Neural Weakly Supervised Fact Check-Worthiness Detection with Contrastive Sampling-Based Ranking Loss

Casper Hansen, Christian Hansen, Jakob Grue Simonsen, and Christina Lioma

Department of Computer Science, University of Copenhagen
{c.hansen,chrh,simonsen,c.lioma}@di.ku.dk

Abstract. This paper describes the winning approach used by the Copenhagen team in the CLEF-2019 CheckThat! lab. Given a political debate or speech, the aim is to predict which sentences should be prioritized for fact-checking by creating a ranked list of sentences. While many approaches for check-worthiness exist, we are the first to directly optimize the sentence ranking as all previous work has solely used standard classification based loss functions. We present a recurrent neural network model that learns a sentence encoding, from which a check-worthiness score is predicted. The model is trained by jointly optimizing a binary cross entropy loss, as well as a ranking based pairwise hinge loss. We obtain sentence pairs for training through contrastive sampling, where for each sentence we find the k most semantically similar sentences with opposite label. To increase the generalizability of the model, we utilize weak supervision by using an existing check-worthiness approach to weakly label a large unlabeled dataset. We experimentally show that both weak supervision and the ranking component improve the results individually (MAP increases of 25% and 9% respectively), while when used together improve the results even more (39% increase). Through a comparison to existing state-of-the-art check-worthiness methods, we find that our approach improves the MAP score by 11%.

Keywords: fact check-worthiness · neural networks · contrastive ranking

1 Tasks performed

The Copenhagen team participated in Task 1 [1] of the CLEF 2019 Fact Checking Lab (CheckThat!) on automatic identification and Verification of claims [4]. This report details our approach and results.

The aim of Task 1 is to identify sentences in a political debate that should be prioritized for fact-checking: given a debate, the goal is to produce a ranked list of all sentences based on their worthiness for fact checking.

Examples of check-worthy sentences are shown in Table 1. In the first example Hillary Clinton mentions Bill Clinton's work in the 1990s, followed by a claim made by Donald Trump stating that president Clinton approved the North American Free Trade Agreement (NAFTA). In the second example Hillary Clinton mentions Donald Trump's beliefs about climate change. While this may be more difficult to fact-check, it is still considered an interesting claim and thus check-worthy.

Table 1. Example of check-worthy sentences (red highlight)

Speaker	Sentence
CLINTON	I think my husband did a pretty good job in the 1990s.
CLINTON	I think a lot about what worked and how we can make it work again...
TRUMP	**Well, he approved NAFTA...**
CLINTON	Take clean energy
CLINTON	Some country is going to be the clean-energy superpower of the 21st century.
CLINTON	**Donald thinks that climate change is a hoax perpetrated by the Chinese.**
CLINTON	I think it's real.
TRUMP	I did not.

2 Main objectives of experiments

The task of check-worthiness can be considered part of the fact-checking pipeline, which traditionally consists of three steps:

1. Detect sentences that are interesting to fact-check.
2. Gather evidence and background knowledge for each sentence.
3. Manually or automatically estimate veracity.

Sentences detected in step 1 for further processing are described as being *check-worthy*. This detection can be considered a filtering step in order to limit the computational processing needed in total for the later steps. In practice, sentences are ranked according to their check-worthiness such that they can be processed in order of importance. Thus, the ability to correctly rank check-worthy sentences above non-check-worthy is essential for automatic check-worthiness methods to be useful in practice. However, existing check-worthiness methods [10,16,11,5,6,9,19] do not directly model this aspect, as they are all based on traditional classification based training objectives.

3 Related work

Most existing check-worthiness methods are based on feature engineering to extract meaningful features. Given a sentence, ClaimBuster [10] predicts check-

worthiness by extracting a set of features (sentiment, statement length, Part-of-Speech (POS) tags, named entities, and tf-idf weighted bag-of-words), and uses a SVM classifier for the prediction. Patwari et al. [16] presented an approach based on similar features, as well as contextual features based on sentences immediately preceding and succeeding the one being assessed, as well as certain hand-crafted POS patterns. The prediction is made by a multi-classifier system based on a dynamic clustering of the data. Gencheva et al. [5] also extend the features used by ClaimBuster to include more context, such as the sentence's position in the debate segment, segment sizes, similarities between segments, and whether the debate opponent was mentioned. In the CLEF 2018 competition on check-worthiness detection [15], Hansen et al. [9] showed that a recurrent neural network with multiple word representations (word embeddings, part-of-speech tagging, and syntactic dependencies) could obtain state-of-the-art results for check-worthiness prediction. Hansen et al. [6] later extended this work with weak supervision based on a large collection of unlabeled political speeches and showed significant improvements compared to existing state-of-the-art methods. This paper directly improves the work done by Hansen et al. by integrating a ranking component into the model trained via contrastive sampling.

4 Neural Check-Worthiness Model

Our Neural Check-Worthiness Model (NCWM) uses a dual sentence representation, where each word is represented by both a word embedding and its syntactic dependencies within the sentence. The word embedding is a traditional *word2vec* model [14] that aims at capturing the semantics of the sentence. The syntactic dependencies of a word aim to capture the role of that word in modifying the semantics of other words in the sentence [13]. We use a syntactic dependency parser [2] to map each word to its dependency (as a tag) within the sentence, which is then converted to a one-hot encoding. This combination of capturing both semantics and syntactic structure has been shown to work well for the check-worthiness task [6,9]. For each word in a sentence, the word embedding and one-hot encoding are concatenated and fed to a recurrent neural network with Long Short-Term Memory Units (LSTM) as memory cells (See Figure 1). The output of the LSTM cells are aggregated using an attention weighted sum, where each weight is computed as:

$$\alpha_t = \frac{\exp\left(\text{score}\left(h_t\right)\right)}{\sum_i \exp\left(\text{score}\left(h_i\right)\right)} \tag{1}$$

where h_t is the output of the LSTM cell at time t, and $score(\cdot)$ is a learned function that returns a scalar. Finally, the attention weighted sum is transformed to the check-worthiness score by a sigmoid transformation, such that the score lies between 0 and 1.

Loss functions. The model is jointly trained using both a classification and ranking loss function. For the classification loss, we use the standard binary cross

entropy loss. For the ranking loss, we use a hinge loss based on the computed check-worthiness score of sentence pairs with opposite labels. To obtain these pairs we use *contrastive* sampling, such that for each sentence we sample the k most semantically similar sentences with the opposite label, i.e., for check-worthy sentences we sample k non-check-worthy sentences. In order to estimate the semantic similarity we compute an average word embedding vector of all words in a sentence, and then use the cosine similarity to find the k most semantically similar sentences with the opposite label. The purpose of this contrastive sampling is to enable the model to better learn the small differences between check-worthy and non-check-worthy sentences. The combination of both the classification and ranking loss enables the model to learn accurate classifications while ensuring the predicted scores are sensible for ranking.

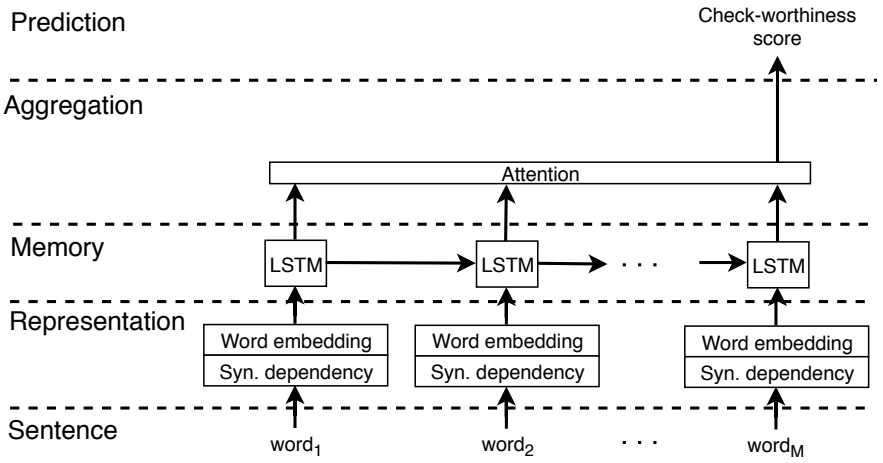

Fig. 1. Architecture of our Neural Check-Worthiness Model (NCWM). The check-worthiness score is used for minimizing the classification and ranking losses.

5 Resources employed

Our approach is summarized in Figure 1, and in the following the underlined values were found to perform the best during validation. The cross validation consisted of a fold for each training speech and debate. The LSTM has {50, 100, 150, 200} hidden units, a dropout of {0, 0.1, 0.3, 0.5} was applied to the attention weighted sum, and we used a batch size of {40, 80, 120, 160, 200}. For the contrastive sampling we found the 5 most semantically similar sentences with the opposite label. For the syntactic dependency parsing we use spaCy[1], and for the neural network implementation TensorFlow.

[1] `https://spacy.io/`

To train a more generalizable model we employ weak supervision [3,6,8,18] by using an existing check-worthiness approach, ClaimBuster[2] [10], to weakly label a large collection of unlabeled political speeches and debates. We obtain 271 political speeches and debates by Hillary Clinton and Donald Trump from the American Presidency Project[3]. This weakly labeled dataset is used for pretraining our model. To create a pretrained word embedding, we crawl documents related to *all* U.S. elections available through the American Presidency Project, e.g., press releases, statements, speeches, and public fundraisers, resulting in 15,059 documents. This domain specific pretraining was also done by Hansen et al. [6], and was shown to perform significantly better than a word embedding pretrained on a large general corpus like Google News[4].

6 Results

For evaluation we use the official test dataset of the competition, while choosing the hyper parameters based on a 19-fold cross validation (1 fold for each training speech and debate). Following the competition guidelines, we report the MAP and P@k metrics for the full test data, only the 3 debates, and only the 4 speeches. This splitting choice was done to investigate how the performance varies depending on the setting.

Overall, our Neural Check-Worthiness Model (NCWM) obtained the first place in the competition with a MAP of 0.1660 (primary run). To investigate the effect of the ranking component and the weak supervision (See Table 2), we also report the results when these are not part of NCWM. The model without the ranking component is similar to the state-of-the-art work by Hansen et al. [6] (contrastive-1 run), and the model without either the ranking component or weak supervision is similar to earlier work by Hansen et al. [9]. The results show that the ranking component and weak supervision lead to notable improvements, both individually and when combined. The inclusion of weak supervision leads to the largest individual MAP improvement (25% increase), while the individual improvement of the ranking component is smaller (9% increase). We observe that the ranking component's improvement is marginally larger when weak supervision is included (11% increase with weak supervision compared to 9% without), thus showing that even a weakly labeled signal is also beneficial for learning the correct ranking. Combining both the ranking component and weak supervision leads to a MAP increase of 39% compared to a model without either of them, which highlights the immense benefit of using both for the task of check-worthiness as the combination provides an improvement larger than the individual parts.

To investigate the performance on speeches and debates individually, we split the test data and report the performance metrics on each of the sets. In both of

[2] `https://idir.uta.edu/claimbuster/`

[3] `https://web.archive.org/web/20170606011755/http://www.presidency.ucsb.edu/`

[4] `https://code.google.com/archive/p/word2vec/`

Table 2. Test results for our full Neural Check-Worthiness Model (NCWM) and when excluding the ranking and weak supervision (WS) components.

Test (Speeches and Debates)	MAP	P@1	P@5	P@20	P@50
NCWM	**0.1660**	**0.2857**	**0.2571**	**0.1571**	**0.1229**
NCWM (w/o. ranking) [6]	0.1496	0.1429	0.2000	0.1429	0.1143
NCWM (w/o. WS)	0.1305	0.1429	0.1714	0.1429	0.1200
NCWM (w/o. ranking and w/o. WS) [9]	0.1195	0.1429	0.1429	0.1143	0.1057
Test (Speeches)	MAP	P@1	P@5	P@20	P@50
NCWM	**0.2502**	**0.5000**	**0.3500**	**0.2375**	**0.1800**
NCWM (w/o. ranking) [6]	0.2256	0.2500	0.3000	0.2250	**0.1800**
NCWM (w/o. WS)	0.1937	0.2500	0.3000	0.2000	0.1600
NCWM (w/o. ranking and w/o. WS) [9]	0.1845	0.2500	0.2500	0.1875	0.1450
Test (Debates)	MAP	P@1	P@5	P@20	P@50
NCWM	**0.0538**	**0.0000**	**0.1333**	0.0500	0.0467
NCWM (w/o. ranking) [6]	0.0482	**0.0000**	0.0667	0.0333	0.0267
NCWM (w/o. WS)	0.0462	**0.0000**	0.0000	**0.0667**	**0.0667**
NCWM (w/o. ranking and w/o. WS) [9]	0.0329	**0.0000**	0.0000	0.0167	0.0533

them we observe a similar trend as for the full dataset, i.e., that both the ranking component and weak supervision lead to improvements individually and when combined. However, the MAP on the debates is significantly lower than for the speeches (0.0538 and 0.2502 respectively). We believe the reason for this difference is related to two issues: i) All speeches are by Donald Trump and 15 out of 19 training speeches and debates have Donald Trump as a participant, thus the model is better trained to predict sentences by Donald Trump. ii) Debates are often more varied in content compared to a single speech, and contain participants who are not well represented in the training data. Issue (i) can be alleviated by obtaining larger quantities and more varied training data, while issue (ii) may simply be due to debates being inherently more difficult to predict. Models better equipped to handle the dynamics of debates could be a possible direction to solve this.

7 Conclusion and future work

We presented a recurrent neural model that directly models the ranking of check-worthy sentences, which no previous work has done. This was done through a hinge loss based on contrastive sampling, where the most semantically similar sentences with opposite labels were sampled for each sentence. Additionally, we utilize weak supervision through an existing check-worthiness method to label a large unlabeled dataset of political speeches and debates. We experimentally verified that both the sentence ranking and weak supervision lead to notable performance MAP improvements (increases of 9% and 25% respectively) compared

to a model without either of them, while using both lead to an improvement greater than the individual parts (39% increase). In comparison to a state-of-the-art check-worthiness model [6], we found our approach to perform 11% better on the MAP metric, while also achieving the first place in the competition.

In future work we plan to investigate approaches for better modelling check-worthiness in debates, as this is important for real-world applications of check-worthiness systems. Specifically, we plan to (1) investigate how context [17] can be included to better model the dynamics of a debate compared to a speech; (2) the use of speed reading for sentence filtering [7]; and (3) extending the evaluation of this task beyond MAP and P@k, for instance using evaluation measures of both relevance and credibility [12].

References

1. P. Atanasova, P. Nakov, G. Karadzhov, M. Mohtarami, and G. Da San Martino. Overview of the CLEF-2019 CheckThat! Lab on Automatic Identification and Verification of Claims. Task 1: Check-Worthiness. CEUR Workshop Proceedings, Lugano, Switzerland, 2019. CEUR-WS.org.
2. J. D. Choi, J. Tetreault, and A. Stent. It depends: Dependency parser comparison using a web-based evaluation tool. In *Annual Meeting of the Association for Computational Linguistics*, pages 387–396, 2015.
3. M. Dehghani, H. Zamani, A. Severyn, J. Kamps, and W. B. Croft. Neural ranking models with weak supervision. In *ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 65–74, 2017.
4. T. Elsayed, P. Nakov, A. Barrón-Cedeño, M. Hasanain, R. Suwaileh, G. Da San Martino, and P. 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.
5. P. Gencheva, P. Nakov, L. Màrquez, A. Barrón-Cedeño, and I. Koychev. A context-aware approach for detecting worth-checking claims in political debates. In *International Conference Recent Advances in Natural Language Processing*, pages 267–276, 2017.
6. C. Hansen, C. Hansen, S. Alstrup, J. Grue Simonsen, and C. Lioma. Neural check-worthiness ranking with weak supervision: Finding sentences for fact-checking. In *Companion Proceedings of the 2019 World Wide Web Conference*, 2019.
7. C. Hansen, C. Hansen, S. Alstrup, J. G. Simonsen, and C. Lioma. Neural speed reading with structural-jump-lstm. *ICLR*, 2019.
8. C. Hansen, C. Hansen, J. G. Simonsen, S. Alstrup, and C. Lioma. Unsupervised neural generative semantic hashing. In *ACM SIGIR Conference on Research and Development in Information Retrieval*, 2019.
9. C. Hansen, C. Hansen, J. G. Simonsen, and C. Lioma. The copenhagen team participation in the check-worthiness task of the competition of automatic identification and verification of claims in political debates of the clef-2018 fact checking lab. In *CLEF-2018 CheckThat! Lab*, 2018.
10. N. Hassan, F. Arslan, C. Li, and M. Tremayne. Toward automated fact-checking: Detecting check-worthy factual claims by claimbuster. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1803–1812, 2017.

11. I. Jaradat, P. Gencheva, A. Barrón-Cedeño, L. Màrquez, and P. Nakov. Claimrank: Detecting check-worthy claims in arabic and english. In *Conference of the North American Chapter of the Association for Computational Linguistics*, pages 26–30, 2018.
12. C. Lioma, J. G. Simonsen, and B. Larsen. Evaluation measures for relevance and credibility in ranked lists. In J. Kamps, E. Kanoulas, M. de Rijke, H. Fang, and E. Yilmaz, editors, *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval, ICTIR 2017, Amsterdam, The Netherlands, October 1-4, 2017*, pages 91–98. ACM, 2017.
13. C. Lioma and C. J. K. van Rijsbergen. Part of speech n-grams and information retrieval. *French Review of Applied Linguistics, Special issue on Information Extraction and Linguistics*, XIII(2008/1):9–22, 2008.
14. T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.
15. P. Nakov, A. Barrón-Cedeno, T. Elsayed, R. Suwaileh, et al. Overview of the clef-2018 checkthat! lab on automatic identification and verification of political claims. In *International Conference of the CLEF Association*, 2018.
16. A. Patwari, D. Goldwasser, and S. Bagchi. Tathya: A multi-classifier system for detecting check-worthy statements in political debates. In *ACM on Conference on Information and Knowledge Management*, pages 2259–2262, 2017.
17. D. Wang, Q. Li, L. C. Lima, J. G. Simonsen, and C. Lioma. Contextual compositionality detection with external knowledge bases and word embeddings. In S. Amer-Yahia, M. Mahdian, A. Goel, G. Houben, K. Lerman, J. J. McAuley, R. A. Baeza-Yates, and L. Zia, editors, *Companion of The 2019 World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019.*, pages 317–323. ACM, 2019.
18. H. Zamani, W. B. Croft, and J. S. Culpepper. Neural query performance prediction using weak supervision from multiple signals. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 105–114, 2018.
19. C. Zuo, A. Karakas, and R. Banerjee. A hybrid recognition system for check-worthy claims using heuristics and supervised learning. In *CLEF-2018 CheckThat! Lab*, 2018.