The IPIPAN Team Participation in the Check-Worthiness Task of the CLEF2019 CheckThat! Lab

Jakub Gąsior and Piotr Przybyła

Institute of Computer Science, Polish Academy of Sciences, Warsaw, Poland
j.gasior@ipipan.waw.pl
p.przybyla@ipipan.waw.pl

Abstract. This paper describes the participation of the IPIPAN team at the CLEF-2019 CheckThat! Lab focused on automatic identification and verification of claims. We participated in Task 1 oriented on assessing the check-worthiness of claims in political debate by identifying and ranking, which sentences should be prioritized for fact-checking. We proposed a logistic regression-based classifier using features such as vector representation of sentences, Part-of-Speech (POS) tags, named entities, and sentiment scores. In the official evaluation, our best performing run was ranked 3^{rd} out of 12 teams.

Keywords: Information retrieval, Fact-checking, Logistic regression.

1 Introduction

The recent spread of misinformation in political debates and media has stimulated further research in fact-checking: the task of assessing the truthfulness of a claim.

The CLEF-2019 CheckThat! Lab [6] aims at streamlining a typical fact-checking pipeline consisting of the following steps:

- Identifying check-worthy text fragments (Task 1) [4];
- Retrieving and supporting evidence for the selected claims (Task 2A) [7];
- Determining whether the claim is likely true or likely false by comparing a claim against the retrieved evidence (Task 2B) [7].

This report details our proposed methods and results for Task 1, where we focused on the English part [4]. The overall aim of this task was to identify check-worthy claims and rank them according to perceived worthiness for fact-checking.

The remainder of this paper is organized as follows: In Section 2, we present the works related to the task of identifying check-worthy claims and fact-checking itself. In Section 3, we provide a detailed description of the task, discuss the datasets and performance metrics. Section 4 details the proposed approach and the evaluation results. Finally, Section 5 concludes the paper.

2 State of the Art

Automating the process of fact-checking has been first discussed in the context of computational journalism [5], where authors outline a vision for a system to support mass collaboration of investigative journalists and concerned citizens. They discuss several features of the system to highlight a few important database research challenges such as privacy, trust, authority, data mining and information retrieval.

Thorne and Vlachos [16] survey automated fact-checking research stemming from natural language processing and related disciplines, unifying the task formulations and methodologies across papers and authors. Similar work in the area of political debate was introduced in [18]. Authors detail the construction of a publicly available dataset using statements fact-checked by journalists available online and discuss baseline approaches for the challenges that need to be addressed. Similar datasets were later released [13, 19], where authors collated labeled claims from Politifact. Wang created a dataset of almost 13 thousand claims with additional meta-data such as the speaker affiliation and the context in which the claim appears [19]. Rashkin and Choi later supplemented the Politifact dataset with numerous news articles deemed as hoax according to a US News & World report in order to build a prediction model [13].

One of the biggest datasets of this kind was released in [17], where authors presented a dataset containing over 185 thousand claims generated by altering sentences extracted from Wikipedia and subsequently verified without knowledge of the sentence they were derived from. These claims were later classified as Supported, Refuted or NotEnoughInfo by annotators achieving 31.87% accuracy on labeling a claim accompanied by the correct evidence and 50.91% accuracy, while ignoring the evidence.

Similar work was introduced by Redi and Fetahu [14], where authors provided an algorithmic assessment of Wikipedia's verifiability. They provided algorithmic models to determine if a statement requires a citation, and to predict the citation reason based on custom taxonomy. Authors provided a complete evaluation of the robustness of proposed models across different classes of Wikipedia articles of varying quality, as well as on an additional dataset of claims annotated for fact-checking. Unfortunately, the model could not reliably detect check-worthy claims in the datasets, labeling most of them as negatives.

One of the first complete tools in the area of assessing check-worthiness was provided in [8, 9], where authors proposed the ClaimBuster: a fact-checking platform, using natural language processing and supervised learning to detect important factual claims in political discourses. ClaimBuster uses a sentiment, sentence length, Part-of-Speech (POS) tags and entity types as features in order to rank claims from the least to the most check-worthy. Authors claim average accuracy for sentences fact-checked by CNN of 0.433 and 0.438 for sentences fact-checked by PolitiFact.

A similar tool was introduced in [10], where authors presented the ClaimRank: a multilingual automatic system to detect check-worthy claims in a given text. A model is trained on annotations from nine reputable fact-checking organizations (PolitiFact, FactCheck, ABC, CNN, NPR, NYT, Chicago Tribune, The Guardian, and Washington Post), and thus it can mimic the optimal claim selection strategies. Authors achieved the Mean Average Precision score of 0.323 for English and 0.302 for Arabic.

Finally, in [11] authors present an approach based on universal sentence representations created in collaboration with Full Fact, an independent fact-checking charity. Authors claim an F1 score of 0.83, with 5% relative improvement over the state-of-the-art methods ClaimBuster and ClaimRank, discussed above.

3 Task Description

The objective of the task was to identify check-worthy claims in order to facilitate manual fact-checking efforts by prioritizing the claims that fact-checkers should consider first.

3.1 Datasets

The task organizers provided two datasets: a training set comprised of 19 political debates and speeches (a total 16421 sentences) and a testing set comprised of 7 files (a total of 7080 sentences). Each file was annotated by its speaker and check-worthiness factor (0 or 1) as determined by experts.

The training set contained 440 annotated check-worthy sentences (2.68% of total), while the final testing set contained only 136 check-worthy sentences (1.92% of total).

3.2 Evaluation metrics

The task was evaluated according to the following metrics:

- **Average Precision** - Precision at N, estimated for N check-worthy sentences and then averaged over the total number of check-worthy sentences;
- **Reciprocal Rank** - The reciprocal of the rank of the first check-worthy sentence in the list of predictions sorted by score (in descending order);
- **Precision@N** - Precision estimated for the first N sentences in the provided ranked list;
- **R-Precision** - Precision at R, where R is the number of relevant sentences for the evaluated set.

The official measure ranking the submission of teams was the Mean Average Precision (MAP), calculated over multiple provided debates (each with its own separate prediction file).

4 Proposed Approach

In this section, we describe the details of our approach and present the evaluation results.

4.1 Feature Design and Selection

The features we extracted for each sentence in the database can be divided into the following categories:

- **Bag-of-Words N-Gram Representation of Sentences**: The first step was vocabulary-based vectorization. We employed term frequency–inverse document frequency (TF-IDF) transformation and extracted and built the n-gram model (up to 3) of the dataset. After pruning the most common terms, we ended up with 1006 unigram features, 1177 bigram features, and 1186 trigram features.
- **Vector Representation of Sentences**: We employed word2vec tool to a text corpus as input and produced the word vectors as output. We used a model pretrained on Google News archive [1]. To represent sentences in the provided dataset, we first create 300-dimensional vectors components for each term in the sentence and then select 300 minimal, 300 maximal, and 300 averaged values resulting in a features vector of 900 elements.
- **Types of Named Entities Detected**: The process of recognizing named entities is one the first step towards information extraction that seeks to locate and classify entities in text into predefined categories such as the names of persons, organizations, locations, expressions of times, quantities, monetary values, percentages. We employ the NLTK library [2] to extract a subset of 18 NER tags for each sentence.
- **Part-of-Speech (POS) Tags**: We employ the NLTK's library POS tagger [2] to mark up individual words in a sentence as corresponding to a particular Penn Treebank Constituent Tag, based on both its definition and relationship with adjacent and related words in a sentence. It results in a 36-dimensional feature vector.
- **Sentiment Scores**: To determine the sentiment of each sentence we use BING, AFINN and NRC Lexicons [3] to extract the sentiment score of each term in the sentence, as well as an averaged sentiment score of the whole sentence. It results in a feature vector consisting of 15 elements (11 tags from NRC indicating feelings or emotions, i.e., anger, fear, joy; 2 tags from BING and AFINN; and an overall averaged sentiment of the whole sentence.).
- **Statistical Analysis of Sentences**: We also calculate basic metrics of each sentence after tokenization, such as word count, character count, average word density, punctuation count, upper case count, and title word count. It results in a feature vector consisting of 6 elements.

4.2 Classifier

As our classifier, we selected a L1-regularized logistic regression model (LASSO) or so-called sparse logistic regression model, where the weight vector of the classifier has a small number of nonzero values. By adding the penalty on the weights w:

$$\underset{w,v}{\operatorname{argmin}}\, l_{avg}(w,v) + \lambda ||w||_1, \tag{1}$$

where λ is a regularization parameter, it is possible to achieve attractive properties such as feature selection and robustness to noise [12, 15, 20].

We carried out multiple evaluating runs with different subsets of features discussed in Section 4.1 in order to select the best combination of dependent variables. In order to select the best model to employ in the testing phase, we performed a Leave-One-Out (LOO) cross validation on the whole set of $N = 19$ training debates for various combinations of selected features - training the model on the set of $N - 1$ debates and testing it on the last one. This process was repeated N times.

Table 1 shows the results of Mean Average Precision, Reciprocal Rank and R-Precision metrics achieved during the training phase. As can be seen, none of the analyzed models achieved maximum scores in all measured performance metrics. As a result, top scoring models for each performance metric were selected for the final submission, i.e.:

- **Text2vec + NER + POS + Sentiment** - Primary submission;
- **Text2vec + N-Gram (2)** - Contrastive submission No. 1;
- **N-Gram (1) + NER + Sentiment** - Contrastive submission No. 2.

4.3 Final Results

Twelve teams submitted a total of 25 runs to this task. Table 2 presents the results of our three submission runs as well as the top-ranked submission from the Copenhagen team.

Overall, our primary submission has been ranked third according to the official measure (*Mean Average Precision*), sixth according to *Reciprocal Rank*, and second according to *R-Precision* score. These results allow us to conclude, that the proposed model was better at finding the most check-worthy claims, than at a task of finding all the check-worthy claims in the provided texts. The precise reasons for such behavior will require further analysis.

Surprisingly, our best performing model was one of the contrastive runs, employing text vectors and POS tags as the only selection features. Our primary submission used also NER tags and sentiment scores, which had a negative impact on Mean Average Precision and Reciprocal Rank scores.

We can conclude that this result is caused by a lower representation of NER features in the final testing set. Also, further analysis of the testing set revealed that most of the check-worthy claims had significantly more positive sentiment scores than the claims in the training dataset (see, Table 3). This also impacted the overall performance of the submitted primary model.

Table 1: Mean Average Precision, Reciprocal Rank and R-Precision scores for each model across the provided training datasets. The best results are marked in a bold font.

	Mean Average Precision	**Reciprocal Rank**	**R-Precision**
N-Gram (1) + NER + POS + Sentiment	.1788	.5100	.1957
N-Gram (1) + NER + Sentiment	.1860	**.6382**	.1972
N-Gram (1) + POS + Sentiment	.1781	.5104	.2029
N-Gram (1) + NER + POS	.1756	.5043	.1834
Text2vec	.2367	.4743	.2495
Text2vec + N-Gram (1)	.2127	.4173	.2501
Text2vec + N-Gram (2)	.2200	.4957	**.2623**
Text2vec + N-Gram (3)	.2181	.4604	.2528
Text2vec + N-Gram (1+2+3) + NER + POS + Sentiment	.2185	.4638	.2463
Text2vec + N-Gram (1+2+3) + NER + Sentiment	.2151	.4361	.2513
Text2vec + N-Gram (1+2+3) + POS + Sentiment	.2166	.4649	.2544
Text2vec + NER	.2403	.4959	.2551
Text2vec + NER + POS	.2383	.5543	.2537
Text2vec + NER + POS + Sentiment	**.2415**	.5488	.2469
Text2vec + NER + Sentiment	.2364	.4879	.2498
Text2vec + POS	.2408	.5562	.2519
Text2vec + POS + Sentiment	.2310	.4789	.2526

5 Conclusion and Future Work

In this paper, we present our solution to Task 1 of the CLEF-2019 CheckThat! Lab. For the task of detecting check-worthy claims, we employed an L1-regularized logistic regression (LASSO) classifier. We selected features such as vector representation, named entities, POS tags, as well as averaged sentiment values achieving 3^{rd} place on the English dataset.

This work opens up several possible avenues for future research. First, we intend to employ syntactic parsing and sentence dependency mapping in order to extract additional information regarding the stance of claims, as well as contradicting or confirming sentences during debates. Secondly, we plan to extend the vector representation of sentences to larger segments of text in order to capture additional nuances of longer debates or speeches.

Table 2: Mean Average Precision, Reciprocal Rank and R-Precision scores for each model across the provided testing datasets. The best results are marked in a bold font, while the best results from our submissions are underlined.

	Mean Average Precision	**Reciprocal Rank**	**R-Precision**
Text2vec + NER + POS + Sentiment (primary)	.1332	.2864	.1481
Text2vec + N-Gram (2)	.1365	.3079	**.1490**
N-Gram (1) + NER + Sentiment	.1013	.2791	.1002
Copenhagen (primary)	**.1660**	**.4176**	.1387

Table 3: Average sentiment scores (negative, positive and overall) calculated for check-worthy sentences in training and testing datasets, respectively.

	Negative Sentiment	**Overall Sentiment**	**Positive Sentiment**
Training Dataset	-2.06676	0.017379	1.23742
Testing Dataset	-0.97227	0.045449	1.04570

Acknowledgment

This work was supported by the Polish National Agency for Academic Exchange through a Polish Return grant number PPN/PPO/2018/1/00006.

Bibliography

[1] Word2vec Project. `https://code.google.com/archive/p/word2vec/`. Accessed: 2019-05-24.

[2] Natural Language Toolkit. `https://www.nltk.org/`. Accessed: 2019-05-24.

[3] NRC Word-Emotion Association Lexicon. `https://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm`. Accessed: 2019-05-24.

[4] Pepa Atanasova, Preslav Nakov, Georgi Karadzhov, Mitra Mohtarami, and Giovanni Da San Martino. Overview of the CLEF-2019 CheckThat! Lab on Automatic Identification and Verification of Claims. Task 1: Check-Worthiness.

[5] Sarah Cohen, Chengkai Li, Jun Yang, and Cong Yu. Computational Journalism: A Call to Arms to Database Researchers. In *Proceedings of the Conference on Innovative Data Systems Research*, pages 148–151, 04 2011.

[6] Tamer Elsayed, Preslav Nakov, Alberto Barrón-Cedeño, Maram Hasanain, Reem Suwaileh, Giovanni Da San Martino, and Pepa Atanasova. Overview of the CLEF-2019 CheckThat!: Automatic Identification and Verification of Claims. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, LNCS, Lugano, Switzerland, September 2019.

[7] Maram Hasanain, Reem Suwaileh, Tamer Elsayed, Alberto Barrón-Cedeño, and Preslav Nakov. Overview of the CLEF-2019 CheckThat! Lab on Automatic Identification and Verification of Claims. Task 2: Evidence and Factuality.

[8] Naeemul Hassan, Chengkai Li, and Mark Tremayne. Detecting Check-worthy Factual Claims in Presidential Debates. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, CIKM '15, pages 1835–1838, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3794-6. doi: 10.1145/2806416.2806652. URL http://doi.acm.org/10.1145/2806416.2806652.

[9] Naeemul Hassan, Fatma Arslan, Chengkai Li, and Mark Tremayne. Toward Automated Fact-Checking: Detecting Check-worthy Factual Claims by ClaimBuster. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17, pages 1803–1812, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4887-4. doi: 10.1145/3097983.3098131. URL http://doi.acm.org/10.1145/3097983.3098131.

[10] Israa Jaradat, Pepa Gencheva, Alberto Barrón-Cedeño, Lluís Màrquez, and Preslav Nakov. ClaimRank: Detecting Check-Worthy Claims in Arabic and English. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 26–30, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-5006. URL https://www.aclweb.org/anthology/N18-5006.

[11] Lev Konstantinovskiy, Oliver Price, Mevan Babakar, and Arkaitz Zubiaga. Towards Automated Factchecking: Developing an Annotation Schema and Benchmark for Consistent Automated Claim Detection. *CoRR*, abs/1809.08193, 2018. URL http://arxiv.org/abs/1809.08193.

[12] Andrew Y. Ng. Feature Selection, L1 vs. L2 Regularization, and Rotational Invariance. In *Proceedings of the Twenty-first International Conference on Machine Learning*, ICML '04, pages 78–, New York, NY, USA, 2004. ACM. ISBN 1-58113-838-5. doi: 10.1145/1015330.1015435. URL http://doi.acm.org/10.1145/1015330.1015435.

[13] Hannah Rashkin, Eunsol Choi, Jin Yea Jang, Svitlana Volkova, and Yejin Choi. 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*, pages 2931–2937, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1317. URL https://www.aclweb.org/anthology/D17-1317.

[14] Miriam Redi, Besnik Fetahu, Jonathan T. Morgan, and Dario Taraborelli. Citation Needed: A Taxonomy and Algorithmic Assessment of Wikipedia's Verifiability. *CoRR*, abs/1902.11116, 2019. URL http://arxiv.org/abs/1902.11116.

[15] Jianing Shi, Wotao Yin, Stanley Osher, and Paul Sajda. A Fast Hybrid Algorithm for Large-Scale L1-Regularized Logistic Regression. *J. Mach. Learn. Res.*, 11:713–741, March 2010. ISSN 1532-4435. URL http://dl.acm.org/citation.cfm?id=1756006.1756029.

[16] James Thorne and Andreas Vlachos. Automated Fact Checking: Task formulations, methods and future directions. *CoRR*, abs/1806.07687, 2018. URL http://arxiv.org/abs/1806.07687.

[17] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. FEVER: a large-scale dataset for Fact Extraction and VERification. *CoRR*, abs/1803.05355, 2018. URL http://arxiv.org/abs/1803.05355.

[18] Andreas Vlachos and Sebastian Riedel. Fact Checking: Task definition and dataset construction. In *Proceedings of the ACL 2014 Workshop on Language Technolo-*

gies and Computational Social Science, pages 18–22, Baltimore, MD, USA, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/W14-2508. URL `https://www.aclweb.org/anthology/W14-2508`.

[19] William Yang Wang. "Liar, Liar Pants on Fire": A New Benchmark Dataset for Fake News Detection. *CoRR*, abs/1705.00648, 2017. URL `http://arxiv.org/abs/1705.00648`.

[20] Peng Zhao and Bin Yu. On Model Selection Consistency of Lasso. *J. Mach. Learn. Res.*, 7:2541–2563, December 2006. ISSN 1532-4435. URL `http://dl.acm.org/citation.cfm?id=1248547.1248637`.