FactRank : developing automated claim detection for Dutch-language fact-checkers

FactRank: Developing Automated Claim Detection for Dutch-Language Fact-Checkers

Bettina Berendt[1,2,c], Peter Burger[3], Rafael Hautekiet[4], Jan Jagers[5], Alexander Pleijter[3], and Peter Van Aelst[6]

[1] Faculty of Electrical Engineering and Computer Science, TU Berlin, Germany
[2] Department of Computer Science, KU Leuven, Leuven, Belgium
[3] Leiden University Centre for Linguistics, Netherlands
[4] Oqton, Ghent, Belgium
[5] Independent Journalist, contributor to Knack, and Department of Applied Linguistics, VU Brussel, Belgium
[6] Department of Political Science, University of Antwerp, Belgium
[c] corresponding author, e-mail: berendt@tu-berlin.de

10 November 2020

https://people.cs.kuleuven.be/~bettina.berendt/FactRank/

Abstract

Fact-checking has always been a central task of journalism, but given the ever-growing amount and speed of news offline and online, as well as the growing amounts of misinformation and disinformation, it is becoming increasingly important to support human fact-checkers with (semi-) automated methods to make their work more efficient. Within fact-checking, the detection of check-worthy claims is a crucial initial step, since it limits the number of claims that require or deserve to be checked for their truthfulness.

In this paper, we present FactRank, a novel claim detection tool for journalists specifically created for the Dutch language. To the best of our knowledge, this is the first and still the only such tool for Dutch. FactRank thus complements existing online claim detection tools for English and (a small number of) other languages. FactRank performs similarly to claim detection in ClaimBuster, the state-of-the-art fact-checking tool for English. Our comparisons with a human baseline also indicate that given how much even expert human fact-checkers disagree, there may be a natural "upper bound" on the accuracy of check-worthiness detection by machine-learning methods.

The specific quality of FactRank derives from the interdisciplinary and iterative process in which it was created, which includes not only a high-performance deep-learning neural network architecture, but also a principled approach to defining and operationalising the concept of check-worthiness via a detailed codebook. This codebook was created jointly by expert fact-checkers from the two countries that have Dutch as an official language (Belgium/Flanders and the Netherlands). We expect FactRank to be very useful exactly because of the way we defined check-worthiness, and because of how we have made this explicit and traceable.

1 Introduction

The past decade has seen the rise of fact-checking as a new journalistic genre: in addition to their 'internal' pre-publication verification routines, an increasing number of news organisations now publish 'external' checks of factual claims by politicians and others (Graves, 2016; Graves, 2018b; Graves & Amazeen, 2019). In the wake of the 2016 Trump election, concerns about 'fake news' and foreign disinformation operations targeting national voters (Hall Jamieson, 2018) have propelled fact-checking to unprecedented heights as part of the solution for this 'information disorder' (Wardle & Derakshan, 2017).

This solution comes at a cost: compared to other news genres such as news articles or sports reports , external fact-checking is relatively time-consuming. However, ongoing efforts to boost fact-checkers' capacity by providing digital tools and using machine learning and AI have already produced valuable results for many stages of the fact-checking process, e.g., claim detection, image verification, publication, and distribution. Whereas a number of tools benefit all fact-checkers (e.g., the image verification plugin InVID), claim detection tools are language-specific and to the best of our knowledge currently only serve English, Spanish and Arabic language fact-checking. Automated claim detection reduces the labor of monitoring parliamentary debates, talk shows, news coverage, and social media discourse, selecting the claims that warrant checking, so newsrooms can devote their scarce resources to the core task of actually checking these claims.

The present article focuses on the development of FactRank, a claim detection tool for the Dutch language, which is spoken by 24 million people, mainly inhabitants of the Netherlands and (parts of) Belgium. FactRank concentrates on the task of claim detection: sifting through large volumes of texts to find statements that are check-worthy: not only factual (and thus amenable, in principle, to a fact-check), but relevant to a broad public (and thus worth the effort of a fact-check). The FactRank website also performs the task of gathering potential claims, by continuously monitoring relevant sources. Monitoring and claim detection can be thought of as part of a pipeline of further tasks, in particular determining the veracity of statements (part of which may be matching statements to already fact-checked content) and building up a knowledge base. As highlighted above, claim detection is regarded by our journalistic experts as the most helpful candidate for automation in their work.

FactRank's classification algorithm was developed in a novel iterative process that rests on expert fact-checker input, a codebook to support reliable human labelling, and an active-learning approach to combining machine learning and human expertise. Initial tests of an upvote/downvote functionality that is novel in the domain of fact-checking show promising results. Experiments on a dataset of 7037 human-labelled sentences and one involving an additional 1270 human-upvoted sentences show a classification accuracy of up to 74.6%, which is similar to state-of-the-art results for English as well as close to a human-annotators baseline of 75.5% that illustrates the inherent ambivalences of the task and possible upper bounds for machine claim detection.

2 Related work

Automated fact-checking projects generally focus on distinct parts of the fact-checking process, using a variety of approaches, such as NLP and machine learning. Several studies survey progress in the field for a wider audience, comprising journalists and fact-checkers (Babakar and Moy, 2016; Graves, 2018a; Thorne and Vlachos, 2018). Taken together, these present a number of promising, mostly small-scale projects, with few exceptions based on English-language materials. Although considerable progress has been achieved in the past decade, reliable end-to-end systems work, if at all, only for a very limited number of input categories. Automated verification is available for claims that have been fact-checked before, or for relatively simple statements, e.g. the current height of the national deficit, or the name of the president of Brazil.

An area that has seen more success than others, according to Graves (2018a, p. 3), is the first stage of the fact-checking process, claim detection. This entails source monitoring and identifying statements that are both factual and 'check-worthy', i.e. relevant input for fact-checkers (or systems) tasked with verification. An obvious addition to identifying check-worthy statements is ranking these according to their relevance.

Babakar and Moy (2016, 14) break up claim spotting into four distinct tasks:

1. Monitoring claims that have been fact-checked before in new text
2. Identifying new factual claims that have not been fact-checked before in new text
3. Making editorial judgments about the priority of different claims
4. Dealing with different phrasing for the same or similar claims.

The present project, FactRank, deals with items 2 and 3: identifying and ranking new claims. Babakar and Moy (2016, 27-31) list a number of projects in this category, most of which are not relevant for the present study since they cover a different range of sources (e.g., Vlachos' Simple Numerical Fact Checker, meant to spot and check claims such as 'Lesotho has a population of nearly 2 million'; Babakar and Moy, 2016, 28).

The four projects that are closest to FactRank regarding scope and approach are ClaimBuster from the University of Texas (Hassan et al., 2017a and 2017b; https://idir.uta.edu/claimbuster/), ContentCheck (a collaboration of academics and Le Monde, http://contentcheck.inria.fr/), Full Fact's claim spotting module (Konstantinovskiy et al., 2018; https://fullfact.org/automated), and Chequeado's Chequeabot (Graves, 2018a, 5).

ClaimBuster aims to support all stages of the process ("end-to-end fact-checking"), in which claim detection is one stage; ContentCheck focuses on the actual checking and looking up of facts, e.g. in Linked Open Data, rather than on claim detection. Full Fact deploys a claim detection system that leverages transfer learning and universal sentence representations, and it outperforms ClaimBuster and ClaimRank that use word-level representations. Konstantinovskiy et al. (2018) also discuss an ontological approach to the labelling of claims (i.e. which categories of claims should be distinguished) and methods for obtaining these labels (especially crowdsourcing, see also ClaimBuster). These systems work on English-language texts.

ClaimRank (Jaradat et al., 2018 and https://claimrank.qcri.org/) uses a richer set of features than ClaimBuster and is, like Chequeado, one of the currently still limited number of claim detection

systems that work on languages other than English (Chequeado: Spanish, ClaimRank: Arabic). An overview also of earlier computational work in claim detection is given by Leblay, Manolescu, and Tannier (2018). The basic approach of applying supervised learning has remained the same, while a closer inspection of the classes used in early studies also shows the roots of the claim-detection task in sentiment mining: For example, the earliest cited article (Yu & Hatzivassiloglou, 2003) aimed at separating facts from opinions and then focussed, like much of the work in sentiment mining, on a further analysis of the opinion sentences. It appears that the increasing interaction with professional fact-checkers over the years since then has brought the relevance of differentiating within the "facts" class to the fore.

Much work has been done on detecting specific signals in texts. Factmata (https://factmata.com), for example, detects signals of, for example, hyperpartisanship, clickbait, deception, stance, claims validation ("whether a claim is supported or refuted by the evidence found"), subjectivity and arguments[1]. We believe that these signals could be components of check-worthiness, but they are very specific and lack the overarching notion of *relevance to a broad audience* that we have identfied as well as circumscribed by features and questions, as central to check-worthiness.

The recent projects tend to go beyond machine learning and involve journalists throughout. However, based on the published papers, it is difficult or impossible to determine how concepts are defined, what procedures and materials have been established, and who contributes what at which stage. Based on our experience of collaboration, we are convinced that a principled approach is needed and that publicly available documentation is useful.

In addition to studies and tools, datasets have been published. Through the CLEF CheckThat! Competition that has taken place annually since 2018[2], claim detection and veracity detection algorithms have been tested on datasets in English and in Arabic. Datasets have comprised between 50 documents and 1500 tweets, and domains include web pages, social media, debates/speeches/press conferences. The authors of ClaimBuster have, in 2020, released a dataset[3] of 23,533 sentences from all U.S. general election presidential debates (1960-2016) along with human-annotated check-worthiness labels. The dataset, example sentences for the label concepts, and the procedure are described by Arslan et al. (2020).

The main novelty of FactRank, compared to these systems, is (a) its being the first system for claim detection for the Dutch language, (b) a principled and openly documented approach to defining the concept of check-worthiness, and (c) an iterative architecture that leverages the skills of both human annotators and machine learning. As part of our work, we have created (d) a Dutch-language dataset of more than 8000 sentences with human annotations of check-worthiness. FactRank is a product of an interdisciplinary collaboration between professional fact-checkers, computer scientists and political scientists, and it has resulted in a live website that is being used by professional fact-checkers and journalists.

---

[1] https://factmata.com/signals.html

[2] https://sites.google.com/view/clef2020-checkthat/datasets-tools, see also (Elsayed et al., 2019)

[3] https://figshare.com/articles/ClaimBuster_A_Benchmark_Dataset_of_Check-worthy_Factual_Claims/11635293/1

An important part of the approach is the creation of a codebook that guides human coders in labelling examples as check-worthy or otherwise. The concept of check-worthiness (and in particular the notion of relevance that it involves) is notoriously difficult to define both in terms of its meaning and in terms of example datasets; it depends on time, place and context (e.g., Allein & Moens, 2020). We provide an English-language version of our codebook as part of the documentation of our *approach*, with the aim to help and encourage others build codebooks tailored to their materials' times, places, and contexts.

3 Method

FactRank is first of all meant to be a tool for fact-checkers. It should provide them with an instrument that can save them time, by automatically collecting claims that could be relevant to fact-check. Towards this end, FactRank aims to detect check-worthy claims in texts. We use the term 'check-worthy' to denote claims that are factual (meaning that it is possible to check whether they are accurate) and relevant (not every factual claim is relevant for fact-checkers to investigate). Of course, not only fact-checkers, but everyone interested in a critical reading of (online and offline) claims can profit from using FactRank.

Within this large domain, FactRank focusses on political content and in particular on utterances of politicians. This influenced our choice of data sources as well as of labellers, and the design of our codebook. The process was bootstrapped and is being accompanied with professional fact checks and expert coders. The remainder of this section describes these choices in detail.

The organisation of the section reflects the mixed-methods approach of the current paper. Since this yields inherently interleaved writing, we provide a reading recommendation for our two main audiences: (1) The reader used to computer-science documentation will recognise, after a flow-chart description of the processing pipeline (Section 3.1), a section on data (Section 3.4) and on the machine-learning models and training and test set-up (Section 3.5). For this reader, Sections 3.2 and 3.3 provide background information on how the data were labelled, i.e. how the human ground-truth labels were defined and obtained, and what role the machine-learning models played in the selection of instances to be labelled. (2) The reader used to descriptions of human-subjects studies will recognise, after the description of the overall procedure (Section 3.1), the standard components of method: participants (Section 3.3), procedure (detailed actions of different participants: Section 3.3), and materials (data: Section 3.4). Since the concepts underlying the coding were themselves operationalised as part of the overall procedure (namely by the creation and use of the codebook), this conceptual description is given in Section 3.2. For this reader, Section 3.5's description of the machine-learning model provides background information on how the materials that the human participants saw were generated and on how the tool generates labels.

3.1 Procedure (1): the FactRank approach

At the heart of the FactRank approach is an iterative procedure that combines human expertise and machine-learning capabilities to achieve a continuous improvement of the FactRank model's automatic detection of check-worthy claims in incoming streams of text.

Figure 1 shows the basic procedure, split into nine phases. The architecture rests on an interleaved sequence of training and labelling by humans and a machine-learning classifier model, with the size of the labelled datasets increasing across phases. Four of these phases involve the introduction of new, unlabelled data.

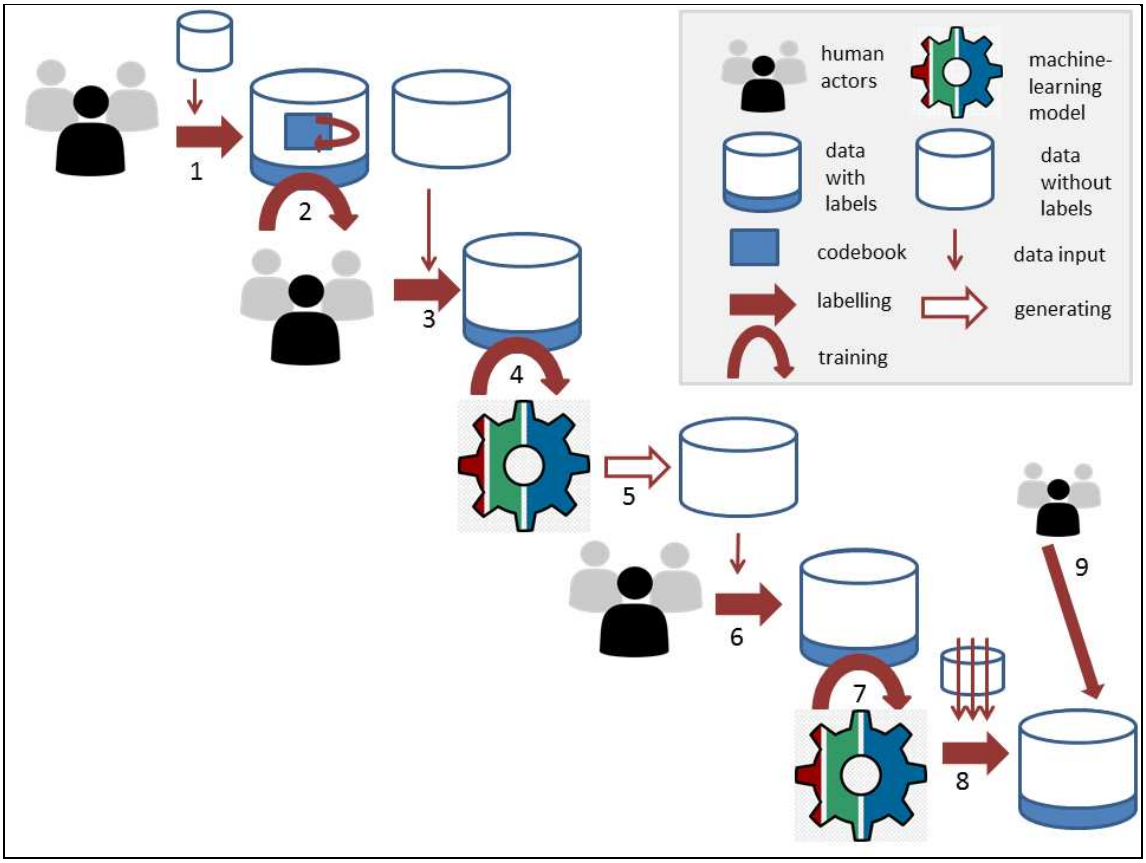

Figure 1. Basic FactRank procedure.

In phase 1, human fact-checking experts labelled sentences concerning their "check-worthiness", and they created a codebook that described the reasoning behind their decisions. The purpose of the codebook was to serve as instructions to knowledgeable (not necessarily expert) human fact-checkers. Specifically, a group of student coders went through a training phase 2, in which they were given the codebook and asked to label the 367 sentences whose ground-truth label had already been established. They received feedback upon mislabelling (the ground-truth label and an explanation of why the experts had assigned that label). At the end of this phase, results were discussed with the experts, and the codebook refined.

Equipped with the refined codebook and their knowledge gained in training, in phase 3 the student coders labelled a set of 2000 new sentences. These labelled sentences were used to train a first version of the machine-learning model in phase 4, in which the model was trained with the student labels as feedback. In phase 5, this model was used to generate a new dataset by selecting a further 5000 sentences that promised to be particularly relevant for learning the concept of check-worthiness, and these 5000 sentences were labelled by "the best" of the student coders in phase 6. In phase 7, all human-labelled sentences accumulated so far were used to train the second version of the machine-learning model. This model is currently (phase 8) being applied "live" on the FactRank website to label new sentences on an ongoing basis, obtained from daily crawls from the Flemish, Belgian and Dutch Parliaments, Twitter, FactCheck Flanders and VRT (Flemish public television) subtitles. Users of the website can give quality feedback by voting sentences up or down, according to their own perception of check-worthiness (phase 9). The usefulness of phase 9 for improving the model was investigated in a pilot study with student coders (different from the ones in phases 2-6). All coders were trained with the codebook and had access to it in all phases of labelling. Further iterations can be added to continue to improve the model.

In the following sections, we provide more details on our notion of check-worthiness and how we operationalised it via a codebook (Section 3.2), the human actors in this pipeline (Section 3.3), the data (Section 3.4), the machine learning (Section 3.5), and the resulting quality of the automatic detection of check-worthiness, including how quality improved through the iterations (Section 4).

3.2 Concepts and their operationalisation: check-worthiness and the codebook

The first step was to compose a codebook[4] with guidelines on how to decide whether sentences contain check-worthy claims. The guiding principle for the coders was to look at sentences from the perspective of a fact-checker: does this sentence contain a claim that could give rise to a fact-check? Coders were therefore instructed as follows.

Take the perspective of a fact-checker: Could this sentence be the start of a fact-checking? Thus, for a sentence to be check-worthy, it should be:

1. *Factual. That means that a sentence should contain a claim that revolves around a fact that can be checked, in other words, that it can be determined whether or not the claim is true.*
2. *Relevant. Not every factual claim is relevant for a fact-checker. Fact-checkers are only concerned with facts that matter to a broad audience. In other words, they are only concerned with claims that, if they turn out to be wrong, are reprehensibly false claims.*

The coding units were entire sentences. They were coded without further context. In other words, the previous and next sentences were not provided to the coders.

The coders had to assign every sentence to one of the following coding categories:

1. NF: Not factual

[4] Available at https://people.cs.kuleuven.be/~bettina.berendt/FactRank/Codeboek_FactRank.pdf (in Dutch) and https://people.cs.kuleuven.be/~bettina.berendt/FactRank/Codebook_FactRank_EN.pdf (in English).

2. FR: Factual and relevant
3. FNR: Factual and non-relevant
4. Error: Not applicable. This code was used for incomprehensible sentences.

These category names derive from ClaimBuster (Hassan et al., 2017a, 2017b), and they operationalise journalistic concepts and practices (e.g. Kwan, 2019). However, a previous study (Laperre et al., 2018) had shown that these categories are far from trivial and require a more rigorous approach, involving explicit coding instructions created by experts. This motivated us to create a detailed codebook that helped to explicate the meaning of the categories,

The *codebook* contains a set of guidelines that should make the coding procedure reliable. The expert coders (see Section 3.3) applied the codebook to a set of sentences and discussed the results. These discussions resulted in additions, adjustments, and refinements of the codebook. The discussions also gave rise to the *Reference Dataset* (see Section 3.4).

The Reference Dataset was used to train six student coders. After a session in which the codebook was explained, the students coded six batches of 50 sentences (phase 2 in Fig. 1). Each batch was discussed with the researchers, and this sometimes this resulted in adjustments or refinements of the codebook.

At first sight, the categories appear relatively clear and easy to spot: statements that claim something about facts (for example, numbers that are or are not correct), would be factual, and opinions would be non-factual (NF). However, relevance for a broad public is key, and the distinction between FR and FNR is often not straightforward. Also, claims are made in different (surface) forms, including as presuppositions, and journalists need to critically investigate all of these. Therefore, a codebook needs to provide more than concept definitions and example sentences: it needs to help coders understand *why* an example sentence would be considered interesting and relevant enough to be checkworthy (or not), and *how* to detect this. This will equip coders with the skills for analysing the topical and linguistic structure of their material. (In a prior step, coders must be selected who having a solid knowledge of the social and political context and a solid competency of the language.)

The following five examples illustrate some of the complexities that a journalistic, fact-checking-based approach entails. Care was taken to describe signals as bases for heuristics that demand holistic judgement rather than mechanistic pattern.

S1: "Together with 122 other countries, we have requested, in the General Assembly of the United Nations, that a ceasefire be declared in Aleppo."

Category: FNR

Explanation: You can check whether the request did indeed involve 122 countries, but for many people a few more or less will not make a difference. In addition, the fact that our country requests a ceasefire is neither controversial nor counterintuitive.

S2: "As regards Canada, it is even 90%."

Category: FR

Explanation: You cannot know what this is about. However, since a number is being mentioned together with the signal word "even", it can be a relevant factual claim. [Other typical phenomena and signal words indicating FR are comparisons: "increasingly", "growing", etc.]

S3: "They do not need paternalism."

S4: "This is therefore a good thing."

Category: NF

Explanation: These sentences may not sound like opinions (they do not contain "I think/I find"), but they are expressions of opinions. You could easily add "I think/I find" to the sentence without changing the meaning: "I think that they do not need paternalism" and "Therefore I find this a good thing."

S5: "I consider it undesirable that 80% of the migrants are unemployed."

Category: FR

Explanation: The sentence begins with an opinion: "I consider". What follows is a factual claim because it needs to be checked whether 80% of the migrants are indeed unemployed. This is also something that many people are likely to find interesting.

3.3 Participants and Procedure (2): The human actors in the FactRank pipeline

Figure 2 gives a more detailed view of the general FactRank procedure, naming actors and datasets.

The *expert coders* were three of the authors of the current paper: Jan Jagers, Peter Burger, and Alexander Pleijter, who work as fact-checkers.

The *student coders* were six students from Leiden University's Master's program Journalism and New Media. They had previously taken a course on fact checking, in which they also conducted a fact-check themselves. All six students labelled in phase 3. In phase 2, these students had been 'scored' by their percent agreement with the ground-truth labels given by the expert coders. The five students with the highest agreement were considered to be the *best student coders*, and they labelled also in phase 5.

Our starting data also contained claims collected from fact-checks done by *other professional fact-checkers*/journalists from relevant Flemish and Dutch media.

Finally, *student coders using the FactRank website* (different from the other student coders) voted on sentences considered check-worthy by the model. The purpose of this phase 9 was two-fold: to serve as a first formative test of a projected functionality for the real-life website and its professional users, and to test the usefulness for model quality of voting as a form of getting human ground-truth assessments, a form that is more convenient than labelling from scratch.

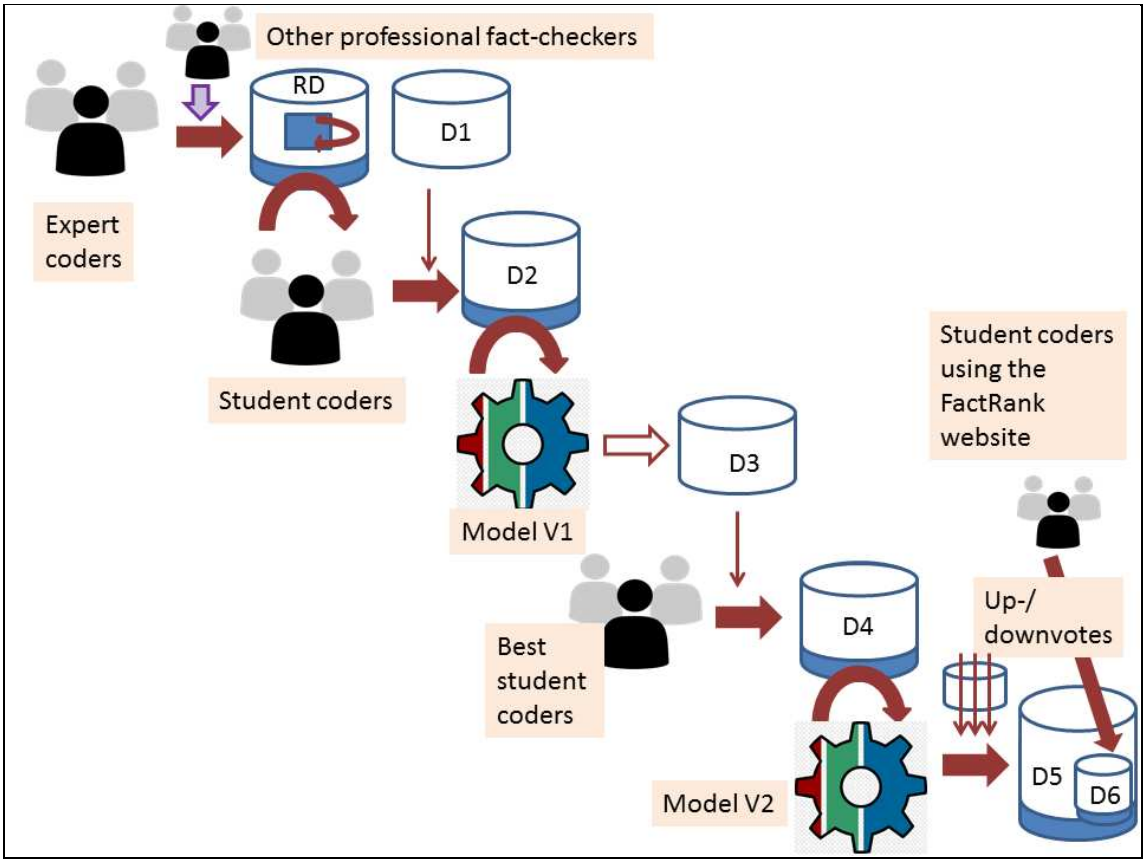

Figure 2. Human actors, machine-learning model versions, and data (figure legend: see Fig. 1)

3.4 Data

Sources

The datasets are derived from a broad range of sources from Flemish and Dutch politics and news sites: BE Parliament Plenary, BE Commission, Interviews BE, Interviews NL, NL tweedekamer, politicians' Twitter accounts, NL fact-check websites, and Factchecks Knack. The sources themselves remained the same over all phases. For phases 1-7 in Fig. 1, a static dataset covering the time between March 2017 and March 2019 was used, whereas phases 8 and 9 draw on a dynamic dataset that is continuously extended by daily collection from the sources.

The reason why we focussed on politicians, and for instance not on clickbait, is twofold. First, politicians embody representative democracy. They hold the parliamentary debate that, in theory at least, leads to, or is part of, the decisions and legislation made. Since that legislation affects all the people politicians speak for and their everyday lives, in a functioning democracy, politicians should use correct facts in their arguments. Here, "correct" means facts that are as undisputed as possible. Political views differ regarding what to do. But the facts on the table – the building blocks of

discussion – should ideally be agreed upon. We thus focus on politicians because their words are the beating heart of democracy.

Second and related to that first argument, monitoring politicians' spoken – and written(-down) - words is very time-consuming. Politicians produce large volumes of text and arguments every day as well in parliament, via their direct communication channels – i.e. social media – and in interviews in news outlets on paper, radio, and TV. Considering the first argument above, we consider the need for journalists to receive assistance, a priority.

Our selection of sources considers the traditional arena of political actors (parliament) as well as mainstream media (from which interviews were taken) and Twitter as the most recent arena. Especially on Twitter, there is no control over what is written and no gate-keeping. This is one of the reasons why – in the Flemish/Dutch environment described here as well as elsewhere – the need for fact-checking has increased tremendously over the past years.

The sources we used are, in descending order of importance, (1) transcripts of plenary debates held in the Belgian federal and in the Dutch parliament; (2) interviews with politicians from different political parties in Flemish and Dutch newspapers; and (3) Flemish and Dutch politicians' writings on their microblog Twitter. These data were supplemented with claims that had already been fact-checked by Dutch and/or Flemish media. Those claims came from politicians, but also from other pundits such as academics or experts cited in media coverage, and also from that coverage itself – for instance, a newspaper headline. The parliamentary debates are complete, as is the list of Twitter accounts from the parliamentarians. Sources are representative in the sense that the dataset in parliament from elected politicians that represent a wide range of different political/ideological viewpoints. Since both Belgium and the Netherlands have proportional electoral systems and low electoral thresholds, the number of parties represented in parliament is relatively high (7 Flemish parties in Belgium and 13 parties in the Netherlands). The selections of interviews and fact-checks were done based on subjective assessment of interestingness by the experts in our team. Data are scraped or obtained via the API (Twitter); no data cleaning issues have arisen. The distribution over sources is reported in Section 3.5 for the main evaluation dataset (D_4, explained further below).

Datasets

The static dataset D_{total} consists of 410,000 sentences. All datasets to be labelled were selected from D_{total} , in ways that aimed at selecting "interesting" sentences, as described below.

First, 300 sentences were selected from this set to be labelled by our fact-checking experts in step 1. To avoid creating a useless dataset consisting mostly of uninteresting NF sentences, the 300 were not chosen randomly. Instead, the machine-learning model from the earlier study (Laperre, Merchiers, & Hautekiet, 2018) was applied to all 410,000 sentences, for each of the categories FR, FNR and NF, sentences were ranked by their score ("most likely to be FR, as judged by the model" etc.), and the top-ranking sentences selected such that the distribution over (likely) FR, (likely) FNR, and (likely) NF was uniform. This resulted in 217 sentences for which our three expert labellers agreed (see Section 3.3). After initially labelling each sentence and explaining their decision individually, our three expert labellers also agreed on one "explanation" of each of these sentences.

A further 150 sentences from other professional fact-checkers were added to this. Regardless of whether these sentences were judged to be true, false, half-true, etc. by the other professional fact-checkers, the fact that they had been selected for this test indicated that the other professional fact-checkers deemed them check-worthy. Our experts agreed with these judgments. The combined dataset of 367 labelled sentences (the Reference Dataset, *RD* in Fig. 2) was used to train the student coders in phase 2. Our experts continued labelling further sentences.

The 2000 sentences chosen for labelling in phase 3 (D_1 in Fig. 2) were selected as follows:

We trained the SVM model from Laperre et al. (2018) on the 517 sentences that the experts had agreed on so far (110 FR, 224 FNR, 183 NF). The model was trained in a binary setting, i.e. FR against the rest, and applied to 10,000 sentences from D_{total}. From the result, 1000 sentences with >50% confidence of being FR were taken, and a further 1000 randomly chosen. This choice of instances reflected a utility metric (Fu, Zhu, & Li, 2013) focused on precision in the early phases of model learning: we wanted to reduce the uncertainty of instances considered FR by our model via obtaining judgments from human labellers. The two-class setting was used only as a step in generating datasets to be labelled; the models were evaluated with respect to the three-class setting (see Section 4 below).

Each sentence in D_1 was labelled by all 6 student coders (i.e. each coder labelled 2000 sentences), and the majority labels were taken to be the ground-truth labels for these sentences in D_2. This resulted in 702 sentences for which there was majority agreement.

The 5000 sentences chosen for labelling in phase 5 (D_3 in Fig. 2) were selected as follows:

We trained the SVM from (Laperre et al., 2018) on the 2661 sentences that the experts (in their continuing labelling process), or the student labellers in phase 3, had agreed on so far (328 FR, 1227 FNR, 1067 NF). The model was again trained in a binary setting and applied to D_{total}. From the result, 5000 sentences close to the decision boundary of the SVM (i.e. around 50%) were chosen, stratified by source (1000 each from interviews BE, interviews NL, plenary/commission transcripts, 2nd chamber NL, and Twitter). This choice in a more advanced phase of model learning reflected a more general utility metric, that of choosing instances that the model is uncertain about.

Each sentence in D_3 was labelled by 2 out of the 5 'best student coders', such that each coder labelled 2000 sentences, and the agreed-upon labels were taken to be the ground-truth labels in D_4. Sentences from D_3 on which the two coders disagreed were excluded from further consideration.

In the next step, the results from all previous steps of human labelling were used as the ground-truth dataset, consisting of a total of 7037 sentences (D_4 in Fig. 2). All sources originally crawled were represented in the dataset (see Table 3). 1100 sentences had been labelled by other professional fact-checkers, and 5937 by our student and expert coders. This dataset is available at https://github.com/lejafar/FactRank/tree/master/factrank/data/training.

Current deployment results (D_5 in Fig. 2) originate from continuous source monitoring and labelling of the new data. They are stored for future iterations of the FactRank pipeline.

We then created a dataset D_6 by asking three student coders (from Antwerp University and not involved in the earlier rounds) to vote on outputs from the live FactRank website, i.e. on D_5. The coders were trained by one of the experts in a similar way as the earlier Leiden coders. First, we explained the goal of Factrank and gave a detailed explanation of the codebook, and we discussed

the examples used in the codebook. Next, we performed a small test with all three coders, giving them the same 20 statements. Since the Factrank website only allowed a binary classification, we discussed the results with a clear focus on distinguishing between check-worthy and non-check-worthy statements. Coders in this test run of step 9 in Fig. 1 were instructed as follows: "Go to factrank.org and look at all sentences from the Flemish Parliament, the [Belgian] Federal Parliament, and the Dutch Parliament. [Each student concentrated on one source.] For every sentence, do the following: If you think this sentence is *indeed* check-worthy, vote it up with the upvote button. If you think the sentence is *not* check-worthy, vote it down with the downvote button. If you are unsure, do not vote."

From these, we selected the upvoted sentences. These sentences can be used directly as FR statements. (Downvotes are either FNR or NF, and they can only be used when shifting to a binary FR vs. not FR classifier, which is left for future development and not considered in the present paper.) A further motivation was to boost the number and variety of positive examples (considered check-worthy by humans, similar to phase 3, inspired by classical strategies of relevance feedback in interactive text retrieval). This resulted in 1270 sentences with an upvote. The model V2 was re-trained using this set D_6 with the new upvotes.

An overview of the numbers of sentences in the human-labelled datasets is given in Table 1. These are at the same time the class distributions in the input datasets used for model training. The outputs of model training are summarised in Table 4.

Dataset	**FR**	**FNR**	**NF**	**Total**
RD	110	224	183	517
D_2	328	1227	1067	2622
D_4	1808	3539	1690	7037
$D_4 \cup D_6$	3078	3539	1690	8307

Table 1. Class distributions in the ground-truth datasets labelled by human annotators.

3.5 The FactRank machine-learning model

The machine-learning model V0 and V1 was inspired by the method for "claim detection" of Claimbuster. It used, like Hassan et al. (2017a, 2017b) did, a support vector machine (SVM). A linear kernel was used because it gave the best classification quality in preliminary tests. The features of the SVM included uni- and bi-grams, POS tags derived using *pattern*[5] and sentiment analysis scores also derived using *pattern*. The process is illustrated in Fig. 3.

[5] https://github.com/clips/pattern

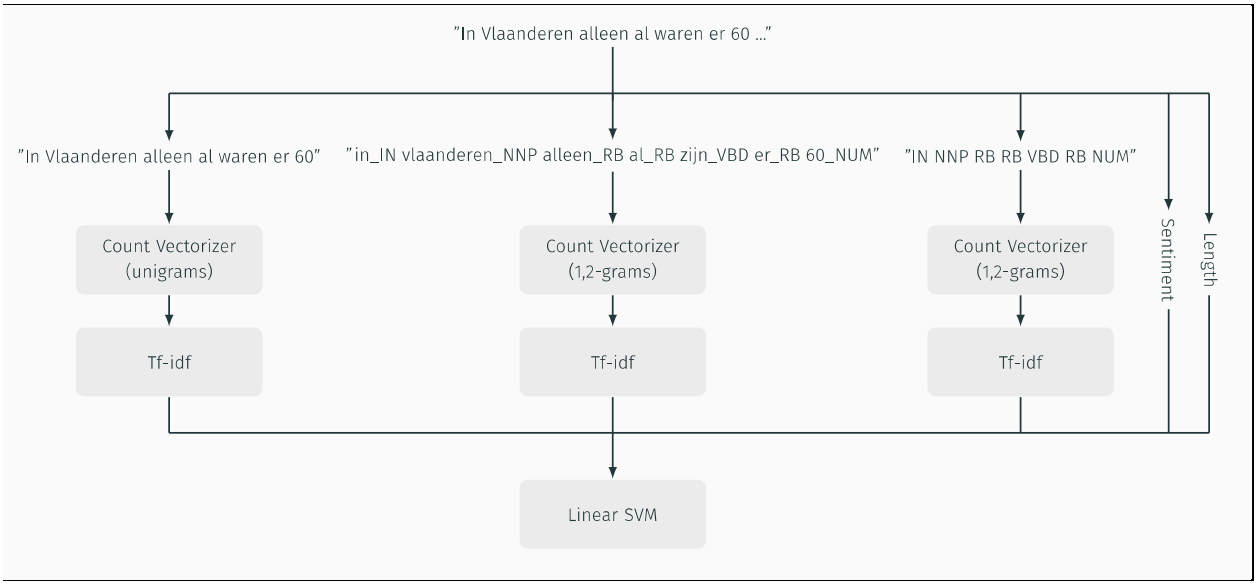

Figure 3. Processing in the SVM model, illustrating features with an example sentence.

The machine-learning model V2 built on the method for sentence classification by Kim (2014). We trained a convolutional neural network (CNN) with one convolutional layer with different kernel sizes from 2 to 6, all consisting of 100 channels each and followed by a max-pooling and a ReLU activation. The resulting matrices were concatenated and followed by one linear layer. Inputs are word vectors obtained from an unsupervised language model. The word vectors are the COW-big vectors of Tulkens, Emmery and Daelemans (2016), a Dutch word-embeddings resource, trained on the COW corpus (Schäfer & Bildhauer, 2012), which is similar to word2vec as used by Kim (2014). In COW-big, 320 word-vector dimensions' values are given for a vocabulary of size 3,110,718. Linking this to our training set resulted in a total vocabulary of 12,000 words enriched by word vectors; if a sentence in the training set contained an unknown word, this was assigned a random vector. If, during inference, a sentence contained an unknown word, this was dropped from the to-be-classified sentence. This mirrors the architecture that Kim (2014) found to perform well and robustly across several different sentence-classification tasks. Experiments with both static and non-static word embeddings were conducted. (Non-static embeddings may be updated during the training process, while static ones remain unchanged.) Static word embeddings resulted in the best performance.

Regularisation was done by the addition of random noise. To prevent overfitting, dropout (p=0.6) was added after the convolutional layer and weight decay (L2 regularization) was also used. We started from the same hyperparameters as Kim (2014) and optimised them via a grid search on the validation set, and, again as in the example method, trained through stochastic gradient descent over shuffled mini-batches with Adam ($\beta_1 = 0.9$, $\beta_2 = 0.999$) as optimizer with an initial learning rate of 0.001, which was decreased after 100 epochs without improvement. In total, the model trained for 1000 epochs.

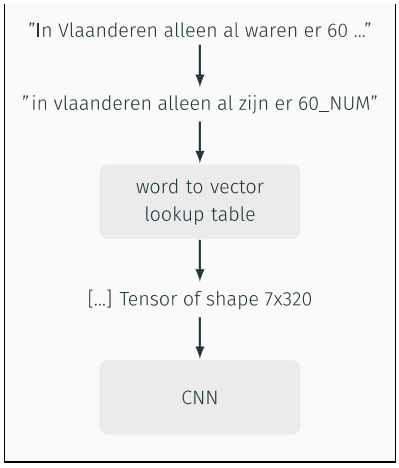

Figure 4. Processing in the CNN model, illustrating features with an example sentence.

For the CNN, the data were pre-processed via part-of-speech tagging and a restriction to the part-of-speech "NUM" that identifies numbers. This NUM identifier was appended to the word in the sentence and treaded as a regular word. Numbers had been identified as valuable indicators of factual statements already in our preliminary tests (Laperre et al., 2018). The codebook discussion confirmed this observation, but also made it clear that not all numerical claims are also check-worthy (see Section 3.2). Therefore, it appeared beneficial to use numbers (which can be identified automatically with high accuracy) as a feature, but of course as one that is complemented by other features. The process is illustrated in Fig. 4. For learning and evaluating, we held out a test set from the labelled dataset of 150 sentences approximately balanced over the three classes FR, FNR, and NF, and split the remaining data into training and validation sets as shown in Table 2. The dataset is available at https://github.com/lejafar/FactRank/tree/master/factrank/data/training.

The distribution over sources is shown in Table 3.

# sentences	**Training set**	**Validation set**	**Test set**
FR	1561	200	47
FNR	3285	200	54
NF	1441	200	49
Total	6287	600	150

Table 2. Dataset statistics for the labelled dataset D_4: Class distribution (# sentences).

Source	% out of 7037
Be Plen 2017-2019	19 %
BE Commission	14 %
Interviews BE	10 %
Interviews NL	6 %
NL tweedekamer	28 %
Twitter	15 %
NL factcheck websites	6.2 %
Factchecks Knack	1.8 %

Table 3. Dataset statistics for the labelled dataset D_4: Source distribution (% of sentences).

Towards the utility metric for the choice of instances for labelling (see Section 3.4) as well as towards an interpretable "check-worthiness score" for the users of the website, we derived probabilistic scores via Platt scaling for the SVM and the softmax over the output nodes of the network for the CNN. We refer to these scores as *confidence* throughout the paper.

The code is available at https://github.com/lejafar/FactRank/.

4. Results: Quality indicators of check-worthiness detection

External Comparison

The authors of Claimbuster report a precision of 72% and a recall of 67% on check-worthy factual claims (Hassan et al., 2017a), and 79% / 74% in (Hassan et al., 2017b), for an SVM whose features include the words and their part-of-speech tags. However, it is difficult to compare these numbers with ours, since the authors used different conventions about check-worthiness and different types of coders ("mostly university students, professors and journalists who are aware of U.S. politics"), a different number of coders (more than 300), only 30 sentences for training and apparently no codebook.

In addition, results on English-language texts tend to be higher than those on languages that are less studied by natural language processing. (For an example of such differences from the domain of claim detection, see the direct comparison in Jaradat et al., 2018).

Comparing Model Versions and a Human Baseline

Table 4 shows the results of the SVM model (Laperre et al., 2018) trained on our data from two phases of the the overall process, when tested on our final test set (see Section 3.5), as well as the improvement obtained through the CNN model, first on the same labelled data and second on these data together with the dataset D_6 obtained by voting. We compare these results to a a "human baseline", described in the following paragraphs, as well as to other classifiers chosen to represent state-of-the-art architectures. All models were trained on the three classes FR, FNR and NF, and accuracy values are averaged over the three classes (which have equal proportions in validation set and test set).

We also investigated the performance on the validation set. Different feature sets were compared, for the SVM model, by Laperre et al. (2018) (the features that we continued to use here were found to perform best). A comparison of different feature sets for the CNN model is the subject of future work.

	Classfication accuracy (test set)	**Classification accuracy (validation set)**
Human baseline	75.5 % (79% resp. 72%)	/
SVM Model (trained on D_2)	44.0 %	/
SVM Model (trained on D_4)	65. 3 %	63.4 %
CNN (Ours) (trained on D_4)	70.6 %	68.9 %
SVM Model (trained on $D_4 \cup D_6$)	66.0 %	63.7 %
Random Forest (trained on $D_4 \cup D_6$)	48.6 %	54.2 %
CNN (Ours) (trained on $D_4 \cup D_6$)	**74.7 %**	72.1 %
BERTje (with frozen encoder, trained on $D_4 \cup D_6$)	70.7 %	**74.8 %**

Table 4. Accuracy comparison: models, data, architectures, human baseline.

BERTje is a Dutch BERT (transformer) model (de Vries et al., 2019)[6]. BERTje scored better on our validation set, but our CNN outperformed it by a clearer margin on the (smaller) test set. One might expect a BERT model to perform well also on the smaller test set due to the model's larger size and

[6] weights reference: https://github.com/wietsedv/bertje

expressive power compared to the CNN model and given its performance on other tasks. However, the specific sentence classification task here might not benefit all that much from long term patterns in the sentence – rather, having a model that efficiently uses word occurrences and small word patterns (up to 5 words long) may already learn as much as there is to learn from our dataset.

Human Baseline

These numbers may appear low. However, an inspection of human inter-rater agreement showed that the problem of detecting check-worthy statements is a hard problem– less so because it is hard to detect a factual statement, more so because it is hard to agree on relevance.

We measured inter-rater agreement by Krippendorff's alpha. In an initial meeting, two of the authors (Jan Jagers (JJ), professional fact-checker, and software developer Rafael Hautekiet (RH), software developer and co-author and rater in (Laperre et al., 2018), obtained an alpha of 0.15 on 75 sentences. The same team reinforced by two further fact-checkers (Peter Burger, PB, and Alexander Pleijter, AP) obtained average alphas of 0.54, 0.5 and 0.5 over three phases of the development of the codebook. In training, the six student coders achieved an average alpha of 0.5 (6 runs of 50 sentences each). Over the next 4 runs of 500 sentences each, their average alpha remained nearly equal at 0.49. In the remaining 10 runs of 500 sentences each, the 5 best student coders achieved an average alpha of 0.4. Finally, we also investigated the alpha between two of our human fact-checkers on the test dataset of 150 sentences and found it to be 0.57. Individually, they agreed with the ground truth on "only" 72% (PB) resp. 79% (JJ) of these sentences. This comparison also suggested that the students tended to label more conservatively: both experts labelled more sentences as FR (factual and relevant) than the ground truth dataset.

5. FactRank live: the Website

The website https://factrank.org runs the software, model V2, plus some additional experimental features. Daily crawls identify sentences from the Flemish, Belgian and Dutch Parliaments, Twitter, FactCheck Flanders, and VRT (Flemish public broadcaster) subtitles. These are labelled with the three classes and shown as "*burningly" check-worthy* (in case of model confidence for FR of >= 99%), *check-worthy* (between 85% and under 99%), *might be check-worthy* (between 50% and under 85%), and *not check-worthy*. An example screenshot is seen in Fig. 5.

The interface contains standard string-search and filtering options (by source, by country: Belgium and/or the Netherlands), and time. In addition, it allows users to give us feedback and thereby help us improve the model: They can upvote ("I think this sentence is check-worthy") or downvote ("I do not think this sentence is check-worthy") the sentence. The interface shows the number of upvotes and downvotes as information in addition to the system's assessment (see the buttons on the right hand side of the check-worthiness label).

Through the site's default ranking by check-worthiness, users will generally focus on the first sentences and give feedback on these. In this way, we again focus on improving the precision of the "check-worthy" class (cf. the remarks on utility metrics in Section 3.4).

In the design of the interface, we have taken the first step towards taking into account context: Where adjoining sentences are given by the original texts, users see this context in grey. In addition, they see the speaker and the political party. In future iterations of the software, we aim at also letting the machine-learning process draw on this context.

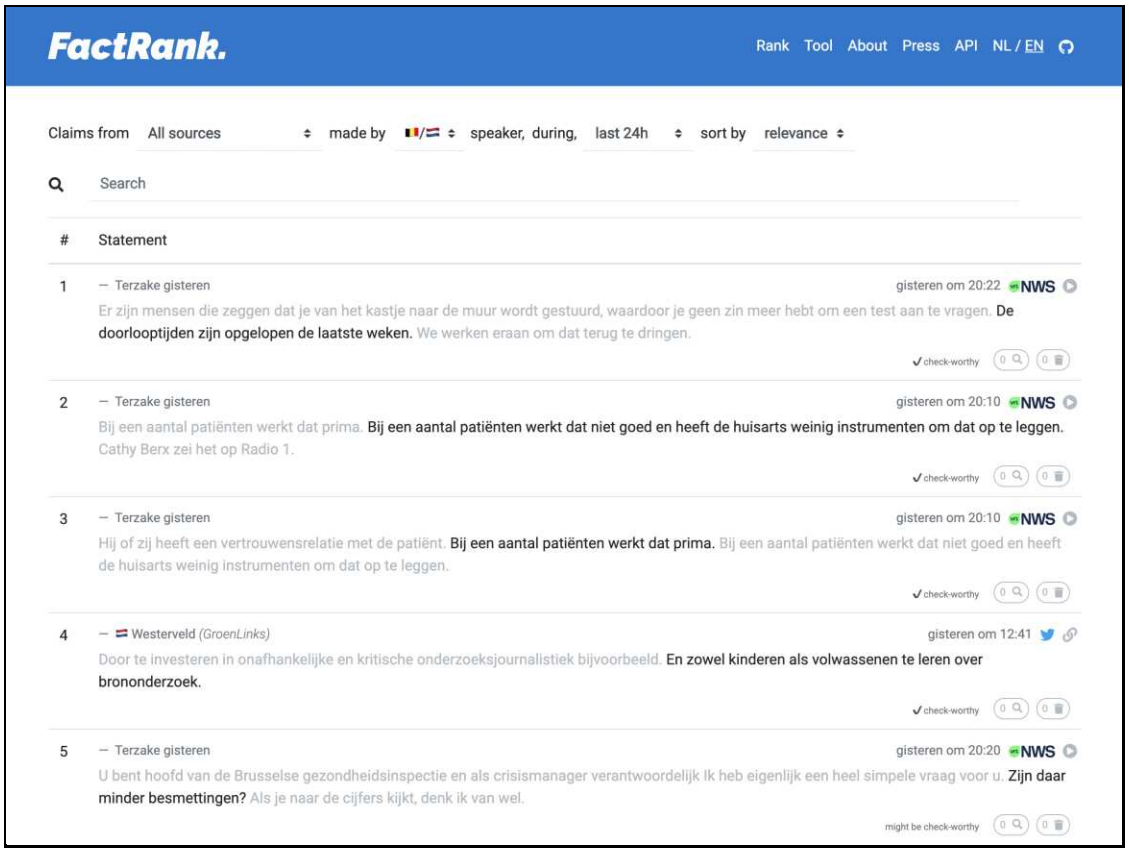

Figure 5. The interface of https://factrank.org

Further uses are possible. For example, people can also use the tool to get a quick 'factual' view on the most recent scanned parliamentary debate or news program, and thus get an overview on what issues were debated and which facts (or claims) were involved.

6 Limitations: Counter-intuitive results, errors, or both?

To identify possible patterns in the classifications of FactRank 'in the wild', we subjected the website version of Factrank (model V2) to a qualitative test. We fed it sentences from current interviews with Flemish politicians and concentrated on sentences that we considered, as human readers of the whole text, as particularly check-worthy. We were surprised to discover that sentences such as the following ones were not considered check-worthy by the model. We group them by the type of limitation they demonstrate and discuss possible reasons.

Factual statements including quantitative claims

E1: In some places, such as the district of Merksem, there is up to 30% child poverty.

E2: 20% of pupils in Antwerp finish [school] without graduating.

E3: For example, in Antwerp alone there are already about 9000 NEET-youngsters. [NEET youngsters are people between 15 and 24 that are neither at school nor employed and that do not receive any social-security / welfare benefits.] [7]

While these results may appear counterintuitive, they are a result of a conscious decision that was made early on in the pipeline. Quantitative claims (whether referring to percentages or absolute numbers) are easy to spot and flag for a computational model, and such claims were given high scores in the previous version V0 of the model. However, while numbers are good indicators of *factual* claims, many such claims are not *relevant*.

This issue was discussed in the construction of the codebook and resulted in the instruction to code a claim about numbers as relevant only if it appears as extraordinary, for example significant developments or increases, or likely to raise controversy (for full details with examples see codebook, pp. 8ff.). Coders were instructed to search for "signal expressions" such as "even" or "only" or "then we will reach". They were also instructed to code unclear sentences as not relevant and to check on abbreviations unknown to them (such as, possibly, NEET) with a quick Web search only. They were also given the semantic instruction to consider what a broad public would find interesting, and the question is whether child poverty and school dropout (a) occurred in the examples labelled by the student coders and, if so, (b) were considered by them a burning social question, and if not, (c) whether this is a valid reflection of the Flemish/Dutch public at large.

It is therefore likely that the computational model was not able to find many indicators of relevance in these examples, neither in the semantics of the nouns and verbs, nor in the words "tot" and "al". While these words mean "up to" and "already" in the example sentences, which suggests a certain emphasis/comparison, the Dutch words have many other meanings too, such that their "signalling power" is likely to have been diluted in the language model employed and learned by the classifier. The classifier is also currently not equipped with a knowledge-enrichment mechanism that would allow it to "look up" abbreviations such as NEET. In this sense, the low scores for E1-E3 are not errors, since they just reproduce the codebook instructions (and the sentences that were coded in line with these).

[7] Dutch original sentences: *Er is op sommige plaatsen, zoals het district Merksem, tot 30% kinderarmoede. / Ja, 20% van de Antwerpse scholieren stopt voor ze een diploma secundair hebben gehaald. / Tja, er zijn bijvoorbeeld alleen in Antwerpen al zo'n 9.000 NEET-jongeren.*

It should also be kept in mind that the model has to classify the sentences without the context of the surrounding text. In our informal test setting, we read the whole interviews, and also the students whom we asked to vote these sentences up or down with the help of the website saw at least some context. (For example, a definition of NEET was given in the subsequent sentence in the interview.) In future work, more textual context should be drawn upon for classification.

Complex factual expressions

E4: The French-speaking [citizens of Belgium] will demand that the transfers to Wallonia, which from 2025 onwards begin to decrease, be extended for at least ten years.

E5: The tax deficit amounts to up to 11 billion Euros; the costs for social security are continuing to rise.

E6: Leuven has 100,000 inhabitants, of which 17,000 non-Belgians.

E7: In the North, life expectancy is 90, in the South, it is 60. [a statement about Chicago.] [8]

These sentences too are factual and contain numbers, but fail to reach the threshold of relevance to which the model has been trained. They again illustrate some problems of lexicality. For example, "continuing to rise" does indicate a significant increase, but the Dutch expression used here is "blijven stijgen", literally "remain increasing", and it is likely that the word "remain", which is frequently an indicator of a *lack* of change, does not activate relevance in the model. The word "non-Belgians" only attains significance in context (such as the implicit statement here that 17% of the inhabitants of the university town of Leuven are non-Belgians, which is higher than the national average of 12%[9]). The statement with high social significance in the last sentence is contrastive but phrased without any remarkable signal words.

A closer inspection of variants of these sentences showed further interesting implications of the current approach. Replacing "Leuven" by "Brussels" in E6 increases the check-worthiness score from 4% to 13%. This suggests that through a combination of the (word-embeddings) language model and the words and word combinations learned in training, sentences about the Belgian capital appear more relevant to the model than sentences about other cities. Changing the "90" in E7 to different numbers increases the check-worthiness score (an effect that again may arise from certain numbers, such as 100, being found in relevant sentences). Changing the "90" in E7 to "60" and thereby annihilating the contrastive statement also increases significance (from 16% to 42%), which of course simply illustrates the lack of grammatical, semantic and pragmatic understanding of our model.

If the two parts of the conjunction in E5 are scored separately, the check-worthiness score increases from 2% to 30% (first conjunct) resp. 5% (second conjunct). This may indicate the problems that arise when a model that does not understand grammar and was trained with the

---

[8] Dutch original sentences*: De Franstaligen zullen eisen dat de transfers naar Wallonië, die vanaf 2025 beginnen te dalen, minstens tien jaar worden verlengd. / Het begrotingstekort loopt op tot 11 miljard euro, de kosten in de sociale zekerheid blijven stijgen./ In Leuven wonen 100.000 mensen, waarvan 17.000 niet-Belgen. / In het noorden wordt men gemiddeld 90 jaar, in het zuiden 60.*

[9] https://de.statista.com/statistik/daten/studie/73995/umfrage/auslaenderanteil-an-der-bevoelkerung-der-laender-der-eu27/

assumption that 1 sentence equals 1 claim, has to classify a sentence that consists of two claims. This effect is strengthened if the two clauses are combined in a grammatically more involved way: If the main clause and the relative clause of E4 are scored separately, the check-worthiness score increases from 0% to 6% (relative clause) and 0% (main clause). In addition, if the tense is changed (from "will demand" to "demand"), the score of the main clause rises from 0% to 5%.

In future work, we will formalise these observations and experimentally investigate effects of feature, resource, and architectural choices on their occurrence.

7. Conclusions and future work

7.1 Novelty, Quality and Significance

We have presented FactRank, a novel claim detection tool for journalists and fact-checkers, specifically created for the Dutch language. To the best of our knowledge, this is the first and still the only such tool for Dutch. FactRank thus complements existing online claim detection tools for English, Spanish and Arabic. As such, it has already garnered substantial media attention.[10]

FactRank performs similarly, in terms of accuracy, to the most comparable system for English (ClaimBuster). Our comparisons with a human baseline also indicate that given how much even expert human fact-checkers disagree, there may be a natural "upper bound" on the accuracy of check-worthiness detection.

The intensive discussions about journalistic questions that inspire, and in turn are influenced by questions of computational modelling, helped us understand possible reasons for the difficulty of capturing the concept of "relevance for a broad public". We saw possible reasons for disagreement between experts, for example the perspective taken on the sentence and the role of time. An example sentence that created much discussion stated that a specific well-known professional footballer earns 300,000 Euros per year. While one expert said that it does not matter whether it is 300,000 or 400,000, another said that "this is a lot of money, and therefore relevant". In addition, the statement becomes more relevant in the context of an ongoing public debate about the ethics of professional athletes' salaries. In other words, factuality may be easier to determine based on linguistic features, while check-worthiness is intimately tied to an assessment of newsworthiness (an essential part of journalistic expertise) and gatekeeping by the media. These considerations, together with the observation that agreement between different raters was higher among experts

[10] See for example https://www.demorgen.be/tech-wetenschap/vlaanderen-bindt-met-technologie-de-strijd-tegen-nepnieuws-aan-b732defa, https://journalist.be/2019/04/factcheck-vlaanderen-van-start, https://www.svdj.nl/nieuws/robot-helpt-factchecken-nieuws-doorzoeken/, https://www.leidschdagblad.nl/cnt/dmf20180926_10120247/factcheckprogramma-komt-journalisten-te-hulp, https://www.unity.nu/Artikelen/leiden/vlaamse-subsidie-voor-factcheck-tool-universiteiten, https://www.villamedia.nl/artikel/vlaamse-subsidie-voor-factcheck-tool-universiteit-leiden-antwerpen-en-leuven

than among non-experts (or between these groups), point to the importance of deploying – as well as teaching and learning-by-doing – journalistic expertise about the 'more tangible' as well as the 'harder to grasp' aspects of fact-checking, and to the value of computational tools for reflexivity in and about this process.

The specific quality of FactRank derives from the interdisciplinary and iterative process in which it was created, which includes not only a well-performing deep-learning neural network architecture, but also a principled approach to defining and operationalising the concept of check-worthiness via a detailed codebook. This codebook was created jointly by expert fact-checkers from the two countries that have Dutch as an official language (Belgium/Flanders and the Netherlands). We expect FactRank to be very useful exactly because of the way we defined check-worthiness, and because of how we have made this explicit and traceable. One illustration of this value is the role of the codebook in an error discussion, as described in Section 6.

Another specific quality of FactRank is that it automatically monitors different sources on one integrated platform (Twitter, transcripts of parliaments, TV interviews, all of these across the two countries). Drawing on these sources may require specific contracts once FactRank moves from being a prototype to being a fully-fledged system, potentially run by a collaboration of public and commercial news organisations. We are currently negotiating these questions with relevant actors in Flanders and the Netherlands.

For all these reasons, we expect FactRank to become a significant source, used by professional journalists/fact-checkers in the Netherlands and Belgium, in practice, to monitor claims that can be the starting point of a fact-check article. Thus, FactRank strengthens the watchdog function of journalism. In addition, the tool can be used by civil society organizations or scholars interested in the (non-)factual nature of political discourse.

7.2 An outlook on future work

Our observations of website performance and the error discussion in Section 6 indicate several ways for improving the performance of the machine-learning model: features, data and learning schemes, language analysis, and leveraging human and machine intelligence in a combined way, supported by interdisciplinary research collaboration.

First, it is likely that performance will increase if/when we have more labelled data available for training, and this is a matter of resources. For example, the sentence labelling task lends itself to crowdsourcing, but requires crowdworkers who are proficient in Dutch, have a sound understanding of the current Flemish or Dutch political/media discourse, and are willing to study the codebook and exercise with the Reference Dataset. Also, such labelling would be an ongoing task, which raises practical and ethical issues such as funding and remuneration. Further ways forward include a transition to semi-supervised learning, which needs a smaller amount of labelled data than our current approach, an investigation of possible performance improvements by transfer learning as in (Konstantinovskiy et al., 2018), and the use of richer Dutch-language models that we are fine-tuning for specific tasks (Delobelle et al., 2020).

Second, the choice of features allows for many extensions. The surrounding sentences could be included when a statement is provided to the model, as this context might change the meaning of the statement at hand. For example, when a statement refers to a person, the statement may or may not be check-worthy depending on whether this person is of any significance. Metadata (e.g. about the speaker or other context) could be collected from data sources where applicable, or inferred, possibly by drawing on additional knowledge bases. (By our focus on elected politicians, our datasets have a built-in 'minimum relevance of the speaker'.) Informal user feedback from the FactRank website suggests that certainly the context of surrounding sentences could be helpful for improving the model, and metadata such as community engagement on social media have been used successfully in other tasks related to fact-checking (e.g. Zhang, Yilmaz, & Liang, 2018).

Third, we aim to further explore the role of words, sequence, and more complex linguistic information. As discussed in the Results section and also with reference to some of the examples in the Limitations section, our models are centered on words and (through the features) a limited form of word-sequence information, and based on the results so far, we have reason to believe that this may be a good choice. Beyond this, the model has no understanding of grammar or semantics, and it has no understanding of pragmatics. While we took care to instruct coders on important cases of sentences containing more than one claim, the examples in Section 6 illustrate that sentences containing more than one claim are difficult for the model to score on purely formal/syntactic grounds. Some grammatical operations that would allow for increased performance are easy for many sentences that contain more than one claim: the separation of sentences into main clauses, subordinate clauses, etc. On the other hand, solving the reverse problem - that a claim can extend over more than one sentence - would require more advanced linguistic processing (for example, coreference resolution may be necessary). In sum, a third way forward towards better performance could be to incorporate more linguistic and more semantic background knowledge.

Fourth, the tests, reflections, and discussions on the seeming or actual errors of the model led us to reconsider choices that we made in the construction of the codebook, and to develop plans for how to investigate the consequences of alternatives to these choices. This shows how the FactRank approach, with its iterations of human and machine labelling and learning, can help us continually improve an architecture for detecting check-worthy statements that leverages the strengths of both human fact-checkers and machine learning. The design space for these iterations is large, and it requires a careful weighing of what is useful theoretically and what is feasible under resource constraints in practice.

One example is the interesting question to what extent differences in time, medium or topic of the materials impede the classifier, and what can be generalised or transferred. An investigation of such factors would require controlled experiments based on a much larger pool of labellers and thus has to be deferred to later projects with additional funding.

Another example is the strategy for active learning. We used simple relevance feedback sampling strategies. From a machine-learning point of view, it appears likely that other strategies may lead to better performance of the classifier (e.g. uncertainty sampling, Lewis & Gale, 1994; version space reduction, Tong & Koller, 2001; importance weighting, Kreemer, Steenstrup Pedersen, & Igel, 2014). Some of these strategies require more complex computational architectures (e.g.

ensembles), and they may involve presenting users with more "uninteresting" sentences. This is not a problem when users are paid labellers (such as those in our phases 2, 3 and 6), but it may drain too much energy from users in real deployment settings (such as those in our phase 9). For the latter, also Lewis and Gale (1994, 3) observe that " 'relevance sampling' [= relevance feedback] is a reasonable strategy in a [...] context [...] where the user is more interested in seeing relevant texts than in the effectiveness of the final classifier produced". In line with the recent trend towards 'the simplest things that can possibly work' (Lin, 2019), we believe that design choices that are simpler and less resource intensive have many advantages in practical settings. In addition to different active-learning strategies, different strategies for quality control (such as how to evaluate and process inter-rater agreements with experts, Arslan et al., 2020) could be compared with respect to their contribution to labelling speed and quality.

A third example is user feedback. With the voting mechanism, we have made the first steps towards a live and ongoing interaction with professionals and the interested public. Our data-centric results indicate that such feedback may improve classification accuracy. However, the small size and scope of our test of this option make it difficult, at the moment, to assess the added value as a feature of the live website. Real-life deployment poses interesting further challenges, such as whose votes should affect the scores visible to everyone. Will these votes be generally helpful, or could they lead to over-personalisation? How can trolling be avoided? Which interface choices work best? Should these questions be explored in laboratory experiments or in live A/B testing (for which a large-enough user base is needed)? How to best integrate user feedback will thus be a relevant question of future work.

Last but not least, the real test of the usefulness of FactRank will come through professional fact-checkers trying it out in practice. Such tests will show whether the tool can indeed save them time and deliver check-worthy claims in an efficient manner. Currently, fact-checkers from, for instance, the weekly Knack and Flemish broadcaster VRT, already use FactRank. However, that use is not monitored, just as the user experience has not been investigated systematically yet. Future research in Belgium and the Netherlands should deliver insights concerning the use as well as the user experience of FactRank, to help improve on the present work.

Acknowledgements

The development of FactRank was made possible by the Vlaams Journalistiek Fonds (VJF), a project of the non-profit organisation journalismfund.eu in collaboration with the Flemish Association of Journalists (VVJ) and the Flemish Government, launched in 2018, under project number VJF_2018_012. We are grateful for the financial support from the VJF and for their continued interest and support. We also thank Brecht Laperre and Ivo Merchiers for their collaboration on the previous version and prototype of the tool and its concepts.

References

Allein L., & Moens M.-F. (2020) Checkworthiness in Automatic Claim Detection Models: Definitions and Analysis of Datasets. In: van Duijn M., Preuss M., Spaiser V., Takes F., Verberne S. (Eds) *Disinformation in Open Online Media. MISDOOM 2020.* LNCS 12259. Springer, Cham. https://doi.org/10.1007/978-3-030-61841-4_1

Arslan, F., Hassan, N., Li, C., & Tremayne, M. (2020). A Benchmark Dataset of Check-worthy Factual Claims. In *Proc. of ICWSM 2020.* Available at https://arxiv.org/pdf/2004.14425.pdf

Babakar, M. & Moy, W. (2016). *The State of Automated Factchecking. How to make factchecking dramatically more effective with technology we have now*. London: Full Fact. Available at https://fullfact.org/media/uploads/full_fact-the_state_of_automated_factchecking_aug_2016.pdf

Delobelle, P., Winters, T., & Berendt, B. (2020). RobBERT: a Dutch RoBERTa-based Language Model. *CoRR abs/2001.06286.* https://arxiv.org/abs/2001.06286

de Vries, W., van Cranenburgh, A., Bisazza, A., Caselli, T., van Noord, G., & Nissim, M. (2019). BERTje: A Dutch BERT model. *CoRR abs/1912.09582* https://arxiv.org/abs/1912.09582

Elsayed, T., Nakov, P., Barrón-Cedeño, A., Hasanain, M., Suwaileh, R., Da San Martino, G., & Atanasova, P. (2019). Overview of the CLEF-2019 CheckThat! Lab: Automatic Identification and Verification of Claims. In *International Conference of the Cross-Language Evaluation Forum for European Languages* (pp. 301-321). Cham, Switzerland: Springer.

Fu, Y. & Zhu, X., & Li, B. (2013). A survey on instance selection for active learning. *Kowledge and Information Systems, 35,* 249–283.

Graves, L. (2016). *Deciding what's true: The rise of political fact-checking in American journalism*. Columbia University Press.

Graves, L. (2018a). *Understanding the promise and limits of automated fact-checking.* Oxford: Reuters Institute for the Study of Journalism. Available at https://ora.ox.ac.uk/objects/uuid:f321ff43-05f0-4430-b978-f5f517b73b9b/download_file?file_format=pdf&safe_filename=graves_factsheet_180226%2BFINAL.pdf&type_of_work=Report

Graves, L. (2018b). Boundaries Not Drawn. *Journalism Studies, 19 (5),* 613-631.

Graves, L. & Amazeen, M.A. (2019). Fact-Checking as Idea and Practice in Journalism. *In Oxford Research Encyclopedia of Communication* (oxfordre.com/communication). Oxford University Press.

Graves, L., & Cherubini, F. (2016). *The rise of fact-checking sites in Europe*. Oxford: Reuters Institute for the Study of Journalism.

Hall Jamieson, K. (2018). *Cyberwar: How Russian Hackers and Trolls Helped Elect a President What We Don't, Can't, and Do Know*. Oxford University Press.

Hansen, C., Hansen, C., Alstrup, S., Grue Simonsen, J., & Lioma, C. (2019, May). Neural Check-Worthiness Ranking with Weak Supervision: Finding Sentences for Fact-Checking. In *Companion Proceedings of The 2019 World Wide Web Conference* (pp. 994-1000). ACM.

Hassan, N., Arslan, F., Li, C., & Tremayne, M. (2017a). 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* (pp. 1803-1812). ACM.

Hassan, N., Zhang, G., Arslan, F., Caraballo, J., Jimenez, D., Gawsane, S., Hasan, S., Joseph, M., Kulkarni, A., Nayak, A.K., Sable, V., Li, C., & Tremayne, M. (2017b). ClaimBuster: the first-ever end-to-end fact-checking system. *Proceedings of the VLDB Endowment*, *10*(12), 1945-1948.

High level Group on fake news and online disinformation (2018). *A multi-dimensional approach to disinformation: Report of the independent high level group on fake news and online disinformation*. Publications Office of the European Union. https://ec.europa.eu/digital-single-market/en/news/final-report-high-level-expert-group-fake-news-and-online-disinformation

Jarada, I., Gencheva, P., Barrón-Cedeno, A., Márquez, L., & Nakov, P. (2018). 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* (pp. 26-30). New Orleans, Louisiana: Association for Computational Linguistics.

Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 1746-1751). Doha, Qatar: Association for Computational Linguistics.

Konstantinovskiy, L., Price, O., Babakar, M., & Zubiaga, A. (2018). Towards automated factchecking: Developing an annotation schema and benchmark for consistent automated claim detection. To appear (2020) in *ACM Digital Threats, Research and Practice*. Preprint at https://arxiv.org/abs/1809.08193, http://www.zubiaga.org/publications/files/dtrap2020-claim-detection.pdf

Kremer, J., Steenstrup Pedersen, K., & Igel, C. (2014). Active learning with support vector machines. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 4(4):* 313-326.

Kwan, v. (2019). *Responsible Reportingin an Age of Information Disorder.* First Draft News. https://firstdraftnews.org/wp-content/uploads/2019/10/Responsible_Reporting_Digital_AW-1.pdf

Laperre, B., Merchiers, I., & Hautekiet, R. (2018). *FactRank: Automated Check-Worthiness Detection in Dutch.* Term paper, KU Leuven, Belgium. https://people.cs.kuleuven.be/~bettina.berendt/FactRank/LaperreEtAl2018.pdf

Leblay, J., Manolescu, I., & Tannier, X. (2018). Computational fact-checking: Problems, state of the art, and perspectives. Tutorial at *The Web Conference*, Apr 2018, Lyon, France. 2018, The Web Conference 2018. https://hal.inria.fr/hal-01791232

Lewis, D. & Gale, W. (1994). A sequential algorithm for training text classifiers. In *Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval* (pp. 3–12). Springer.

Lin, J. (2019). The simplest thing that can possibly work: Pseudo-relevance feedback using text classification. *CoRR abs/1904.08861* https://arxiv.org/abs/1904.08861

Schäfer, R. & Bildhauer, F. (2012). Building Large Corpora from the Web Using a New Efficient Tool Chain. In *LREC* (pp. 486-493).

Thorne, J., & Vlachos, A. (2018). Automated fact checking: Task formulations, methods and future directions. In *Proc. of the 27th Int. Conference on Computational Linguistics* (pp. 3346–3359). ACL.

Tong, S. & Koller, D. (2001). Support Vector Machine active learning with applications to text classification. *Journal of Machine Learning Research, 2:* 45-66.

Tulkens, S., Emmery, C., & Daelemans, W. (2016). Evaluating Unsupervised Dutch Word Embeddings as a Linguistic Resource. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016).* Paris, France: European Language Resources Association (ELRA). Data available at https://github.com/clips/dutchembeddings

Wardle, C., & Derakhshan, H. (2017). *Information Disorder: Toward an interdisciplinary framework for research and policy making*. Council of Europe Report, 27.

Yu, H., & Hatzivassiloglou, V. (2003). Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 129–136.

Zhang, Q., Yilmaz, E., & Liang, S. (2018). Ranking-based Method for News Stance Detection. In *Companion Proceedings of the Web Conference 2019* (pp. 41-42), ACM.