**Table 1.** Fake news datasets reviewed in this work.

Dataset	Type of content	Size
Fakeddit	Reddit posts incl. images, social engagement	1,063,106
FakeNewsCorpus	Articles	9,408,908
FakeNewsNet	Articles, social engagement	1,056
FA-KES	Syrian war articles	804
FEVER	Short statements	185,445
FNID	Statements	15,212
Getting real about fake news	Articles, social engagement	12,999
LIAR	Short statements	12,836
MisInfoText B	Articles	1,380
MisInfoText S	Articles	312
NELA-GT-2018	Articles	713,534
NELA-GT-2019	Articles	1,118,821
NELA-GT-2020	Articles	1,779,127
Twitter	Tweets, user info., social engagement	401,414

labels. After the application of document representation, weakly labelled data is passed to the end model. We have experimented with *Snorkel* and *Snuba*, two weak labeling frameworks, and five classifiers: Logistic Regression, XGBoost [5], ALBERT [12], XLNET [33], and RoBERTa [13].

In the following sections, all these steps are explained in detail and the results from various experiments are presented.

#### 4.1 Data Pre-processing

The pre-processing steps includes applying natural language processing (NLP) techniques such as normalization, stop word removal, and tokenization to the news text. More specifically we have normalized the text, removed punctuation, digits and stop-words, and tokenized into words, bigrams, trigrams and sentences. We used the NLTK word tokenizer<sup>9</sup>, NLTK sentence tokenizer<sup>10</sup>, NLTK part-of-speech tagger<sup>11</sup>, WordNetLemmatizer<sup>12</sup> and Python’s built-in lowercasing function. Each step has been applied to both the title and the body of the articles.

<sup>9</sup> [https://www.nltk.org/api/nltk.tokenize.html#nltk.tokenize.word\\_tokenize](https://www.nltk.org/api/nltk.tokenize.html#nltk.tokenize.word_tokenize).

<sup>10</sup> [https://www.nltk.org/api/nltk.tokenize.html#nltk.tokenize.sent\\_tokenize](https://www.nltk.org/api/nltk.tokenize.html#nltk.tokenize.sent_tokenize).

<sup>11</sup> [https://www.nltk.org/\\_modules/nltk/tag.html#pos\\_tag](https://www.nltk.org/_modules/nltk/tag.html#pos_tag).

<sup>12</sup> [https://www.nltk.org/\\_modules/nltk/stem/wordnet.html](https://www.nltk.org/_modules/nltk/stem/wordnet.html).

## 4.2 Feature Engineering

After the pre-processing step, we have determined four types of relevant features based on the literature: stylistic features, complexity features, POS-tagging features, and sentiment features [3, 11, 16, 18].

Stylistic features include author’s writing style such as the use of exclamation marks and uppercase words; complexity features include implicit features of the text such as type-token ratio and words per sentence; POS-tagging features include all POS-tag related features such as the presence of verbs, nouns, and adjectives; finally, sentiment features include the sentiment scores of the text such as the scores for subjectivity, positiveness, and negativeness.

In total, we have used 68 extracted features such as “*ratio of stop words*”, “*number of quote marks*”, “*ratio of nouns per word*” and “*document negative score based on sentences*”. A complete list of these features can be found in [8]. For each of these features, the resulting numerical values such as the sentence count, word count etc. are then passed as an input to the weak labeling system.

## 4.3 Weak Labeling System

The first main component of our architecture focuses on the generation of weak labels for fake news detection. For this, we consider two weak labeling frameworks: Snorkel<sup>13</sup> and Snuba<sup>14</sup>. We have run a set of experiments to compare these two frameworks in the context of fake news detection. Figure 3 shows the overall pipeline of the experiments to evaluate the weak supervised fake news detection.

During our experiments with Snorkel, in order to enhance the performance, we have developed three components for the weak labeling system: *Automatic threshold search*, *automatic labeling function (LF) generation*, and *labeling function (LF) selection*. In order to create this weak labeling system we have used a small portion of the labeled data we have which is not included in the evaluation of the end models to prevent the data leakage. Figure 2 shows the overall pipeline.

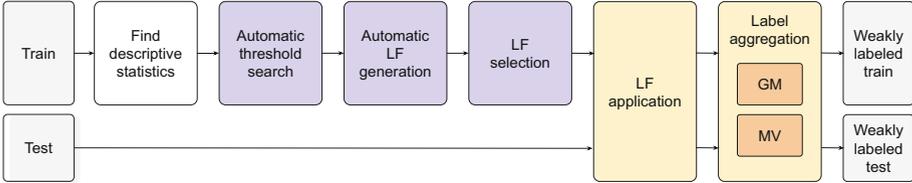
*Automatic threshold search* takes the instances described with descriptive statistics (such as title word count) as input and selects best feature values (thresholds) that define an instance being fake or real. *Automatic LF generation* component handles the automatic generation of labeling functions in Snorkel. The labels are assigned automatically based on the thresholds defined in the previous step by checking if a feature value of an instance is above or below the threshold. It is also important to find values that cover a large portion of the data set since the higher the coverage the higher the amount of labels assigned is. *LF selection* component handles the possible extremely noisy labels by selecting a portion of LFs. To do that, we evaluated three sets of LFs by using Snorkel’s generative model and majority vote approaches: All LFs (*All*), LFs with

<sup>13</sup> <https://www.snorkel.org>.

<sup>14</sup> <https://github.com/HazyResearch/reef>.

an individual accuracy above 65% ( $Acc > 65\%$ , this value has been chosen as a result of separate experiments) and top 25 LFs based on their accuracy (*Top 25*).

As a result of our experiments, we have found that the best performing model was  $Acc > 65\%$  with an accuracy of 0.710 and coverage of 0.860. More details of these components can be found in [8].



**Fig. 2.** The pipeline of the automatic weak labeling system in Snorkel. The purple color indicates the components developed in this work. The white color indicates preliminary processing, yellow color indicates the processes handled by Snorkel and the gray color indicates the input and output of the system. (Color figure online)

Snuba framework has been proposed by [29] and it creates heuristics that assign probabilistic labels to instances. Compared to Snorkel, it generates less noisy labels and provides more diversity of instances labeled. In this work we have implemented a weak labeling system using Snuba and tested it with tree types of heuristics, namely decision trees, logistic regression and k-nearest neighbor (KNN). Following the findings of [29] which suggested that the maximum cardinality below four would be sufficient for most real-world tasks, we have experimented with the values below four. Due to the hardware limitations we could not get any results from KNN max cardinality three. The results from these experiments are shown in Table 2. Based on these results we have chosen the best method based on accuracy and coverage. Note that the portion of the data set we have used for these experiments does not contain the data from the weak label generation part to prevent data leakage.

As a result of our experiments with Snorkel and Snuba, we found that Snuba achieves an accuracy of 0.765 and coverage of 0.902, outperforming Snorkel both in terms of accuracy and coverage. We explain this with Snuba’s heuristics being more complex than Snorkel and taking the heuristic’s diversity into account. Therefore we use Snuba as our weak labeling component. Then, we run the best performing weak labeling system on the *manually labeled test set* to assure that the classifiers would perform better than the weak labeling system so that it is reasonable to train end models. We observed that *Snuba, DT, 3* achieved an accuracy of 0.646,  $F_1$  score of 0.668 and coverage of 0.956.

#### 4.4 Document Representation

Classifiers require the input to come in the form of numerical vectors. We experiment with two different methods to obtain such vectors from the output of

**Table 2.** The results from the experiments with different types of heuristics of Snuba.

Model	Max cardinality	Accuracy	F1 score	Coverage
<i>DT</i>	1	0.836	0.911	0.077
	2	0.753	0.769	0.873
	<b>3</b>	<b>0.765</b>	0.765	0.902
<i>LR</i>	1	0.774	0.845	0.221
	2	0.766	0.816	0.384
	3	0.760	0.777	0.551
<i>k-NN</i>	1	0.610	0.483	1.000
	2	0.650	0.584	1.000
	3	N/A	N/A	N/A

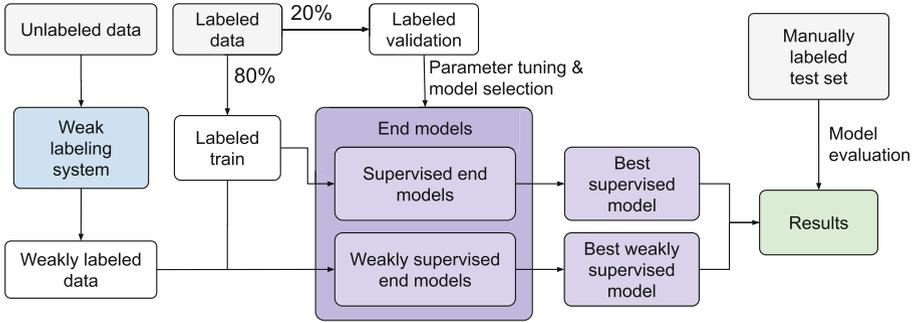
the weak labeling system: TF-IDF and BERT-specific. BERT-based models are designed to deal with raw text which reduces the processing to two simple steps. First, we merge the articles’ title and content. Second, we trim the text to conform to the maximum length of token supported by the models. For Logistic regression and XGBoost, we used TF-IDF with an array size of 6000.

#### 4.5 Weakly Supervised Learning

We have trained five models—Logistic Regression, XGBoost, ALBERT, XLNet, and RoBERTa—to determine the best performing classification model for weakly supervised learning in this domain. We have chosen these models based on their previous success for fake news classification [14]. We have also trained the same models as supervised end models for the comparison. Table 3 shows the size of datasets used in this experiment. As it is shown in Fig. 3, both weakly supervised models and supervised models take a portion of the labeled data as input. The weakly supervised models take the weakly labeled data from the weak labeling system as an additional input.

Table 4 presents the results from our experiments with these models using weak labels. Results show that RoBERTa outperforms the four other classifiers, reaching to an accuracy of 0.753, F1 score of 0.779 for supervised and an accuracy of 0.779, F1 score of 0.798 for weakly supervised method on the manually created test set. The second best performing model in this setting is the XLNet with an accuracy of 0.719, F1 score of 0.742 for supervised and an accuracy of 0.733, F1 score of 0.752 for the weakly supervised method. Results of these experiments show that weakly supervised method performs slightly better than the supervised approach. These results also suggest that the combination of weak labeling system and classifier perform better than the weak labeling system alone as it was explained in Sect. 4.3.

In order to understand how the amount of weak labels introduced affects the weakly supervised model, we have experimented with three different ratios of weak labels. Based on the result of the previous experiment, we have used



**Fig. 3.** Experimental pipeline for the end models.

RoBERTa for both weakly supervised and supervised models. First, we have trained our models with all the weak labeled instances (approx. 170K), and then 50K and 25K weak labeled instances respectively, where the total number of instances in the dataset for this set of experiments is approximately 201K. Table 5 shows the results from these experiments. The results of these experiments indicate that the supervised model performs better than the weakly supervised method. As we keep adding more weak labeled data the performance decreases. The weak labeled instances are selected by confidence. This suggests that high-confidence labels contribute best to the detection, whereas low-confidence labels spoil the performance. However, results also show that the difference between these models, (especially the supervised, weak 25K and weak 50K) is marginal. Given that we have tested our system with only one test set, we do not know how the results would change for other datasets. Additionally, our test set is relatively small compared to the training set (see Table 3). We expect weakly supervised models to perform better in conditions where the test set is similar or larger in size as the training data set. We believe that weakly supervised learning for fake news detection is a promising method and should be explored further. Also more research is required to verify the effect of weakly labeled data for fake news detection.

**Table 3.** Size of datasets used.

	# of samples	Origin dataset
Labeled training set	1 380	NELA-19
Unlabeled training set	201 604	NELA-19
Weakly labeled training set	5 520	NELA-19
Labeled validation set	345	NELA-19
Manually labeled test set	434	Manual

**Table 4.** Comparison of classifiers. For each of the five classifiers, we list the scores on the *manually created test set*, as well as the difference between the usage of weakly supervised labels. The rows refer to Logistic Regression (LR), XGBoost (XG), ALBERT (AL), XLNet (XL), and RoBERTa (Ro).

	Accuracy			$F_1$		
	Supervised	Weak	$\Delta$	Supervised	Weak	$\Delta$
LR	0.624	0.641	-0.017	0.630	0.653	-0.023
XG	0.578	0.618	-0.040	0.592	0.623	-0.031
AL	0.696	0.696	0.000	0.726	0.717	0.009
XL	0.719	0.733	-0.014	0.742	0.752	-0.010
Ro	<b>0.753</b>	<b>0.779</b>	-0.026	<b>0.779</b>	<b>0.798</b>	-0.019

**Table 5.** The comparison of supervised and weakly supervised models with different ratios of weak labels.

	Validation		Test set	
	Accuracy	F1 score	Accuracy	F1 score
<i>Weak<sub>all</sub></i>	0.800	0.808	0.671	0.721
<i>Weak<sub>50k</sub></i>	0.900	0.901	0.753	0.778
<i>Weak<sub>25k</sub></i>	0.942	0.942	0.781	0.801
<i>Supervised</i>	<b>0.959</b>	<b>0.959</b>	<b>0.793</b>	<b>0.813</b>

## 5 Conclusions and Future Work

Automation will remain necessary to combat fake news as long as fact-checkers remain a scarce resource. Fake news classifiers rely on accurate labels. This work proposed and explored the use of weakly supervised learning that relies only on the content features. Our observations on the performance of different weak labeling frameworks suggest that Snuba performs better than Snorkel for this task. As a result of our experiments with five different classifiers, RoBERTa outperformed the other four classifiers both in supervised and weakly supervised tasks. We tested the weak labels’ utility for fake news detection with help of the NELA-GT-2019 data set and a manually created test set where it has been made publicly available. We observed that the more weak labels we introduced, the more the classification performance dropped. However, this decrease is not significant. Therefore weakly supervised learning may be a suitable method to use in the absence of labeled data. More research is necessary to investigate successful ways to blend weak labels without compromising performance.

As a future work, we intend to use additional data sets to verify our findings. Further, we will explore how to effectively use confidence score to estimate weak label’s effect.

## References

1. Asr, F.T., Taboada, M.: MisInfoText. A collection of news articles, with false and true labels (2019). [https://github.com/sfu-discourse-lab/Misinformation\\_detection](https://github.com/sfu-discourse-lab/Misinformation_detection)
2. Badene, S., Thompson, K., Lorré, J., Asher, N.: Weak supervision for learning discourse structure. In: EMNLP/IJCNLP (2019)
3. Bhutani, B., Rastogi, N., Sehgal, P., Purwar, A.: Fake news detection using sentiment analysis. In: 2019 12th International Conference on Contemporary Computing, IC3 2019 (2019). <https://doi.org/10.1109/IC3.2019.8844880>
4. Castelo, S., et al.: A topic-agnostic approach for identifying fake news pages. In: Companion Proceedings of the 2019 World Wide Web Conference (2019)
5. Chen, T., et al.: XGBoost: extreme gradient boosting. R package version 0.4-2 1(4), 1–4 (2015)
6. Färber, M., Burkard, V., Jatowt, A., Lim, S.: A multidimensional dataset based on crowdsourcing for analyzing and detecting news bias. In: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, pp. 3007–3014 (2020)
7. Fernández-Torres, M.J., Almansa-Martínez, A., Chamizo-Sánchez, R.: Infodemic and fake news in Spain during the COVID-19 pandemic. *Int. J. Environ. Res. Public Health* 18(4), 1781 (2021)
8. From, A.R., Netland, I.U.: Fake news detection by weakly supervised learning: a content-based approach. Master's thesis, NTNU (2021)
9. Gruppi, M., Horne, B.D., Adali, S.: NELA-GT-2019: a large multi-labelled news dataset for the study of misinformation in news articles (2020)
10. Helmstetter, S., Paulheim, H.: Weakly supervised learning for fake news detection on Twitter. In: 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), pp. 274–277. IEEE (2018)
11. Horne, B.D., Adali, S.: This just in: fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news. In: Proceedings of the First Workshop on Fact Extraction and Verification, pp. 40–49, March 2017. <http://arxiv.org/abs/1703.09398>
12. Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., Soricut, R.: ALBERT: a lite BERT for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942* (2019)
13. Liu, Y., et al.: RoBERTa: a robustly optimized BERT pretraining approach (2019)
14. Oshikawa, R., Qian, J., Wang, W.Y.: A survey on natural language processing for fake news detection. In: Proceedings of the 12th Language Resources and Evaluation Conference, Marseille, France, pp. 6086–6093. European Language Resources Association, May 2020. <https://aclanthology.org/2020.lrec-1.747>
15. Özgöbek, Ö., Gulla, J.A.: Towards an understanding of fake news. In: CEUR Workshop Proceedings, vol. 2041, pp. 35–42 (2017)
16. Pennebaker, J.W., Francis, M.E., Booth, R.J.: *Linguistic Inquiry and Word Count*. Lawrence Erlbaum Associates (2001)
17. Pérez-Rosas, V., Kleinberg, B., Lefevre, A., Mihalcea, R.: Automatic detection of fake news. *arXiv preprint arXiv:1708.07104* (2017)
18. Rashkin, H., Choi, E., Jang, J.Y., Volkova, S., Choi, Y.: Truth of varying shades: analyzing language in fake news and political fact-checking. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Stroudsburg, PA, USA, pp. 2931–2937. Association for Computational Linguistics (2017). <https://doi.org/10.18653/v1/D17-1317>. <http://aclweb.org/anthology/D17-1317>

19. Ratner, A., Varma, P., Hancock, B., Ré, C.: Weak supervision: the new programming paradigm for machine learning (2017). <http://ai.stanford.edu/blog/weak-supervision/>
20. Rodrigues, U.M., Xu, J.: Regulation of COVID-19 fake news infodemic in China and India. *Media Int. Aust.* **177**(1), 125–131 (2020)
21. Shrestha, A., Spezzano, F., Joy, A.: Detecting fake news spreaders in social networks via linguistic and personality features. In: CLEF (2020)
22. Shu, K., Mahudeswaran, D., Wang, S., Lee, D., Liu, H.: FakeNewsNet: a data repository with news content, social context and spatialtemporal information for studying fake news on social media (2019)
23. Shu, K., Wang, S., Liu, H.: Understanding user profiles on social media for fake news detection (2018). <https://doi.org/10.1109/MIPR.2018.00092>
24. Spohr, D.: Fake news and ideological polarization: filter bubbles and selective exposure on social media. *Bus. Inf. Rev.* **34**(3), 150–160 (2017). <https://doi.org/10.1177/0266382117722446>
25. Starosta, A.: Building NLP classifiers cheaply with transfer learning and weak supervision (2019). <https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1194/reports/custom/15577251.pdf>
26. Tacchini, E., Ballarin, G., Della Vedova, M.L., Moret, S., de Alfaro, L.: Some like it hoax: automated fake news detection in social networks. arXiv preprint [arXiv:1704.07506](https://arxiv.org/abs/1704.07506) (2017)
27. Tsfati, Y., Boomgaarden, H.G., Strömbäck, J., Vliegenthart, R., Damstra, A., Lindgren, E.: Causes and consequences of mainstream media dissemination of fake news: literature review and synthesis. *Ann. Int. Commun. Assoc.* **44**(2), 157–173 (2020)
28. Van Der Linden, S., Panagopoulos, C., Roozenbeek, J.: You are fake news: political bias in perceptions of fake news. *Media Cult. Soc.* **42**(3), 460–470 (2020)
29. Varma, P., Ré, C.: Snuba. *Proc. VLDB Endow.* **12**(3), 223–236 (2018). <https://doi.org/10.14778/3291264.3291268>
30. Wang, Y., et al.: Weak supervision for fake news detection via reinforcement learning, December 2019. <http://arxiv.org/abs/1912.12520>
31. Wang, Y., et al.: Weak supervision for fake news detection via reinforcement learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 516–523 (2020)
32. Wendling, M.: The (almost) complete history of ‘fake news’, January 2018
33. Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R.R., Le, Q.V.: XLNet: generalized autoregressive pretraining for language understanding. In: Advances in Neural Information Processing Systems, vol. 32 (2019)
34. Yuan, C., Ma, Q., Zhou, W., Han, J., Hu, S.: Early detection of fake news by utilizing the credibility of news, publishers, and users based on weakly supervised learning. In: Proceedings of the 28th International Conference on Computational Linguistics, Barcelona, Spain, pp. 5444–5454. International Committee on Computational Linguistics, December 2020. <https://doi.org/10.18653/v1/2020.coling-main.475>. <https://aclanthology.org/2020.coling-main.475>
35. Zhou, X., Zafarani, R.: A survey of fake news. *ACM Comput. Surv.* **53**(5), 1–40 (2020). <https://doi.org/10.1145/3395046>

**Open Access** This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

