Information Processing and Management 59 (2022) 103105

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/ipm

Automated monitoring of online news accuracy with change classification models☆

Yoram Timmerman *, Antoon Bronselaer

Department of Telecommunications and Information Processing, Ghent University, Sint-Pietersnieuwstraat 41, 9000 Ghent, Belgium

ARTICLE INFO

Keywords:
News accuracy
Liquid news
Error corrections
Supervised machine learning

ABSTRACT

In the past decade, news consumption has shifted from printed news media to online alternatives. Although these come with advantages, online news poses challenges as well. Notable here is the increased competition between online newspapers and other online news providers to attract readers. Hereby, speed is often favored over quality. As a consequence, the need for new tools to monitor online news accuracy has grown. In this work, a fundamentally new and automated procedure for the monitoring of online news accuracy is proposed. The approach relies on the fact that online news articles are often updated after initial publication, thereby also correcting errors. Automated observation of the changes being made to online articles and detection of the errors that are corrected may offer useful insights concerning news accuracy. The potential of the presented automated error correction detection model is illustrated by building supervised classification models for the detection of objective, subjective and linguistic errors in online news updates respectively. The models are built using a large news update data set being collected during two consecutive years for six different Flemish online newspapers. A subset of 21,129 changes is then annotated using a combination of automated and human annotation via an online annotation platform. Finally, manually crafted features and text embeddings obtained by four different language models (TF-IDF, word2vec, BERTje and SBERT) are fed to three supervised machine learning algorithms (logistic regression, support vector machines and decision trees) and performance of the obtained models is subsequently evaluated. Results indicate that small differences in performance exist between the different learning algorithms and language models. Using the best-performing models, F_2-scores of 0.45, 0.25 and 0.80 are obtained for the classification of objective, subjective and linguistic errors respectively.

1. Introduction

Digital data are playing an increasingly important role in almost any aspect of modern society. Indeed, the quality of many organizational and economical processes is impacted directly by the quality of the data that are used. As a consequence, the topic of data and information quality has received a lot of attention in the last couple of decades, both in research and industry (Timmerman & Bronselaer, 2019). An important industry that is almost synonymous with information quality is the news industry. Indeed, it is of uttermost importance and at the fundamental core of good journalistic work to deliver accurate information to news consumers (Porlezza, 2019).

☆ This document is the result of the research project funded by the Bijzonder Onderzoeksfonds (BOF) of Ghent University, Belgium [grant number 01D19919].
* Corresponding author.
E-mail address: yoram.timmerman@ugent.be (Y. Timmerman).

https://doi.org/10.1016/j.ipm.2022.103105
Received 5 July 2022; Received in revised form 26 September 2022; Accepted 27 September 2022

However, similarly to other industries, the news industry has been significantly impacted by digitization. In the last two decades, textual news consumption has been shifted from mainly reading news on paper to mainly consulting news via online platforms. These include online versions of printed newspapers and new, online-exclusive newspapers, but blogs and social media are also very important types of platforms through which (younger) people consume news on a daily basis. According to the most recent Digital News Report, published by Reuters Institute, online channels are a news source for 78% of the Flemish population, compared to only 32% of the Flemish people consuming news via printed media. In comparison with five years ago, this consumption rate has remained almost stable for online news (82% in 2016 versus 78% in 2021), while a significant decline in the consumption of printed media has been observed (45% in 2016 versus only 32% in 2021) (Newman et al., 2021).

It is clear that the digitization of news consumption has come with several advantages. These include the possibility for news consumers to interact with journalists and even create their own news, the possibility to receive news 24/7 at a higher pace and the possibility to access different newspapers at a very cheap cost (Nguyen, 2010). On the other hand, the shift to online news leaves us with a number of challenges as well, not in the least with regards to news accuracy. The most well-known problem in this regard is the dissemination of fake news, misinformation and disinformation (Nakov et al., 2021). As it is possible for anyone to publish online content on The World Wide Web, social media and untrustworthy websites and blogs are used on a large scale to (intentionally) spread false information to large amounts of people (Lazer et al., 2018; Vosoughi, Roy, & Aral, 2018). Notwithstanding the importance and detrimental effects of the presence of fake news on social media, it cannot be forgotten that high online news quality should be guaranteed by traditional online newspapers as well in order to preserve news quality on the internet. In this way, it is not only guaranteed that people consuming news via traditional online news websites are reading legitimate and high quality news, but also the extent to which people tend to search for alternative (possibly illegitimate) news on social media or other platforms is minimized. Indeed, research indicates that one of the common characteristics of people trusting, reading and sharing fake news is their distrust in traditional media (Pérez-Escoda, Pedrero-Esteban, Rubio-Romero, & Jiménez-Narros, 2021; Tsfati et al., 2020). Moreover, it has been proven that the trust level of people in the media is negatively correlated with error perceptions of these media by news consumers (Wilner, Wallace, Lacasa-Mas, & Goldstein, 2021). Taking both observations together, preventing people from reading, believing and spreading fake news starts by providing high quality and accurate news on traditional (online) media platforms.

Despite its importance, the accuracy of online newspapers is under pressure as well. Literature suggests several possible reasons to explain this phenomenon. First, because of the fact that news consumers can easily consume several online newspapers or social media platforms at a time without any additional cost (as compared to news consumers having a fixed subscription to a newspaper in the printed alternative), competition between different news producers has never been higher (Burggraaff & Trilling, 2020). This has lead to the commercialization of news, in which economic considerations are taken into account in the daily work of journalists (Tandoc, 2014). As an example, website metrics are used in order to monitor which articles are accessed a lot by news consumers and which are not. Based on such analyses, news articles can be (re)written such that audience is maximized (Burggraaff & Trilling, 2020). It is not difficult to see that such strategies may have detrimental effects on information quality, e.g. by making changes in the process that decides on which information to publish or not (Anderson, 2011; Lee, Lewis, & Powers, 2014; Usher, 2018; Welbers, Van Atteveldt, Kleinnijenhuis, Ruigrok, & Schaper, 2016). Secondly, competition between online newspapers is growing even further because of more stringent time constraints. While printed media typically work with a fixed daily deadline, online media continuously publish news. This leads to a competition in terms of who is capable to report on a given news story first (Burggraaff & Trilling, 2020). Accordingly, less attention is often given to quality-management procedures. This leads to many articles "being published first and checked for accuracy later" (Porlezza, 2019). The journalism model "that puts the highest value on accuracy and context" has been replaced by "a newer model that puts the highest value on immediacy and volume" (Brautović et al., 2021; Kovach & Rosenstiel, 2011).

In the past years, several fact-checking initiatives have been launched in order to monitor online news accuracy and detect (or even correct) fake news present on the web. Moreover, a lot of research has also been performed in developing automated systems for fact-checking (Lazer et al., 2018; Li et al., 2016; Shu, Sliva, Wang, Tang, & Liu, 2017; Vo & Lee, 2018; Vosoughi et al., 2018). Many of these systems primarily focus on the retrieval of fake news on social media. Nonetheless, in essence most of them can be used for fact-checking in general. Most of the existing tools have proven their usefulness when used in combination with the expertise of professional fact-checkers. However, the fully automated discovery and checking of claims has not been realized yet. An important reason for this is the difficulty associated with the last step of the fact-checking pipeline: automated claim verification (Nakov et al., 2021). The challenges that accompany this process have been well-documented recently (Arnold, 2020). Explainable automated claim verification typically requires some kind of "fundamental truth" to verify the input claim against. In the past, researchers have approached this problem in several ways: by using tables or databases to check information against Chen et al. (2020) and Karagiannis, Saeed, Papotti, and Trummer (2020), by fact-checking claims against knowledge retrieved from Wikipedia (Nie, Chen, & Bansal, 2019; Thorne, Vlachos, Cocarascu, Christodoulopoulos and Mittal, 2018), or by using inference over a knowledge graph (Gad-Elrab, Stepanova, Urbani, & Weikum, 2019). Some of these models may deliver good results in specific contexts. However, it is highly improbable that it is possible to check every factual claim in every news article against an official record (Porlezza, 2019).

In this paper, we present a radically different strategy that is capable of detecting false statements in an automated way. Moreover, our approach does not require an external fundamental truth. The key idea of our approach is that online news is *liquid* (Widholm, 2016). Contrary to printed news articles, the content of online news articles can be changed after original publication. This leaves online journalists with a lot of options. For example, new information can be added to an earlier published article, old information can be updated or multimedia or hyperlinks can be added (Karlsson, 2012). However, changes to online news articles can also serve the purpose of correcting errors that were present in the original version of the article. Earlier research

has already indicated that such error corrections in online news are performed on a regular basis (Brautovic, Maštrapa, & John, 2020; Brautović et al., 2021; Hettinga & Smith, 2021; Karlsson, 2012; Karlsson, Clerwall, & Nord, 2017).

We hypothesize in this paper that corrections applied to online news articles can be used to collect information about the quality of the news provider. Instead of using an external "fundamental truth", inaccurate claims in the article can be detected by monitoring the changes made to that article. Indeed, the correction of an error can be seen as a signal that there was a problem with the accuracy of the original version of the article. As an example, assume that a news article is reporting on a terrorist attack, stating that *at least five people were killed*. Moreover, no source is quoted in the article confirming the statement. Subsequently, a correction is made to the original content of that article. This correction replaces the previously mentioned sentence by *the police confirmed that two people were killed*. It is then clear that (part of) the information in the original article was inaccurate.

Our approach is thus based on tracking changes that are made to online news articles over time and distinguishing error changes from other change types. As such, one is capable of acquiring a lot of information about the accuracy of online news articles in an automated way. The proposed procedure requires that online journalists extensively use the error correction functionality to increase the accuracy of articles. Earlier research has indicated that this is indeed the case (Forde, Gutsche, & Pinto, 2022; Kutz & Herring, 2005; Saltzis, 2012). As such, the proposed procedure could be very useful to professional fact-checkers as an additional tool for monitoring accuracy of online newspapers. The practical implementation of such an approach requires three consecutive steps:

- The automated monitoring and capturing of the changes made to news articles that are published by the online newspapers of interest.
- The identification of all *atomic changes* within the transition from an original article version to an updated version. Sometimes multiple changes can be made by a journalist in the same update (e.g., the correction of a linguistic error and the addition of new information). It should then be identified automatically which of the changed text parts in the original and new version belong to the same atomic change (e.g., which changed text parts belong to the correction of the linguistic error and which ones belong to the addition of information).
- The type of each identified atomic change should be determined automatically. The most important part here is to be able to distinguish error corrections from other types of atomic changes.

In this work, solutions are provided for the (partial) automation of the first and third part of the presented pipeline. Focus is hereby dedicated to online news articles written by Flemish news websites (and, as such, articles written in Dutch). The main reason for this is that the amount of research concerning Flemish news quality is very limited. Moreover, as will be evident from the literature overview in Section 2, research concerning automated tools for the verification of online Flemish news accuracy is almost non-existent. However, our approach can be easily adopted and implemented for news published in different countries and languages.

In this work, a large amount of published online news articles is obtained from six important Flemish online newspapers. Therefore, software was written using the Selenium package in Python.[1] Using this software, all online articles were visited automatically on a frequent basis (every 15 min) in the 24 h after their initial publication. During each article visit, it was verified whether updates were performed to the previously stored article version or not. As such, a large data set consisting of online news articles and the changes made to these articles was obtained. An online platform was then developed on which users could identify the atomic changes within the news updates, and indicate the type of these atomic changes. The finally obtained data set consists of 21,129 annotated atomic changes. Using these data, predictive models are constructed that are capable of distinguishing the correction of three error types (objective errors, subjective errors and linguistic errors) from other types of changes in Flemish online news articles. The results obtained by three different learning models (logistic regression, decision trees and support vector machines) and four different language models (TF-IDF, word2vec (Mikolov, Chen, Corrado, & Dean, 2013), BERTje (de Vries et al., 2019) and SBERT (Reimers & Gurevych, 2019)) are compared.

We believe our tool can be useful for the monitoring and analysis of news accuracy of online newspapers over time. The model can also be used in order to help researchers and professional fact-checkers with detailed analyses with regards to online news accuracy. These may include analyses on how many errors are corrected by journalists, which type of errors are made, how the number of corrections changes over time or how different online newspapers compare to each other in terms of accuracy?

The main results and contributions of this work are the following:

- To our knowledge, this is the first paper that proposes a model to evaluate and monitor online news accuracy by exploiting information that is present in the changes that are made to articles. The proposed approach circumvents the need for an external "fundamental truth" when evaluating news or claim accuracy.
- Using the help of scientific volunteers and our own manual annotations, an annotated data set is constructed. The data set consists of 21,129 atomic changes that were made to online articles published by six important Flemish online newspapers.
- Three predictive models are constructed that are capable of detecting corrections of objective, subjective and linguistic errors in news articles written in Dutch.

[1] https://selenium-python.readthedocs.io/.

The remainder of this paper is organized as follows. In Section 2, an overview of the related work is given. Section 3 describes the methodology, including the gathering and annotation of the data, measures taken in order to guarantee data quality and creation of the error-detection models. Section 4 then presents the results obtained by the models, whereas Section 5 analyzes and discusses them. Finally, conclusions, theoretical and practical implications and future work are discussed in Section 6.

2. Related work

2.1. News accuracy

Accuracy is a long-lasting research topic in journalism studies, the first study being published in 1936 (Charnley, 1936). In this fundamental paper, news accuracy of three local newspapers was investigated. This was done manually by asking sources that were mentioned in news articles to look for errors. Several types of errors were investigated, such as errors in meaning, errors in names, errors in titles ... In total, 591 local news stories were analyzed of which 46% contained at least one error. Charnley's approach has formed the basis for most of the news accuracy research that has been performed since then (Berry, 1967; Blankenburg, 1970; Brown, 1965; Marshall, 1977). Conceptually, later studies used a methodology very similar to the approach taken by Charnley, but Berry (1967) was the first to make a clear distinction between *factual* (objective) errors and *subjective* errors. Objective errors are clearly wrong because of the fact that non-debatable errors are present. Contrary, subjective errors are more subtle errors in which misleading or ideologically phrased information is present that may still be (partially) correct. More recently, news accuracy studies following Charnley's approach have also been performed in other countries than the United States of America, including Ireland, Switzerland and Italy (Fox, Knowlton, Maguire, & Trench, 2009; Porlezza, Maier, & Russ-Mohl, 2012). Measured error rates were also high in these countries: 60% in Switzerland, 54% in Ireland and 52% in Italy.

These studies are very useful and deliver a detailed image of news accuracy internationally. However, it is clear that a number of disadvantages are inherently part of the used methodology. First, as news accuracy is measured by consulting sources mentioned in the article, subjectivity is imported into the measurement procedure (Porlezza, 2019). Alternative models were proposed in which accuracy is measured by comparing all claims in a news article with an official record (Kocher & Shaw, 1981). However, as already discussed in Section 1, it is practically infeasible to obtain a "fundamental truth" containing a record for each possible claim. Second, the aforementioned studies are quite limited in scale. The reason fir this is that the procedure of contacting news sources and discussing accuracy is very time-consuming. This makes large scale accuracy studies less relevant in the context of a very rapidly evolving online environment. Contrary, the approach proposed in this work requires no news sources and allows for the (semi)-automated analysis of large amounts of news articles.

2.2. Updates and error corrections in online news

Only a moderate amount of studies have been performed with regards to how and how often online news articles are changed after initial publication. An important study was performed by Saltzis (2012), in which articles about 44 breaking news topics, published by six news websites in the United Kingdom, were monitored. It was discovered that the update functionality is regularly used, especially in the first few hours after initial publication. Consequently, the largest part of updating effort was devoted to adding new information to the articles. As expected, error corrections were only found in a minority of news updates. Nonetheless, in 26% of the updates a correction of an error was detected. This supports the claim that the online update functionality is also used for error correction.

Recently, it was found that some changes to online news are more subtle than simply correcting factual errors or adding information (Forde et al., 2022). Forde et al. performed a study in which 48 news updates of six *New York Times* online articles were analyzed. It was illustrated that through several updates, journalists sometimes change the explanation and meaning that is given to facts that are presented in the article. The authors refer to this phenomenon as "ideological corrections". In the use case of protests in Portland in July 2020 for example, it was shown that through different consecutive updates, the role of law enforcement was originally framed to be virtuous. However, after the updates were performed, the role of law enforcement was framed as being violent. Contrary, exactly the opposite happened for the way in which the actions by the protesters were perceived by the media. Although no clear errors were corrected when performing the updates, it is clear that the meaning and message of the articles has shifted. Meanwhile, readers were not informed about this. Finally, some studies investigated the extent to which readers are informed of the fact that errors were corrected and compared common practices with how readers actually should be informed (Appelman & Hettinga, 2021; Brautović et al., 2021; Karlsson et al., 2017). These studies indicate that corrections are often not noticed by news consumers. The reason for this is that journalists most often do not mention the fact that errors were corrected.

The studies mentioned above all have in common that the data collection and analysis was performed manually. As a consequence, the amount of data that was analyzed was minimal. Moreover, a lot of updates were missed because of limits to the frequency with which article URLs could be visited. Finally, because of the limited amount of data being investigated, the conclusions of such small scale studies can also be strongly biased. The reason for this is that the newspapers and articles under consideration are typically not selected randomly. Therefore, a good alternative is to automate the monitoring of news updates. However, only very few relevant studies were found in this context. Kutz and Herring performed a study in 2005 in which changes to the titles and introductory texts of the headline articles on the front page of three online newspapers were tracked (Kutz & Herring, 2005). In order to detect all changes made to these headline stories, scraping software was developed that was capable of looking for changes with a frequency of one minute. In total, this was performed for three consecutive weeks, analyzing 185

headline stories. Although their approach is comparable to the way in which our data collection process was performed, a couple of important remarks should be made. First, only the headline stories of each online newspaper were examined. However, they only form a minor fraction of all news produced and consumed on a daily basis. Secondly, only changes to the title, the introductory text or to the image accompanying the stories were monitored. Nonetheless, error corrections are performed extensively within the full text of an article as well. As such, their approach did not allow for getting a complete overview of the accuracy of the entire article. Finally, the study dates back to 2005. Since then, competition in online news has only been growing, and the importance of online news has increased significantly. The conclusions of this study can thus not be transferred easily to today's context. In 2008, a new approach, called Regular Interval Content Capture (RICC), was introduced in order to automatically visit URLs containing news articles multiple times. Subsequently, the content of the articles was captured by making a screenshot and storing it in PDF format. Multiple visits of the same URL were separated by a fixed time interval (Kautsky & Widholm, 2008). The approach was applied in order to analyze 64 h of online news produced by CNN.com. As the analysis of the gathered PDF documents had to be performed manually, the scale of the study was still very limited. A last notable study was performed more recently by Zamith (2017). In contrast to the aforementioned studies, Zamith proposed and applies an automated approach for the gathering of a very large amount of news articles snapshots. In total, 125,000 article versions were gathered, written by 21 news organizations. This was achieved by writing Python code implementing the RICC approach mentioned above with an interval of 15 min between subsequent visits of the same news article. All HTML pages corresponding to the different articles were subsequently downloaded and stored locally. Although this approach sounds very promising, until now, to our knowledge, no practical application of the RICC implementation has been used to actually investigate online news accuracy. As such, to the best of our knowledge, until now no studies have been performed in which an automated error correction-based approach such as ours is proposed for online news accuracy monitoring.

2.3. Automated fact-checking tools

As already mentioned in Section 1, a lot of diverging research has already been performed with regards to fake news and automated fact checking. It is not always clear what is meant by "automated fact checking", as it is a term encompassing a lot of different problems. Two interesting and complete overviews regarding automated fact checking are given by Nakov et al. (2021) and Thorne and Vlachos (2018). As Nakov et al. (2021) indicate, the process of automated fact checking consists of several intermediate steps. These are finding claims that are worth fact-checking, detecting previously fact-checked claims, retrieving evidence that is relevant for a particular claim, and finally automatically verifying the claim. Thorne and Vlachos (2018) on the other hand make another distinction between research initiatives. This distinction is based on the inputs that the task at hand considers, the sources of evidence that are used and the outputs that are returned by the constructed models.

An important property of automated fact checking tools is how and to which extent external evidence is used in order to verify whether a claim is correct or not. Several approaches were taken to this end. A first group of approaches does not use any external evidence in order to automatically assess claim validity. Here, the veracity assessment is done by using e.g. linguistic features of the text containing the claim or features regarding the source of the information. Examples include Lee et al. (2020), Rashkin, Choi, Jang, Volkova, and Choi (2017) and Wang (2017). However, most approaches use some kind of external evidence when trying to assess the veracity of a claim. One possible way to do this is by comparing triple inputs representing claims against information present in semi-structured knowledge bases (Bollacker, Evans, Paritosh, Sturge, & Taylor, 2008; Gad-Elrab et al., 2019; Thorne & Vlachos, 2017; Vlachos & Riedel, 2015). Claims can also be verified against structured information that is stored in tables or in databases (Ahmadi, Lee, Papotti, & Saeed, 2019; Chen et al., 2020; Karagiannis et al., 2020). Another approach is to use unstructured, textual data as evidence input in order to verify claim accuracy. This can be done by providing the evidence in short, textual format (Pomerleau & Rao, 2017), by needing to acquire evidence from multiple textual documents (e.g. Wikipedia pages) (Nie et al., 2019; Thorne, Vlachos, Christodoulopoulos and Mittal, 2018) or by acquiring text-based evidence from previously fact-checked claims (Hassan et al., 2017; Shaar, Babulkov, Da San Martino, & Nakov, 2020). Finally, it should be noted that all of the mentioned tools and frameworks are created for the English language. In the case of the Dutch language, to the best of our knowledge, only one relevant tool exists. FactRank is a tool that is capable of assessing the extent to which it is useful and relevant to verify a given claim within a larger Dutch text (Berendt et al., 2021). However, the tool does not say anything about the truthfulness of claims.

In conclusion, a number of tools exist that help professional fact-checkers with automated verification of claims. These tools typically rely on external evidence that in itself may not be trustworthy. Even worse, it might even not exist (Nakov et al., 2021). In the context of online news, with new information reported at very high speed, it is very unlikely that it will be possible to instantaneously find evidence for claim verification. In addition, almost no tools for the support of automated fact checking exist for the Dutch language. This adds additional proof for the necessity of the approach and models presented in this work.

3. Methodology

In this section, the methodology used to automate (part of) the process of detecting error corrections in Flemish online news articles, is explained. As mentioned in Section 1, the approach presented in this paper consists of three steps: (1) automated capturing of updates made to online news articles, (2) distinguishing different atomic changes within the same update, and (3) automatically annotating each atomic change with its type. In the following, Section 3.1 illustrates how step one of the pipeline (collecting data to monitor news accuracy) was performed in an automated way. In Section 3.2, steps two and three of the pipeline are then performed manually in order to obtain a high quality data set consisting of individual, annotated atomic changes. Finally, using the data set

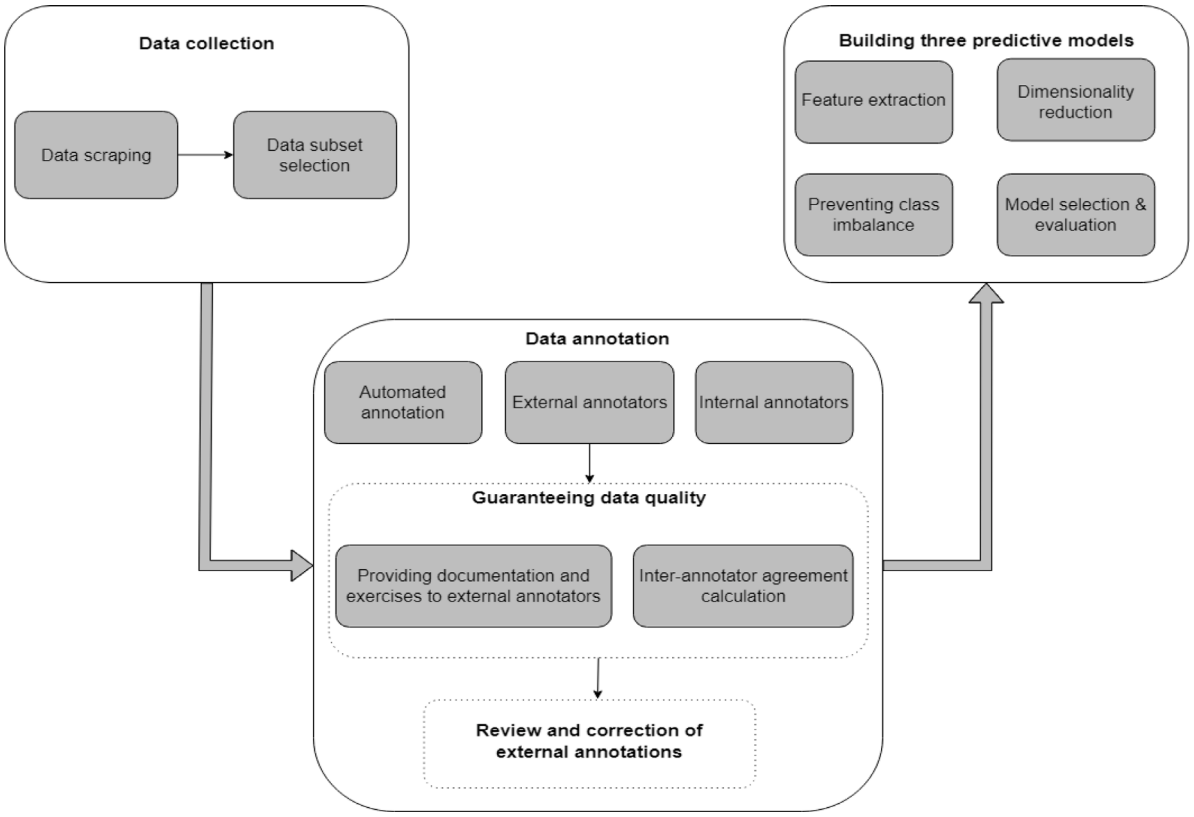

Fig. 1. Schematic overview to make the methodological approach taken throughout this project clear. The methodology consists of a data collection, data annotation and model building phase. Data quality and, thus, model quality were monitored and preserved during each step.

obtained by manual annotation, in Section 3.3 supervised models are created for the automated detection of error corrections in the set of atomic changes (i.e., the automation of step three of the pipeline). A schematic overview of the different steps that constitute the methodology is given in Fig. 1. All code concerning the automated capturing of the updates and the model building and a link to the raw and processed data set can be found on a public GitHub repository.[2]

3.1. Data collection

Online news articles under consideration were selected from six online newspapers: Het Laatste Nieuws,[3] Het Nieuwsblad,[4] VRT NWS,[5] De Standaard,[6] De Morgen[7] and Knack.[8] Newspapers were selected based on three criteria. First, each of these six online newspapers were amongst the most consulted online newspapers in Flanders at the beginning of the data collection process in 2019 (respectively being first, second, third, fifth, seventh and eleventh in the ranking of most consulted Flemish online newspapers) (Newman, Fletcher, Kalogeropoulos, & Nielsen, 2019). Second, each of them was primarily focusing on textual content, which is the primary target of our research as well. Finally, they all handle a broad scope of topics, thereby avoiding newspapers that have a very narrow-scoped focus such as sports- or economy-only. In total, fifteen topic categories for the news articles in the data set can be distinguished. An overview of their prevalence is given in Table 1.

In order to obtain a data set containing articles published by these six online newspapers, scraping software was developed in Python. For each of the newspapers, this scraper composed a list of the most recently published news articles using information present on the newspaper websites. When a new article was detected in this list, the web page containing that new article was visited using Selenium and relevant data (i.e., URL, title, introductory text, full text and publication time) were downloaded using

[2] https://github.com/yytimmer/newstracker_model.
[3] https://www.hln.be/.
[4] https://www.nieuwsblad.be/.
[5] https://www.vrt.be/vrtnws/nl/.
[6] https://www.standaard.be/.
[7] https://www.demorgen.be/.
[8] https://www.knack.be/.

Table 1
Overview of the number of articles (and their relative share) for each of the different topics. The most important news topics (accounting for more than 75% of all articles) present in the data set are sports, international news, domestic news and showbizz.

Topic	Number of articles (percentage)
Sports	2877 (25.22%)
International news	2495 (21.85%)
Domestic news	2292 (20.08%)
Showbizz	1325 (11.64%)
Local news	655 (5.75%)
Economy	497 (4.44%)
Lifestyle	461 (4.05%)
Science	317 (2.78%)
Tech	146 (1.28%)
Miscellaneous	89 (0.78%)
Car	86 (0.75%)
Factchecking	69 (0.61%)
Weather forecast	58 (0.51%)
Video & podcast	22 (0.19%)
Tips & tricks	8 (0.07%)

XPath. Finally, the retrieved data were stored in a relational database. In the 24 h following the publication of the article, this URL was visited on a very frequent basis. During each visit, the content of the article was compared with the content of the article that was present in the database. If changes were detected in comparison to the most recent version of the article, an additional version of the article was added to the database containing all data of the new version. As such, for each article that was retrieved from one of the six news websites, one or multiple article versions are present in the database. Moreover, the database contains information on the time periods during which each of these article versions were available online. The frequency with which URLs were revisited in order to detect new updates varies for different articles, newspapers and moments in time. This visiting frequency depends on the amount of articles the scraping software needed to visit at that time. Moreover, at all time we prevented that our scraping processes would be a burden on the normal functioning of the online news websites. However, as on average each URL was visited every 15 min, it is clear that the updates that were obtained were fine-grained enough for our purpose.

The scraping process was performed for two years in total, starting on April 1st, 2019 until March 31st, 2021, thereby obtaining 849,681 article versions belonging to 343,134 unique article URLs. It was clear that the data set was too large in order to be manually handled and annotated. However, a couple of remarks should be made. First, some online newspapers published the same article multiple times on pages with different URLs. As such, some articles (and corresponding updating patterns) were present multiple times in the obtained data set. After gathering the data, these articles were detected and deduplicated automatically. Second, a non-negligible part of the articles were liveblogs. Liveblogging is a relatively new trend in which new information and updates are added almost in real-time to an online web page that gathers all information with respect to a current topic (O'Mahony, 2014). The nature of liveblogs is however very different from the nature of a traditional online news article. Moreover, it is very clear to news consumers that the information in liveblogs will change over time (while this is not at all clear in the case of a standard online article). Therefore, it was chosen to leave liveblogs out of our current analysis, removing the corresponding article versions from the data set. In this way, it was made sure that the eventual predictive model was not impacted by the presence of this special type of articles. This left us with 485,553 article versions belonging to 291,826 unique articles.

These numbers were still too high in order to annotate all news updates manually. Further filtering of the data set was therefore required. In order to make sure that the final data set of selected online articles was not too large and still representative for an average day of online news publishing, a set of dates were chosen that were equally spread over the two years during which data were gathered. More specifically, for every period of two months, a time period of two consecutive days was chosen for which the amount of articles and article versions published was closest to the average daily amount of articles and article versions published during the entire two years. As such, we prevented that, by coincidence, our model would be trained on (and biased by) data that were captured during very specific circumstances, e.g., when an important breaking news event was taking place and getting all attention. Moreover, the selected time periods were also uniformly spread over the two years for which data are available. Further filtering the data as described above, we finally obtained a data set consisting of 18,952 article versions belonging to 11,389 unique online articles. Table 2 gives an overview of the number of article versions and articles gathered for every online newspaper and for all of the selected dates throughout the two years.

3.2. Data annotation

3.2.1. Annotation procedure

Following the data gathering process, the selected data set was annotated manually. More specifically, each couple of two subsequent versions of the same article (i.e., each news update) was analyzed. The annotation of a news update consists of two steps: (1) the detection of all *atomic changes* in the news update, and (2) the determination of the type of each of the identified atomic changes. Based on available literature (Berry, 1967; Charnley, 1936; Kutz & Herring, 2005; Saltzis, 2012) on the different types of atomic changes to online news and our own analysis of the obtained data, nine categories were distinguished (see Table 3).

Table 2
Overview of the number of article versions (and articles) for each online newspaper and each of the selected dates in the final data set. The number of articles and article versions are more or less equally spread over the different time periods for the same newspaper. The selected time periods reflect periods without significant deviations in the number of article versions as compared to global averages, and are chosen such that a uniform spread over time is obtained.

Time period	VRT	Knack	HLN	Het Nieuwsblad	De Morgen	De Standaard
09–10/04/2019	442 (159)	109 (69)	524 (356)	398 (279)	108 (67)	182 (99)
24–25/06/2019	412 (146)	58 (47)	553 (349)	392 (295)	108 (64)	194 (106)
20–21/08/2019	412 (150)	63 (46)	573 (400)	381 (291)	128 (88)	184 (101)
10–11/10/2019	480 (144)	89 (57)	579 (391)	351 (249)	139 (70)	210 (106)
03–04/12/2019	395 (157)	84 (59)	544 (330)	449 (301)	116 (69)	241 (116)
03–04/02/2020	437 (150)	95 (64)	527 (334)	343 (250)	112 (59)	158 (80)
27–28/04/2020	314 (150)	59 (53)	494 (325)	321 (234)	99 (69)	186 (80)
03–04/06/2020	323 (150)	66 (53)	544 (338)	319 (261)	111 (65)	159 (103)
25–26/08/2020	368 (151)	55 (35)	518 (280)	337 (261)	122 (64)	129 (85)
25–26/10/2020	235 (114)	66 (52)	320 (212)	304 (238)	75 (51)	140 (93)
06–07/12/2020	247 (121)	58 (52)	415 (293)	305 (237)	59 (42)	150 (92)
15–16/02/2021	294 (130)	84 (60)	482 (310)	367 (282)	98 (76)	149 (79)
Total	4359 (1722)	886 (647)	6083 (3918)	4267 (3178)	1275 (784)	2082 (1140)

Table 3
Overview of the different categories of atomic changes together with the definition of each of the categories as it was provided to the platform users.

Category	Description
Correction of an objective error	In the original version of the article an objective error is present, but this is not the case anymore in the new version of the article. Typical objective errors include wrong names, wrong function titles, wrong numbers, wrong locations, wrong dates, wrong citations …
Correction of a subjective error	In the original version of the article a subjective error is present, but this is not the case anymore in the new version of the article. Typical subjective errors include a non-representative (sub)title, sensationalized information, information that is presented too modest, essential information that is missing, misleading numbers, quotes that are taken out of the original context …
Correction of a linguistic error	In the original version of the article a spelling mistake or another linguistic (e.g. grammatical) error is present, but this is not the case anymore in the new version of the article.
Addition or clarification of information	Change in which information is added in comparison to the previous version of the article.
Deletion of information	Change in which information is deleted in comparison of the previous version of the article.
Update of information	Change in which old information is replaced by more recent information. This implies both the deletion of old information and the addition of new, corresponding information.
Content-neutral reformulation	Change in which information is described in another way, without old information being removed, new information being added or errors being corrected.
Displacement of information	Change in which a given piece of text that is present in the original version of an article is displaced to another place in the new version of that article.
Other	Change that cannot be categorized in any of the prior categorization options.

In total, 7563 news updates were present in the selected data set. To reduce the workload of the annotation process, two important measures were taken:

1. A part of the data set was annotated automatically. This was possible because of the fact that a significant amount of news updates consisted of only one, large atomic change. Here, we refer to atomic changes in which either a large amount of text was added to the article or a large amount of text was deleted from the article. As it is very easy to retrieve these kinds of news updates programmatically and categorize them as either "addition or clarification of information" or "deletion of information", no human intervention was needed for the annotation of these cases. Only additions and deletions in which entire paragraphs are added or deleted were annotated automatically. The reason for this is that the addition or deletion of only a limited amount of sentences (or even words) could in fact be another atomic change type than simply addition or deletion of information. Especially in the case of deletions it is necessary to take appropriate precautions, as errors could be deleted as well. In total, 2506 (33%) of the updates were annotated in this way. Almost all (i.e., 2454) of these updates contained an addition of information, only a very small minority of 52 updates consisted of the removal of a large amount of textual information. In order to be absolutely sure that no atomic changes are mistakenly labeled as "deletion of information", an additional manual check was performed for each of these 52 updates.
2. An online platform[9] was developed on which people could annotate article versions themselves 24/7. After creating an account, people were each time offered two subsequent versions of the same article (the original version on the left side of

[9] https://newstrack.ugent.be.

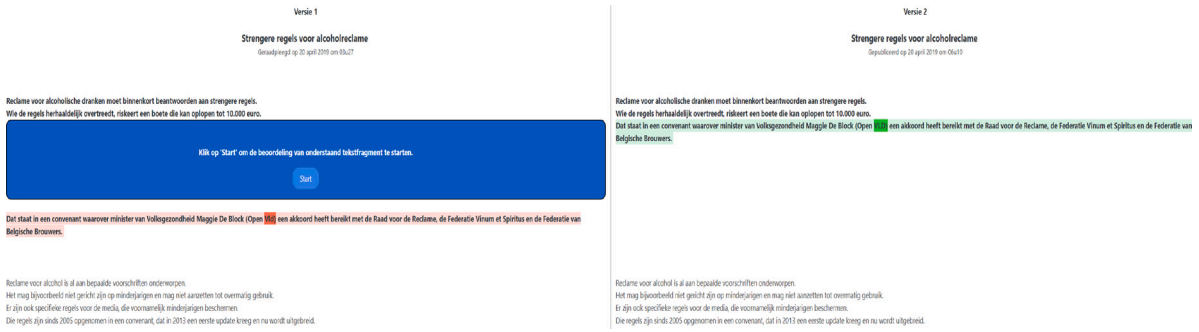

Fig. 2. Example screenshot of the Newstrack platform to illustrate how the platform works. The screenshot contains two subsequent versions of the same article published by VRT NWS. The original article version is shown on the left of the screen, the new article version is shown on the right. Changes made to the original article are marked in red, while textual replacements and additions necessary to obtain the new article are marked in green. Each atomic change is accompanied by a blue rectangle that annotators can click on to start the annotation of the atomic change. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 4
Overview of the number of atomic changes for each online newspaper and each of the selected dates in the final data set. The number of atomic changes varies significantly over different newspapers, as a result of large differences in the number of articles published. The number of atomic changes during different time periods for the same newspaper also varies more in comparison with the extent to which the number of published article versions varies (shown in Table 2).

Time period	VRT	Knack	HLN	Het Nieuwsblad	De Morgen	De Standaard
09–10/04/2019	636	168	425	357	91	254
24–25/06/2019	637	15	524	223	125	254
20–21/08/2019	642	46	445	332	129	224
10–11/10/2019	714	66	594	325	177	351
03–04/12/2019	518	49	641	577	185	309
03–04/02/2020	812	87	517	259	162	223
27–28/04/2020	343	8	498	303	70	409
03–04/06/2020	408	28	643	196	165	173
25–26/08/2020	617	44	795	211	161	127
25–26/10/2020	319	22	374	150	65	100
06–07/12/2020	267	8	565	166	33	225
15–16/02/2021	341	71	568	234	55	275
Total	6254	600	6601	3332	1418	2924

the screen, the new version on the right side of the screen, see Fig. 2). Following a procedure that was thoroughly explained and practiced before starting to annotate real data, users were capable of indicating all atomic changes within the given news update and the corresponding type of each of those atomic changes. In total, 1978 (26%) of the updates were annotated by 40 external annotators. The remaining 3079 (41%) of the news updates were annotated by the authors of this paper.

The methodology as described above, including the participation of human subjects, was approved by the ethics committee of the Faculty of Arts and Philosophy of Ghent University in advance. External annotators were acquired by consulting relatives and friends, by making promotion within university internally and by making promotion outside university through several online channels (website, social media...).

After all 7563 news updates were annotated, in total 23,028 unique atomic changes were detected. 1899 of these atomic changes were annotated with more than one category. These cases are indicative that annotators were convinced that one atomic change in fact still consisted of two types of changes. In order to limit the complexity of the task at hand, it was decided to keep the number of labels to predict at one. Therefore, the 1899 atomic changes containing multiple labels were removed from the data set leaving us with 21,129 single-labeled atomic changes. An overview of the number of identified atomic changes for each newspaper and for each selected time period is given in Table 4.

3.2.2. Annotation quality

Because external annotators were included in the annotation process, monitoring and guaranteeing annotation quality is an important factor for further processing. Several measures were taken to this extent:

1. Before being able to start the online annotation process, annotators were informed extensively. The online annotation platform contained information about the research goals, the annotation procedure, the types of atomic changes that exist... This information was available both textually and in the form of two tutorial videos.

2. After having consulted all introductory information present on the website, each annotator had to annotate atomic changes belonging to at least five news updates as an exercise. Only if annotators annotated the majority of atomic changes in the five news updates with the correct change type, they were allowed to start annotating real data. If not, annotators were given five additional exercises. This was repeated three times at most. As such, it was made sure that people that were less comfortable with the annotation process were capable of making 15 exercises before having to annotate real data.
3. In order to motivate external annotators to perform the annotation task rigorously, all participants that analyzed at least 50 news updates were given a cinema ticket. Depending on the exact number of news updates, some annotators were even given multiple cinema tickets for their work.
4. In order to estimate annotation reliability, in approximately 10% of the cases annotators were offered news updates that were already annotated by another annotator (without being aware of the type of annotation provided by the other annotator). As such, inter-annotator agreement could be estimated. In total, 441 news updates were annotated by two annotators instead of one.

Inter-annotator agreement was estimated using Fleiss' kappa (Fleiss, 1971). However, as annotators were not only responsible for determining the type of a given atomic change, but also for determining which text parts in the original and new article version were part of the atomic change, it was not trivial to estimate inter-annotator agreement.

As an example, consider the update of the original sentence *Belgium has 10 million residents* to the updated sentence *Belgium has 11 million inhabitants*. While transforming the original sentence into the new sentence, two atomic changes are applied: *10* becomes *11* (being a correction of an objective error), and *residents* become *inhabitants* (a content-neutral reformulation). An annotator could however annotate this news update by distinguishing only one atomic change of type "correction of an objective error". This atomic change then would contain both *10* and *residents* in the original article version and *11* and *inhabitants* in the new version. In assessing inter-annotator agreement, we would then be comparing the types of two atomic changes with the type of only one identified atomic change.

Therefore, inter-annotator agreement was determined by looking at the annotations of individual changed text parts instead of the annotations of atomic changes. Again considering the example on the number of inhabitants in Belgium presented above, this leads to four different changed text parts, each of them having two associated annotations. Indeed, *10* and *11* are then classified by both annotators as part of the correction of an objective error. On the other hand, *residents* and *inhabitants* are then classified by one annotator as being part of a content-neutral reformulation and by another author as part of the correction of an objective error. Using this method, 1621 annotated changed text parts were used in order to assess inter-annotator agreement using Fleiss' kappa. It showed that there was moderate agreement between the annotators' judgments, $\kappa = .464$ (95% CI, .436 to .492), $p < .0005$. These results indicate that there is moderate agreement amongst annotators' judgments. Moreover, the annotations performed by external annotators form only a minority in the available data set (16% of the atomic changes). Nonetheless, it is clear that in order to eventually obtain a good performing predictive model, the quality of the annotations by the external annotators was not good enough. Therefore, it was decided by the authors to review and correct all 3423 atomic changes annotated by external annotators. Reviewing and correcting all external annotations made clear that the initial annotation accuracy of the external annotators was 66%. As such, the quality and reliability of the annotated data set was significantly enhanced by the reviewing procedure. Despite the fact that all external annotations had to be reviewed (which was not planned in advance), the inclusion of external annotators in the data annotation process can still be considered to be useful. First, as it was already indicated by the annotators which text parts were part of which atomic changes, this review could be performed much faster compared to re-annotating all atomic changes by the authors themselves. This saved us a lot of time. Second, the external annotations were useful to estimate the complexity of the classification tasks at hand. The results of the external annotation review were used to compute the relative occurrences of true positives, true negatives, false positives and false negatives in the external annotations. As such, when analyzing model performance, task complexity was taken into account by considering the numbers obtained by human annotation.

3.3. Model building

Using the annotated data set, three supervised models were trained that were capable of detecting, respectively, (1) corrections of objective errors, (2) corrections of subjective errors and (3) corrections of linguistic errors. In the original data set, all records were labeled by their atomic change type (see overview in Table 3). In order to obtain labels that could be fed to a binary classifier, the type values were replaced by a boolean label indicating whether a record represented the correction of an error (objective, subjective or linguistic) or not. In the following, more details are given about how different aspects of building a supervised model were approached. More specifically, information is given on the processes of feature extraction, dimensionality reduction and early stopping, solving class imbalance issues, algorithm selection, hyperparameter tuning and model evaluation. All coding was performed in Python, making use of the scikit-learn package.[10]

[10] https://scikit-learn.org/stable/.

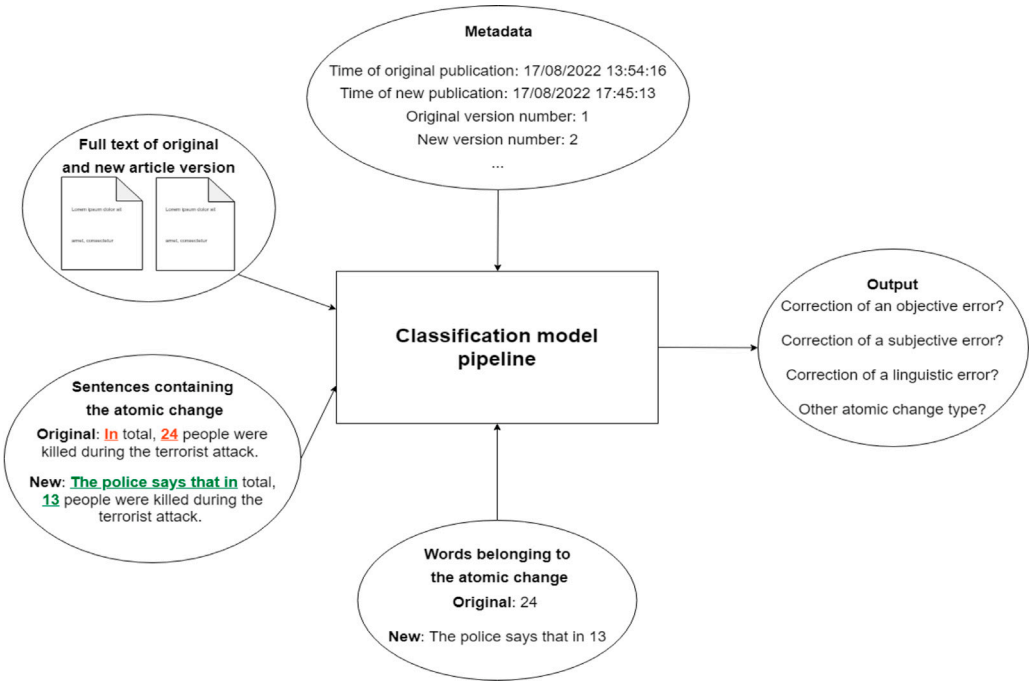

Fig. 3. Example values for the input that can be provided to the model pipeline. Input information consists of metadata, the full text of the original and new article versions, the sentences that contain the atomic change and the words that are part of the actual atomic change.

3.3.1. Feature extraction

In general, new features were added after inspecting and evaluating results of the models that had been built so far, hereby particularly investigating incorrect predictions by these models. As such, new features were created and added to make those models better capable of preventing these types of incorrect predictions. These features were calculated based on the input that was fed to the model pipeline. This input consists of the metadata associated with the article in which the atomic change appears, the full text of the original and new article version, the sentences containing the atomic change and the words that are actually part of the atomic change. Example values for the input are given in Fig. 3. Based on these inputs, features belonging to five different feature categories were constructed. These features are explained in more detail in the following.

Metadata

A first category of features represents metadata of the article and article version in which the atomic change is present. These include, amongst others, the day of the week on which the original and new article version were published, the hours at which both versions were published, the difference in time between both publications, the newspaper in which the article was published, the topic of the article (22 different topics were distinguished), the version number of the new version of the article and the total amount of different versions of the given article.

Textual statistics

A second category holds (statistical) information that can be easily computed based on the textual information in both article versions. This includes the textual length of the title, intro and full text of both versions, the fraction of changed text in the three text parts of both versions, the length of the sentences in the original and new version that hold the atomic change, the fraction of changed text in the sentences that hold the atomic change of interest, the Levenshtein distance between the sentences that hold the atomic change (Levenshtein et al., 1966), the sum of the Levenshtein distances calculated using only the words being part of the atomic change, the number of insert, delete and replace operations as calculated by Levenshtein's algorithm, the Jaccard similarity of the sentences (Jaccard, 1912), the number of changed words in both versions ... Finally, it is also verified whether the original and new parts of the atomic change are identical when converted to lowercase or not, and whether the changed text part of one version is a substring of the changed text part of the other version (or reversely).

Word particularities

A third group of features that can be distinguished tries to catch the semantics of the words that are part of the atomic change. This is mainly done by importing information into the model that is present in self-composed or publicly available structured data sets. One type of feature belonging to this category include the amount of stop words in the atomic change compared to the total amount of words in the atomic change. Indeed, many changed stop words could hint toward the correction of a linguistic error. Other examples of such features represent information about the extent to which colors, days, currencies, months, winds, countries and nationalities, capitals and Belgian communities are present in the original and/or new article version. A change in these kind

of entities could indicate that the atomic change is a correction of an objective error. Furthermore, a list was composed containing doubt words (such as "often", "typically", "sometimes", "maybe", "probably" ...) and strong words (such as "always", "never", "undeniably", "clearly" ...). Their presence in the original and new parts of the atomic change was summarized in numerical features. Depending on the context, the presence of doubt words and strong words in both version could be a hint toward information being sensationalized (i.e., correction of a subjective error). Finally, information on numerical (in)equality was also added. As such, the feature extraction process was able to recognize that, for example, numbers that are written fully (e.g. "fifty") are semantically equivalent to their digit counterparts ("50").

Part-of-speech tagging and Named Entity Recognition

More information on the semantics of the changed texts was added to the feature space by obtaining information from part-of-speech tagging and named entity recognition software. Part-of-speech tagging labels words in unstructured text with their word type (e.g., a proper noun or a verb), hereby possibly giving additional pieces of information such as the conjugation of a verb. Named entity recognition software tries to find named entities within unstructured text and label them with their type (e.g., a person, a product, a political party ...). Both types of software can be considered to be very useful for our tasks, especially in terms of detecting corrections of objective errors and corrections of linguistic errors. A well-known part-of-speech tagger and named entity recognizer for (amongst others) the Dutch language that is readily available in Python is spaCy.[11]

Part-of-speech tagging and named entity recognition should be applied on full sentences in order to obtain good results. Because of that, using the spaCy library for Python, part-of-speech tagging was first applied to all sentences that contained text that was part of the atomic change. This tagging information could then be used to identify the tags of all words that are part of the atomic change itself. The fraction of each word type in the original and new part of the atomic change was then calculated, and for each word type and for each article version a feature was added to the model. As an example, a change of the word "the" into the word "a" then leads to the features "determiner_original" and "determiner_new" taking on the values 1.0, and all other part-of-speech features taking on the value 0. Moreover, the features related to specific word types were split up even further:

- Features related to verbs were further split based on the tense in which the verb was conjugated and the fact whether the verb was conjugated in singular or in plural (leading to features such as "finite_verb_singular_original").
- Adjective features were split up based on the fact that they are either in standard (e.g. "weak"), comparative ("weaker") or superlative ("weakest") format.
- A different feature was created for every type of punctuation (e.g., ":", ".", "...").

Text representations

Finally, next to the manually crafted features described in the previous subsections, vectors representing the original and new text sequences constituting the atomic change were added to the model. Two vectors are created: one for the original text parts and one for the new text parts. Four different alternative language models were considered to represent the textual information:

- **TF-IDF**: this model represents each text sequence by a vector containing one dimension for each token in the entire corpus. Instead of simply creating vectors counting the number of appearances of each word in the text sequence (TF — Term Frequency), the Inverse Document Frequency (IDF) is also included. In this way, the vector representation takes into account the extent to which a word is common (or uncommon) in a corpus. The underlying corpus is built using the large amount of words that are present in atomic changes throughout the entire data set. Therefore, the textual information contained in the atomic change was represented by two very large vectors.
- **word2vec** (Mikolov et al., 2013): this language model is capable of finding good vector representations of individual words after learning from a large corpus using a neural network. Contrary to TF-IDF, the model captures semantics. If properly trained, two semantically similar words should have close vector representations. In the context of this work, a pretrained word2vec model for the Dutch language was used to represent individual words (Tulkens, Emmery, & Daelemans, 2016). This model was obtained by learning from the Sonar-500 corpus, a well-known Dutch corpus consisting of more than 500 million words of text obtained from various domains (Oostdijk, Reynaert, Hoste, & Schuurman, 2013). This word2vec model provided us with individual word representations consisting of 512 dimensions. Vector representations for the text sequences were then finally obtained by averaging out the vectors representing the individual tokens that are part of the text sequences.
- **BERTje** (de Vries et al., 2019) and **SBERT** (Reimers & Gurevych, 2019): recently, performance on many natural language processing tasks has been improved with the introduction of the transformer-based BERT (Devlin, Chang, Lee, & Toutanova, 2018) model. Since then, several BERT-inspired models have been proposed for various tasks and languages. BERTje is a monolingual Dutch BERT model trained on a large and diverse corpus consisting of 2.4 billion tokens, obtaining state-of-the-art performance on several well-known NLP tasks on Dutch text. Contrary, SBERT is another BERT-inspired approach on which several pretrained multilingual models are based, one of which provides support for the Dutch language (Reimers & Gurevych, 2020). The model is specialized for the vector representation of entire sentences and paragraphs. BERTje represents text sequences using vectors of 768 dimensions, while SBERT produces vectors of size 512. Both BERT-based pretrained models are considered and compared in this work. For both BERT-based models, features were extracted directly from the pretrained models without further fine tuning the model. This choice was made because of two reasons. First, earlier literature has

[11] https://spacy.io/.

Table 5
Overview of the number (and relative fraction) of positive and negative examples (out of 21,129 atomic changes) for each error type in the data set. The number of objective errors, subjective errors and linguistic errors is very small in comparison with the total amount of examples that are present in the data set.

Type	True	False
Correction of an objective error	546 (2.6%)	20 583 (97.4%)
Correction of a subjective error	344 (1.6%)	20 785 (98.4%)
Correction of a linguistic error	4010 (19.0%)	17 119 (81.0%)

already indicated that classification models applying the traditional approach (in which features are directly extracted from the pretrained model) have comparable performance to fine tuned models in most cases (Peters, Ruder, & Smith, 2019). Secondly, it was not our aim to achieve the best possible BERT-based classification model. Instead, the large set of algorithms and textual representations considered in the manuscript should give the reader an idea of the potential of the proposed methodology for automated news accuracy monitoring, without claiming to have already obtained best possible classification performance.

The four different language models were applied on three different types of text for both the original and new article version:

1. **Full sentences**: a concatenation of all full sentences that contain text parts that are part of the atomic change.
2. **Minimized sentences**: a concatenation of only the actual text parts that are part of the atomic change.
3. **Lemmatized sentences**: a concatenation of the actual text parts that are part of the atomic change, after which all tokens were lemmatized.

3.3.2. Dimensionality reduction and early stopping

Section 3.3.1 has made clear that a lot of features were considered during model building. However, including too many features in a model can be detrimental to its reliability. The more features that are used within a machine learning model, the more data points that are needed in the training set of the model in order to achieve good performance. If the number of dimensions is too high for the amount of available data, the trained model will be less good at generalizing its performance on the training set to unseen data (i.e., overfitting). This is often referred to as the "curse of dimensionality" or the Hughes phenomenon (Hughes, 1968). As a rule of thumb, it is often stated that for any additional dimension in the model, at least five data points in the training set are needed (Theodoridis & Koutroumbas, 2006). While the dimensionality of two word2vec, BERTje or SBERT vectors does not exceed 1536, using TF-IDF vectors as text representations leads to 32,917 features in total. The obtained data set consists of 21,129 samples. Therefore, it is clear that dimensionality reduction techniques should be applied in the case of TF-IDF to investigate the impact of the high dimensionality on the obtained results.

In total, TF-IDF embeddings account for 32,747 of the 32,917 features (99.5%) that are part of the models in which TF-IDF vectors are used. As such, for these models, latent semantic analysis (LSA) was used for reducing the number of features (Landauer & Dumais, 1997). LSA is a dimensionality reduction technique that has been specifically designed for the efficient handling of large and sparse textual matrix representations. Underlying, this technique uses singular value decomposition (SVD). Using this mathematical concept, a reduced number of (alternative) dimensions can be found that describe the data (almost) as well as the complete feature space. Therefore, first the number of features to reduce the data to was determined by setting the desired explained variance ratio (i.e., the sum of the percentages of variance that are attributed by each of the selected features) to 95%. Applying latent semantic analysis showed that the number of features needed to reach this threshold was 1850. This is a very strong dimensionality reduction compared to the original number of features of 32,917. The impact of this dimensionality reduction on the performance of our models was then investigated.

Another relevant source of overfitting problems is the fact that too many iterations of a given machine learning algorithm are run when training the model. Similarly to having too many dimensions representing the data, this may again lead to models with low generalization capability to unseen data. Therefore, early stopping was applied when training all models in order to determine the optimal number of training iterations and, as such, prevent overfitting.

3.3.3. Preventing class imbalance

An important challenge in building the supervised models lies in the fact that the gathered data set is heavily imbalanced. Table 5 depicts the number of positive and negative examples for each error type.

While the different error types are more important than there non-erratic counterparts, it is clear that they only form a small minority of the gathered records. Class imbalance is a difficult challenge to deal with, as most of the machine learning algorithms assume that the number of records for each class is more or less the same (Liu, Wu, & Zhou, 2008). Neglecting the imbalance of our data set would probably lead to a model that favors the prediction of negative records, hereby maximizing prediction accuracy, but also increasing the amount of false negatives while neglecting the fact that the positive minority class is the one that we are really interested in Hu, Gan, Zhu, Liu, and Shi (2022). Several approaches exist to deal with this problem. In this work, two different approaches were combined.

First, records belonging to the minority class were randomly oversampled. Stated differently, existing minority records were picked randomly with replacement from the data set and were added twice to the data set. As such, the amount of minority examples

was artificially increased in order to obtain a more balanced data set. The amount of oversampling was dependent on the specific model at hand. For the detection of corrections of objective errors 32% of the samples in the final data set were minority samples, while this was 40% for the subjective errors and the linguistic errors. These percentages are the result of hyperparameter tuning being performed, as explained further. Moreover, it should also be noted that oversampling was only performed on the training set, not on the validation and test sets. In this way, the evaluation of the model performance was not affected.

Secondly, the weights of the classes in the cost function of the model were altered such that they were inversely proportional to the relative number of cases in the data set. The weight assigned to class C, denoted W_C, is given by Eq. (1).

$$W_C = \frac{N_{tot}}{N_C \times M} \tag{1}$$

Here, N_C and N_{tot} denote respectively the number of cases belonging to class C and the number of cases in the data set in total. M represents the number of classes. As an example, after having applied random oversampling in the case of the model for linguistic errors, 40% of the samples represented a linguistic error. In the cost function that was subsequently minimized in order to obtain the best model, individual samples representing linguistic errors were taken into account with weight 1.25 instead of 1. Contrary, the samples representing negative examples of linguistic errors had corresponding weight 0.83. As such, a more balanced scenario was obtained.

3.3.4. Algorithm selection

The main focus of this research is on the presentation of the idea of monitoring news accuracy automatically by looking at changes made to online news articles. In order to illustrate the potential of this procedure, three supervised models are built that are capable of detecting the different error types in online news updates. However, the main purpose of this paper is not to immediately find the best possible model for the given task at hand. That is, only a selected set of well-known predictive algorithms is considered. Probably, better results could certainly be obtained using more sophisticated models such as higher order models, ensembles or neural networks. However, in this paper the performance of three different algorithms is compared: logistic regression (Cramer, 2002), decision trees (Quinlan, 1986) and support vector machines (Cortes & Vapnik, 1995). For each of these algorithms, efficient implementations are available in scikit-learn.

3.3.5. Hyperparameter tuning and model evaluation

In order to obtain the best possible model, given the set of features and algorithms available, specific attention was given to the way in which hyperparameters were tuned for each model. Moreover, we focused on the way in which model performance was evaluated. Relevant aspects in this regard are detailed out in the following.

Training and testing the model

In order to accurately estimate the performance of our models, the original data set was split into a training set (80%) and a test set (20%). Furthermore, stratified 5-fold cross validation was performed within the 80% of training data, meaning that the training data set was split up into five different folds of equal size. Moreover, the relative amount of positive samples was held equal in training and test set. Evaluating model performance is then done by iterating through five different rounds, each time taking the combination of four out of five folds as the actual training data in this round and the other fold as the validation set on which the model is evaluated. During each round, the validation fold changes. Finally, the model is then trained on all available training data and the resulting model is evaluated on the test set. As already mentioned, oversampling techniques were only applied on the actually used training data in order to be able to evaluate model performance in a realistic, practical data setting.

Hyperparameter optimization

Each of the algorithms under consideration has a number of associated hyperparameters of which the value can heavily impact the performance of the eventual model. Therefore, it is important to tune these hyperparameters in order to identify which combinations of parameters perform best. In this work, for each of the three algorithms under consideration, a grid search was performed in combination with stratified 5-fold cross validation. An overview of the grid of hyperparameters that were considered from the hyperparameters present in sci-kit learn is given in Table 6.

Evaluation measures

Finally, a good evaluation metric had to be chosen in order to comply with the goals of our work. Typical measures that are often considered in the context of imbalanced data are precision

$$P = \frac{\mathsf{tp}}{\mathsf{tp} + \mathsf{fp}} \tag{2}$$

and recall

$$R = \frac{\mathsf{tp}}{\mathsf{tp} + \mathsf{fn}} \tag{3}$$

where tp represents the number of true positives, fp the number of false positives, tn the number of true negatives and fn the number of false negatives. In addition, a generalized F_β score was used to combine P and R in one single measure of quality:

$$F_\beta = (1 + \beta^2) \cdot \frac{P \cdot R}{\beta^2 \cdot P + R} \tag{4}$$

Table 6
Overview of the hyperparameter combinations in sci-kit learn that were considered during model evaluation for logistic regression, decision trees and support vector machines. The extensive grid search that was performed over the combination of all these hyperparameters guarantees the selection of the best model investigated in this work.

Algorithm	Parameter grid
Logistic regression	**C (inverse regularization strength):** 0.01, 0.1, 1 **penalty:** L1, L2, elasticnet **solver (optimization algorithm):** newton-cg, lbfgs, liblinear, sag, saga
Decision trees	**criterion (function to measure quality of a split):** entropy, gini, log_loss **max_depth:** 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30
Support vector machines	**C (inverse regularization strength):** 0.00001, 0.0001, 0.001, 0.01, 0.1, 1 **penalty:** L1, L2 **loss (the loss function used):** hinge, squared_hinge

Here, β is a parameter indicating the importance of recall in comparison to precision for your specific use case. As such, a higher value for β implies a stronger focus on the optimization of recall, while lower values of β tend to primarily optimize the precision of the model. In order to set a correct value for β, it should be noted from Eqs. (2) and (3) that precision is mainly determined by the number of false positives and recall by the number of false negatives. False negatives are model mistakes in which objective, subjective or linguistic error corrections are wrongly predicted to be another type of atomic change. As an example, a classification model predicting the replacement of the word *hause* by the word *house* to be a content-neutral reformulation (while it is in fact the correction of a linguistic error) constitutes a false negative. False positives are model mistakes in which atomic changes that do not correct any errors are predicted to be the correction of an objective, subjective or linguistic error. An example of a false positive is the model prediction that the replacement of the word *house* by the word *home* is a correction of a linguistic error, while it is in fact a content-neutral reformulation.

While both types of mistakes should preferably be prevented as much as possible, it is clear that false negatives are more severe in the current context. If an atomic change is predicted to be an error correction, a more detailed human analysis should be performed. As such, the samples requiring human effort will be either true positives or false positives. Although an increase in the number of false positives raises the number of atomic changes that need to be manually checked, their actual impact on the correctness of the monitoring process is negligible. However, on the other hand, false negatives will remain undetected by the monitoring process. The higher the number of false negatives, the lower the quality of the monitoring process thus is. As such, it is clear that preventing false negatives from occurring is more important than preventing the occurrence of false positives. Therefore, β was set equal to two, leading to the final evaluation metric F_2.

4. Results

In this section, an overview is given of the results that were obtained when building models for distinguishing corrections of objective, subjective and linguistic errors from other types of atomic changes in online news updates. Three different learning algorithms were considered (logistic regression, decision trees and support vector machines) in combination with four language models (TF-IDF, word2vec, BERTje and SBERT). Moreover, the case in which no text embeddings are added to the feature set was also investigated. The best set of hyperparameters is chosen for each possible combination of feature set, learning algorithm and error type using hyperparameter optimization.

Tables 7–9 show F_2-scores obtained for the detection of objective, subjective and linguistic errors respectively. These results are obtained after performing hyperparameter optimization using a training set, validation set and test set in combination with five-fold cross validation as explained in Section 3.3.1. For the models in which text embeddings are included, only the highest F_2-scores, obtained by applying language models on either the full, minimized or lemmatized sentences, are given. Moreover, the tables also provide 95% confidence intervals for the obtained F_2-scores. These were obtained by following the approach proposed by Goutte and Gaussier (2005), who treat the F_β-score as a probabilistic random variable. As such, means (μ) and standard deviations (σ) can be obtained. 95% confidence intervals for the F_2-scores are then calculated as given by Eq. (5).

$$CI_{95} = [\mu - 1.96 \times \sigma, \mu + 1.96 \times \sigma] \tag{5}$$

The results indicate that logistic regression and support vector machines perform very similar for the three tasks at hand, while the performance of the decision trees is somewhat lower for most tasks. However, the obtained confidence intervals for the detection of objective errors and subjective errors are quite wide. As such, no generalized conclusions concerning these differences can be drawn. The set of optimal hyperparameters for the best performing models for each task are given in Table 10.

The impact of dimensionality reduction on the obtained F_2-scores was determined for each model in which TF-IDF vectors were used. This was done by reducing the number of dimensions to 1850 (corresponding to 95% of explained variance) and performing hyperparameter optimization again. Applying the optimized models on the reduced data lead to exactly the same F_2-scores as presented in Tables 7–9. Moreover, further increasing the number of dimensions above 1850 did not increase F_2-scores. Therefore, it can be concluded that the high dimensionality of our data does not affect the prediction performance of our models. Moreover, analysis of the obtained F_2-scores in function of the number of iterations for each algorithm at hand showed that after five to fifteen

Table 7
F_2-scores obtained on the test set for the task of detecting objective error corrections. Scores (and corresponding 95% confidence intervals) are given for three different algorithms (logistic regression, decision tree and support vector machine) applied on five different types of linguistic features (no textual representation, TF-IDF, word2vec, BERTje and SBERT). The highest score obtained for each individual language model is highlighted in bold.

	Logistic regression	Decision tree	Support vector machine
No textual representation	**0.35 [0.29, 0.40]**	0.28 [0.21, 0.35]	0.34 [0.28, 0.38]
TF-IDF	0.41 [0.35, 0.47]	0.28 [0.20, 0.35]	**0.45 [0.37, 0.53]**
word2vec	0.37 [0.31, 0.43]	0.34 [0.27, 0.41]	**0.38 [0.32, 0.44]**
BERTje	**0.39 [0.32, 0.46]**	0.33 [0.26, 0.40]	0.38 [0.30, 0.46]
SBERT	0.41 [0.35, 0.47]	0.31 [0.23, 0.38]	**0.43 [0.35, 0.49]**

Table 8
F_2-scores obtained on the test set for the task of detecting subjective error corrections. Scores (and corresponding 95% confidence intervals) are given for three different algorithms (logistic regression, decision tree and support vector machine) applied on five different types of linguistic features (no textual representation, TF-IDF, word2vec, BERTje and SBERT). The highest score obtained for each individual language model is highlighted in bold.

	Logistic regression	Decision tree	Support vector machine
No textual representation	0.15 [0.11, 0.19]	**0.23 [0.15, 0.31]**	0.16 [0.12, 0.20]
TF-IDF	0.23 [0.17, 0.29]	0.22 [0.15, 0.30]	**0.25 [0.17, 0.35]**
word2vec	0.16 [0.12, 0.20]	**0.21 [0.15, 0.28]**	0.19 [0.14, 0.24]
BERTje	**0.22 [0.16, 0.29]**	0.16 [0.09, 0.23]	0.20 [0.12, 0.28]
SBERT	**0.25 [0.18, 0.31]**	0.24 [0.16, 0.32]	0.25 [0.18, 0.32]

Table 9
F_2-scores obtained on the test set for the task of detecting linguistic error corrections. Scores (and corresponding 95% confidence intervals) are given for three different algorithms (logistic regression, decision tree and support vector machine) applied on five different types of linguistic features (no textual representation, TF-IDF, word2vec, BERTje and SBERT). The highest score obtained for each individual language model is highlighted in bold.

	Logistic regression	Decision tree	Support vector machine
No textual representation	**0.78 [0.76, 0.80]**	0.75 [0.72, 0.77]	0.78 [0.76, 0.80]
TF-IDF	**0.79 [0.77, 0.81]**	0.75 [0.72, 0.77]	0.79 [0.77, 0.81]
word2vec	0.78 [0.76, 0.80]	0.76 [0.73, 0.78]	**0.77 [0.75, 0.79]**
BERTje	**0.80 [0.79, 0.83]**	0.71 [0.68, 0.74]	0.79 [0.76, 0.80]
SBERT	**0.80 [0.78, 0.82]**	0.77 [0.75, 0.80]	0.80 [0.78, 0.82]

Table 10
Set of optimal hyperparameters for the best performing models in Tables 7, 8 and 9 for each task at hand.

	Best algorithm	Best language model	Optimal hyperparameters
Objective errors	Support vector machine	TF-IDF	**C:** 0.1 **penalty:** L2 **loss:** squared-hinge
Subjective errors	Support vector machine	TF-IDF	**C:** 0.01 **penalty:** L2 **loss:** squared-hinge
Linguistic errors	Logistic regression	BERTje	**C:** 0.1 **penalty:** L1 **solver:** liblinear

iterations, the same scores were obtained as given in Tables 7 and 8. Further increasing the number of iterations did not improve the obtained scores.

In order to have better insight in the performance of the best-performing models for each task at hand (based on the mean of the obtained F_2-score distributions), on the left side of Fig. 4 the confusion matrices (and normalized scores) are given after application of the models on the 20% test data. Moreover, the numbers in the confusion matrices can be put in perspective of the complexity of the tasks by looking at the confusion matrices that represent the accuracy of the external, human annotators. These confusion matrices are given on the right side of Fig. 4 for objective errors, subjective errors and linguistic errors respectively. The presented numbers for human annotation were obtained by using the number of annotations by external annotators that had to be corrected during review (as was described in Section 3.2.2). Finally, Table 11 compares the F_2-scores obtained by the best performing model for each task with the scores obtained by human annotation. The latter scores can hereby be seen as an indication of the complexity of the tasks at hand.

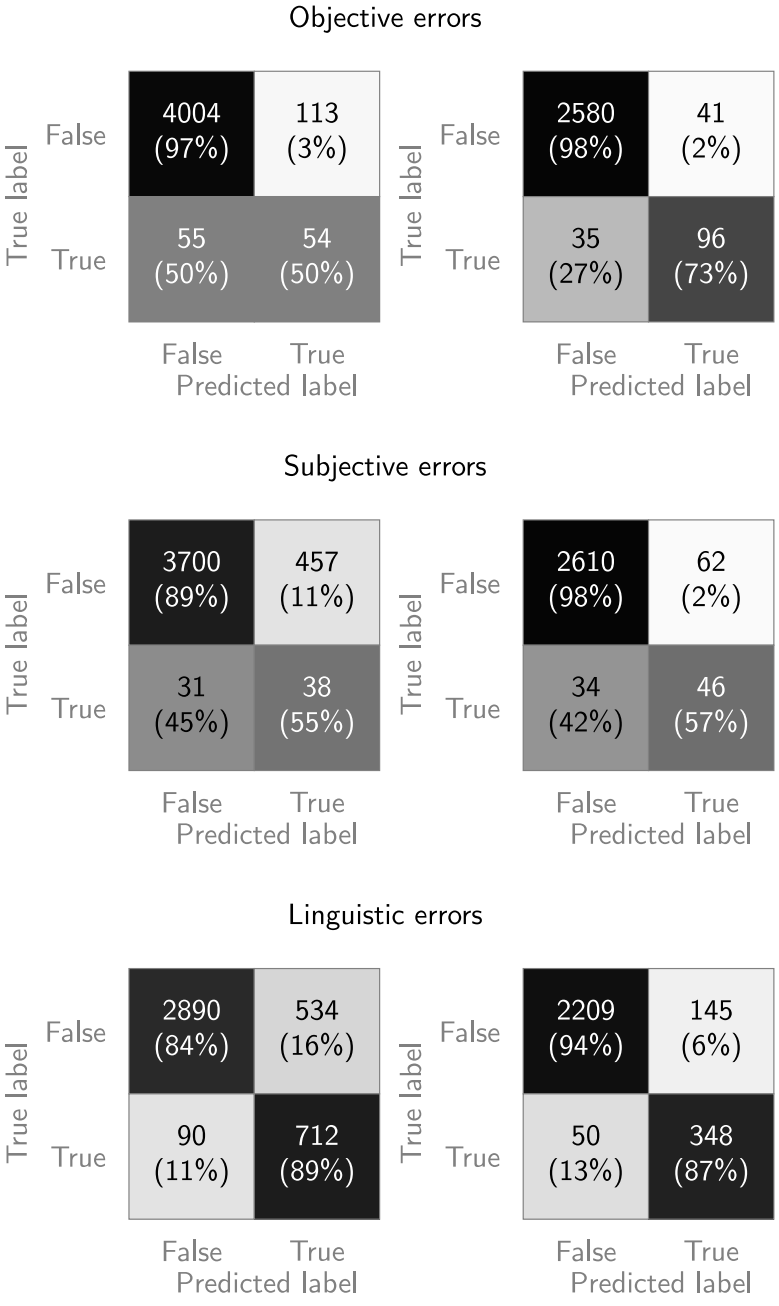

Fig. 4. *Left*: confusion matrices obtained on the objective errors (first line), subjective errors (second line) and linguistic errors (third line) detection problems by applying the **best automated classification model** that is available for that task (support vector machine in combination with TF-IDF text representation for objective and subjective error detection, logistic regression in combination with BERTje text representation for linguistic error detection). *Right*: confusion matrices obtained on the objective errors (first line), subjective errors (second line) and linguistic errors (third line) detection problems by reviewing **annotations by external, human annotators**. Normalized confusion matrix numbers are given between brackets.

5. Analysis & discussion

The results presented in Section 4 indicate that the best results for the given set of features are obtained by either the logistic regression model (in the case of objective and subjective error detection) or the support vector machine (for linguistic error detection). Decision trees perform a little less compared to the other algorithms that were tested. However, the 95% confidence intervals of the three learning algorithms overlap in most cases. It can thus be concluded that the suitability of logistic regression of

Table 11
Comparison of F_2-scores (and recall and precision) obtained by the best automated classification model available (support vector machine for objective and subjective error detection, logistic regression for linguistic error detection) with F_2-scores (and recall and precision) obtained by human annotation for the tasks of the detection of corrections of objective, subjective and linguistic errors.

	Objective errors	Subjective errors	Linguistic errors
Automated classification	0.45 (R = 0.50, P = 0.32)	0.25 (R = 0.55, P = 0.08)	0.80 (R = 0.89, P = 0.57)
Human annotation	0.73 (R = 0.73, P = 0.70)	0.54 (R = 0.57, P = 0.43)	0.83 (R = 0.87, P = 0.71)

support vector machines for the supervised tasks at hand is more or less comparable, obtaining slightly better results in comparison with decision trees.

More profound differences can be observed when looking at the performance obtained by models built on different language models. First of all, the results in Tables 7–9 indicate that in comparison with the models in which only manually extracted features are present ("No textual representation"), the models to which word2vec embeddings were added did not perform any better. The highest F_2-scores obtained for models without textual representations and word2vec models are very similar for both objective, subjective and linguistic error detection. A possible explanation for this is that word2vec is primarily used for the representation of individual words. While sentences can be represented by vectors by averaging out the vectors of the individual words constituting the sentence, this might not lead to the best possible vector representation. Secondly, the TF-IDF vectors improve the F_2-scores obtained in comparison with the scenario in which no embeddings are added to the model. This is especially true in the case of the detection of objective errors. An increase of 0.10, 0.02 and 0.02 in F_2-score was registered for objective, subjective and linguistic errors respectively. It should be noted that the obtained scores on the validation and test set for the TF-IDF models deviate significantly from the ones obtained on the training data (F_2-scores around 0.85 for the three models at hand), especially for the models for the detection of objective and subjective errors. The reason for this can be found in the inclusion of terms with very low document frequency in the model. Apparently, the inclusion of features related to terms with low document frequency significantly improves the scores obtained on the training set, while improving scores obtained on the validation and test scores only to a limited extent. Nonetheless, because of the fact that their inclusion still improves the obtained validation and test scores, it was decided to keep all TF-IDF features in the final models.

Finally, the results indicate that performance of BERTje and SBERT is more or less similar, although the multilingual SBERT model (which is specialized for the representation of individual sentences) performs slightly better for two out of three tasks. In comparison with TF-IDF based models, the impact of including BERT-based embeddings in the model is fairly limited. For the case of the detection of objective and subjective errors, the best TF-IDF model performs slightly better on the test set than its best-performing BERTje and SBERT counterparts. In the case of linguistic error detection, both best-performing BERT-based models obtained slightly higher F_2-scores on the test set than the best-performing TF-IDF model. However, in all three scenarios, the inclusion of BERTje or SBERT embeddings instead of TF-IDF vectors did not lead to models with significantly higher predictive performance. A possible explanation for this is given by the fact that the pretrained BERT-based models that were used have been trained on standard NLP tasks such as named entity recognition or part-of-speech tagging. The tasks considered within this work are however very specific and also have very specific textual information associated with them (i.e., words representing changes between two subsequent article versions). Therefore, the limited impact of BERTje and SBERT embeddings may be explained by the difference between the linguistic context in which the models were pretrained and the linguistic context in which the embeddings are actually used in this work.

It can be seen that obtained F_2-scores vary significantly for the three tasks at hand. While a very high score is obtained for the detection of the correction of linguistic errors ($F_2 = 0.80$) and a quite good score is obtained for the detection of the correction of objective errors ($F_2 = 0.45$), the obtained F_2-score for the detection of subjective error corrections is low ($F_2 = 0.25$). The large difference in obtained F_2-scores between the task of classifying objective errors, subjective errors and linguistic errors is, too a large extent, a direct consequence of the complexity of the tasks at hand. Although the measured F_2-scores obtained by the external annotators are higher than the ones obtained using machine learning, the same trends can be observed for human annotation as well: human annotators perform best in detecting corrections of linguistic errors, somewhat less for objective errors and clearly less in the case of subjective errors. For the three models at hand, the recall values obtained by the best-performing models are better than the corresponding precision values (see Table 11). An obvious reason for this is the fact that the data are heavily imbalanced for both cases. This implies that precision can decrease very rapidly with an increasing number of false positives, even if that number of false positives is still very low in comparison to the number of true negatives. It is clear that the fact of the data being heavily imbalanced adds additional complexity to the tasks at hand.

The best performance is obtained in the detection of linguistic error corrections, as the machine learning model obtains an F_2-score that is almost as good as the performance obtained by human annotators. As expected, characteristic features of spelling errors are that the length of the texts being part of the atomic changes are typically small and Levenshtein distances between original text version and new text version are small too. Moreover, other types of linguistic errors are detected by information of changes of plural form to singular form (or reversely), changes in the articles of words, addition or deletion of punctuation ... The main type of atomic changes with which the model still confuses corrections of linguistic errors are content-neutral reformulations, as these also have the associated characteristic to consist of only a number of different words in each text version. However, obtaining a maximal F_2-score of 0.80, this first model for the detection of the correction of linguistic errors can already be very useful in practice to monitor linguistic quality of online news in Flanders.

With regards to the detection of objective error corrections, the F_2-score obtained by the best-performing model is reasonable ($F_2 = 0.45$). The model performs very well at detecting atomic changes in which one given type of entity is replaced by another entity of the same type (e.g. changing names of persons or organizations). These atomic changes indeed often happen to be a correction of an objective error in practice. However, this is not always the case, as names could also be simply misspelled, organizations may have multiple names (e.g. abbreviations) ... Moreover, contrary to the case of linguistic errors, objective errors are also sometimes corrected in a more elaborate and subtle way. In these cases, full sentences are rewritten instead of changing only one word. Analysis of the current model performance indicates that the model performs less well in these contexts. Nonetheless, we believe this first attempt of creating a classifier distinguishing objective error corrections from other atomic changes could be definitively useful in combination with other automated fact checking tools.

Finally, with regards to the corrections of subjective errors, it is clear that even human annotators found it difficult to distinguish them from other atomic changes. It is therefore no surprise that the performance of our first attempt to automatically detect those corrections is also less good compared to the other types of error corrections. Subjective errors are (as the name suggests) often matter for discussion. While in the case of objective errors it is often very clear that an error was present in the original article, subjective errors are more subtle. For example, while a headline may be too sensational to one person, another person may find that nothing is wrong with this headline. Although all external annotations were reviewed in order to be sure that all samples were annotated in the same, rigorous way, it is clear that the current set of features is not satisfactory in order to precisely retrieve most of the subjective error corrections. Features that have been found particularly useful in increasing the recall of the subjective error classifier are the part-of-speech tagging features related to adjectives (basic, comparative or superlative) and the features containing information with regards to the presence of doubt words and strong words in the original and new texts of the atomic change. Indeed, many subjective errors are made by making statements in the original text version in which superlatives are used (e.g. *He is the best.*), while they have to be rephrased less expressively later (e.g. *He was better than his opponent today.*) in order to better reflect reality. Similarly, atomic changes in which uncertain information is initially presented in a very sure way (e.g. *It is certain that at least one person died.*) but that later have to be weakened (e.g. *Possibly one person did not survive*) can be detected using the information in these features as well. In conclusion, while the presented models for the detection of linguistic error corrections and objective error corrections may already be very useful in combination with other existing tools and human supervision to monitor news accuracy (partially) automatically, it is clear that additional research is required for the case of subjective error corrections.

It is important to notice that the presented strategy will not work in the combat against disinformation (i.e., the intended spreading of misinformation), as people that are intentionally spreading fake news will not correct errors present in their articles. However, we believe it can be very useful in order to monitor online news accuracy. In general, although human supervision is certainly still required in order to monitor news accuracy using the presented models, it is clear that the amount of work when applying our approach is significantly lower compared to the situation in which atomic change annotations would be performed fully manually. Taking our test data set as an example, manual monitoring of the accuracy would require the inspection of 4226 atomic changes. Using the three proposed models, this would be lowered to at most 1908 samples (taking together all samples that were predicted to be positive in any of the three models). This is a reduction in workload of 55%. Probably this number would be even less in practice, as samples could be predicted to be positive by different models at a time. Of course, this reduction in workload comes with a cost, namely an associated decrease in annotation accuracy. In total 176 false negatives (at most) would have been missed on a total of 980 positive cases (corresponding to (at most) 18% of the errors present in the data set). Although still an important share of false negatives are present in the final results, we believe these numbers, that were obtained by only using standard predictive algorithms, strongly advocate the advantages of the presented approach and for further research regarding correction-based news accuracy monitoring.

6. Conclusion

In this work, a new way of looking at online news accuracy and automated accuracy monitoring is proposed. Online news accuracy is monitored by looking at the extent to which corrections are made to news articles after publication. Contrary to many existing approaches, the proposed procedure does not require an external fundamental truth in order to verify the correctness of a textual statement. Three subsequent steps are required by the procedure in order to be successful: (1) the automated monitoring of the publication of new online articles and the changes that are made to these, (2) the automated clustering of the changed text parts into several atomic changes and (3) the categorization of each atomic change in terms of its type.

The work presented in this paper focuses on the automation of steps one and three of the pipeline. Using scraping software, six Flemish news websites were monitored over a period of two years and all articles and updates that were published were collected. In order to automate the third step of the pipeline, three supervised models were created for the detection of objective, subjective and linguistic error corrections respectively. Annotated data were obtained by performing a large-scale manual annotation process using annotators both internal and external to the project. As such, a data set of 21,129 labeled atomic changes was obtained. During model building, several well-known algorithms were investigated and optimized. Moreover, different language models were considered for the representation of textual information. No significant differences in performance was observed for the investigated learning models (logistic regression, decision trees and support vector machines). However, larger differences in obtained F_2-scores were noticed between the models using different textual representations. Maximal F_2-scores of 0.25, 0.45 and 0.80 were obtained for the detection of subjective, objective and linguistic error corrections respectively. These were obtained using TF-IDF representations in the case of subjective and objective error detection and BERTje/SBERT-based embeddings in the case of linguistic error detection. Comparison with human annotation results illustrates that the detection models for objective and linguistic error corrections could definitively prove their usefulness in practice in combination with other existing tools. The model for subjective errors performs less well. This is in correspondence with our initial expectations and task difficulties.

6.1. Theoretical and practical implications

The main theoretical implication of this work is that it provides researchers with a new theoretical way of looking at the problem of detecting misinformation in online news. Earlier approaches regarding the detection of erroneous information in online news and fake news typically required the existence of an external ground truth. However, an existing problem in this regard is the fact that such external ground truths are difficult to construct and even do not exist in many practical contexts. The methodology proposed in this work provides a solution to this problem, as it does not require any external information. The theoretical approach taken in this work can also be easily extended to and implemented for other languages.

Two practical implications can be distinguished. First, a large-scale annotated data set containing Flemish online news articles, their associated updates and the types of the changes performed in those updates gathered and made publicly available for further use by other researchers. To our knowledge, this is the first data set containing such a large amount of valuable information on the usage of the update functionality in online Flemish news. Independently of the context of this research and its results, this data set will offer plenty of opportunities for further work, both for researchers in the field of computer science and in the field of journalism studies. Secondly, researchers and news practitioners are offered practically usable models for the automated monitoring of online news accuracy. Although the model for the detection of subjective error corrections should be used with caution due to the difficulty of the task, we believe that the supervised models that were obtained for objective errors and certainly for linguistic errors can be useful for practical news accuracy monitoring experiments. As such, journalists and researchers have a new tool in their toolkit to monitor and guarantee the quality of online news in Flanders, doing less work than before while being able to preserve reliability of the monitoring process.

6.2. Future work

Although the first results described in this work are promising, future work is definitely needed in order to fully automate the news accuracy monitoring approach presented and in order to improve results. First, the second step of the pipeline, i.e. the identification of atomic changes within news updates, should still be automated. Despite the work presented in this research, the news accuracy monitoring process cannot be fully automated at this moment, because manual identification of the atomic changes within a news update are still needed. This is an important issue that should be solved in order to make the presented system applicable on a large scale. Secondly, although the performance of the presented objective and linguistic error models is already good, until now only three algorithms were evaluated. More advanced machine learning techniques, such as ensembles or neural networks, in combination with additional feature engineering (e.g., fine tuning BERTje and SBERT models) could probably still increase performance of the models. Finally, the performed data collection process and associated models that were built are only useful in the context of Flemish online news. However, it is clear that news accuracy is a very important topic in other countries as well. In conclusion, a high, worldwide demand for automated accuracy monitoring tools exists. Therefore, further research into the development of these kinds of correction-based models is certainly required.

CRediT authorship contribution statement

Yoram Timmerman: Conceptualization, Methodology, Software, Formal analysis, Investigation, Resources, Data curation, Writing – original draft, Visualization, Funding acquisition. **Antoon Bronselaer:** Conceptualization, Methodology, Validation, Resources, Writing – review & editing, Supervision, Project administration.

Data availability

Data are shared via a GitHub repository that is referenced to in the manuscript.

References

Ahmadi, N., Lee, J., Papotti, P., & Saeed, M. (2019). Explainable fact checking with probabilistic answer set programming. In *Conference on truth and trust online*. http://dx.doi.org/10.36370/tto.2019.15.

Anderson, C. W. (2011). Between creative and quantified audiences: Web metrics and changing patterns of newswork in local US newsrooms. *Journalism*, *12*(5), 550–566. http://dx.doi.org/10.1177/1464884911402451.

Appelman, A., & Hettinga, K. (2021). Correcting online content: The influence of news outlet reputation. *Journalism Practice*, *15*(10), 1562–1579. http://dx.doi.org/10.1080/17512786.2020.1784776.

Arnold, P. (2020). *The challenges of online fact checking: (Tech. rep.)*, Technical report, Full Fact.

Berendt, B., Burger, P., Hautekiet, R., Jagers, J., Pleijter, A., & Van Aelst, P. (2021). FactRank: Developing automated claim detection for dutch-language fact-checkers. *Online Social Networks and Media*, *22*, Article 100113. http://dx.doi.org/10.1016/j.osnem.2020.100113.

Berry, F. C., Jr. (1967). A study of accuracy in local news stories of three dailies. *Journalism Quarterly*, *44*(3), 482–490. http://dx.doi.org/10.1177/107769906704400309.

Blankenburg, W. B. (1970). News accuracy: Some findings on the meaning of errors. *Journal of Communication*, *20*(4), 375–386. http://dx.doi.org/10.1111/j.1460-2466.1970.tb00896.x.

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., & Taylor, J. (2008). Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on management of data* (pp. 1247–1250). http://dx.doi.org/10.1145/1376616.1376746.

Brautovic, M., Maštrapa, S. B., & John, R. (2020). Accuracy in online media: Insufficient journalistic routines in fact-checking and corrections. *Media Studies*, *11*(21), 66–86. http://dx.doi.org/10.20901/ms.11.21.4.

Brautović, M. (2021). Corrections practice in the Croatian online media: Between legislation and tradition. *Društvena Istraživanja-Časopis Za Opća Društvena Pitanja*, *30*(4), 785–806. http://dx.doi.org/10.5559/di.30.4.07.

Brown, C. H. (1965). Majority of readers give papers an a for accuracy. *Editor & Publisher*, *13*, 482–490.

Burggraaff, C., & Trilling, D. (2020). Through a different gate: An automated content analysis of how online news and print news differ. *Journalism*, *21*(1), 112–129. http://dx.doi.org/10.1177/1464884917716699.

Charnley, M. V. (1936). Preliminary notes on a study of newspaper accuracy. *Journalism Quarterly*, *13*(4), 394–401. http://dx.doi.org/10.1177/107769903601300403.

Chen, W., Wang, H., Chen, J., Zhang, Y., Wang, H., Li, S., Zhou, X., & Wang, W. Y. (2020). TabFact : A large-scale dataset for table-based fact verification. In *International conference on learning representations (ICLR)*. Addis Ababa, Ethiopia: http://dx.doi.org/10.48550/arXiv.1909.02164.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297. http://dx.doi.org/10.1023/A:1022627411411.

Cramer, J. S. (2002). The origins of logistic regression. *Tinbergen Institute Working Paper*, http://dx.doi.org/10.2139/ssrn.360300.

de Vries, W., van Cranenburgh, A., Bisazza, A., Caselli, T., Noord, van, G., & Nissim, M. (2019). BERTje: A dutch BERT model. arXiv http://dx.doi.org/10.48550/arXiv.1912.09582.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. N. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv http://dx.doi.org/10.48550/arXiv.1810.04805.

Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin*, *76*(5), 378. http://dx.doi.org/10.1037/h0031619.

Forde, S. L., Gutsche, R. E., Jr., & Pinto, J. (2022). Exploring "ideological correction" in digital news updates of portland protests & police violence. *Journalism*, Article 14648849221100073. http://dx.doi.org/10.1177/14648849221100073.

Fox, C., Knowlton, S., Maguire, Á., & Trench, B. (2009). Accuracy in Irish newspapers. *Press Council of Ireland*, *200609*, 20.

Gad-Elrab, M. H., Stepanova, D., Urbani, J., & Weikum, G. (2019). Tracy: Tracing facts over knowledge graphs and text. In *The world wide web conference* (pp. 3516–3520). New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/3308558.3314126.

Goutte, C., & Gaussier, E. (2005). A probabilistic interpretation of precision, recall and F-score, with implication for evaluation. In D. E. Losada, & J. M. Fernández-Luna (Eds.), *Advances in information retrieval* (pp. 345–359). Springer Berlin Heidelberg, http://dx.doi.org/10.1007/978-3-540-31865-1_25.

Hassan, N., Zhang, G., Arslan, F., Caraballo, J., Jimenez, D., Gawsane, S., Hasan, S., Joseph, M., Kulkarni, A., & Nayak, A. K. (2017). ClaimBuster: The first-ever end-to-end fact-checking system. *Proceedings of the VLDB Endowment*, *10*(12), 1945–1948. http://dx.doi.org/10.14778/3137765.3137815.

Hettinga, K., & Smith, E. (2021). How a copy desk "edit" influenced corrections at the new york times. *Newspaper Research Journal*, *42*(2), 182–197. http://dx.doi.org/10.1177/07395329211013506.

Hu, R., Gan, J., Zhu, X., Liu, T., & Shi, X. (2022). Multi-task multi-modality SVM for early COVID-19 diagnosis using chest CT data. *Information Processing & Management*, *59*(1), Article 102782. http://dx.doi.org/10.1016/j.ipm.2021.102782.

Hughes, G. (1968). On the mean accuracy of statistical pattern recognizers. *IEEE Transactions on Information Theory*, *14*(1), 55–63. http://dx.doi.org/10.1109/TIT.1968.1054102.

Jaccard, P. (1912). The distribution of the flora in the alpine zone. *New Phytologist*, *11*(2), 37–50.

Karagiannis, G., Saeed, M., Papotti, P., & Trummer, I. (2020). Scrutinizer: A mixed-initiative approach to large-scale, data-driven claim verification. *Proceedings of the VLDB Endowment*, [ISSN: 2150-8097] *13*(12), 2508–2521. http://dx.doi.org/10.14778/3407790.3407841.

Karlsson, M. (2012). Charting the liquidity of online news: Moving towards a method for content analysis of online news. *International Communication Gazette*, *74*(4), 385–402. http://dx.doi.org/10.1177/1748048512439823.

Karlsson, M., Clerwall, C., & Nord, L. (2017). Do not stand corrected: Transparency and users' attitudes to inaccurate news and corrections in online journalism. *Journalism & Mass Communication Quarterly*, *94*(1), 148–167. http://dx.doi.org/10.1177/1077699016654680.

Kautsky, R., & Widholm, A. (2008). Online methodology: Analysing news flows of online journalism. *Westminster Papers in Communication & Culture*, *5*(2), http://dx.doi.org/10.16997/wpcc.69.

Kocher, D. J., & Shaw, E. F. (1981). Newspaper inaccuracies and reader perceptions of bias. *Journalism Quarterly*, *58*(3), 471–516. http://dx.doi.org/10.1177/107769908105800322.

Kovach, B., & Rosenstiel, T. (2011). *Blur: how to know what's true in the age of information overload*. Bloomsbury Publishing USA.

Kutz, D. O., & Herring, S. C. (2005). Micro-longitudinal analysis of web news updates. In *Proceedings of the 38th annual Hawaii international conference on system sciences* (p. 102a). IEEE, http://dx.doi.org/10.1109/HICSS.2005.409.

Landauer, T. K., & Dumais, S. T. (1997). A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, *104*(2), 211. http://dx.doi.org/10.1037/0033-295X.104.2.211.

Lazer, D. M., Baum, M. A., Benkler, Y., Berinsky, A. J., Greenhill, K. M., Menczer, F., Metzger, M. J., Nyhan, B., Pennycook, G., & Rothschild, D. (2018). The science of fake news. *Science*, *359*(6380), 1094–1096. http://dx.doi.org/10.1126/science.aao2998.

Lee, A. M., Lewis, S. C., & Powers, M. (2014). Audience clicks and news placement: A study of time-lagged influence in online journalism. *Communication Research*, *41*(4), 505–530. http://dx.doi.org/10.1177/0093650212467031.

Lee, N., Li, B. Z., Wang, S., Yih, W.-T., Ma, H., & Khabsa, M. (2020). Language models as fact checkers? In *Proceedings of the third workshop on fact extraction and verification (FEVER)* (p. 36). http://dx.doi.org/10.18653/v1/2020.fever-1.5.

Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady, Vol. 10* (pp. 707–710). Soviet Union.

Li, Y., Gao, J., Meng, C., Li, Q., Su, L., Zhao, B., Fan, W., & Han, J. (2016). A survey on truth discovery. *ACM SIGKDD Explorations Newsletter*, *17*(2), 1–16. http://dx.doi.org/10.1145/2897350.2897352.

Liu, X.-Y., Wu, J., & Zhou, Z.-H. (2008). Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, *39*(2), 539–550. http://dx.doi.org/10.1109/TSMCB.2008.2007853.

Marshall, H. (1977). Newspaper accuracy in tucson. *Journalism Quarterly*, *54*(1), 165–169. http://dx.doi.org/10.1177/107769907705400127.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. In *Proceedings of the international conference on learning representations* (pp. 1–12).

Nakov, P., Corney, D., Hasanain, M., Alam, F., Elsayed, T., Barrón-Cedeño, A., Papotti, P., Shaar, S., & Da San Martino, G. (2021). Automated fact-checking for assisting human fact-checkers. In Z.-H. Zhou (Ed.), *Proceedings of the thirtieth international joint conference on artificial intelligence, IJCAI-21* (pp. 4551–4558). International Joint Conferences on Artificial Intelligence Organization, http://dx.doi.org/10.24963/ijcai.2021/619.

Newman, N., Fletcher, R., Kalogeropoulos, A., & Nielsen, R. (2019). Reuters institute digital news report 2019. *Reuters Institute for the Study of Journalism*.

Newman, N., Fletcher, R., Schulz, A., Andi, S., Robertson, C. T., & Nielsen, R. (2021). Reuters institute digital news report 2021. *Reuters Institute for the Study of Journalism*.

Nguyen, A. (2010). Harnessing the potential of online news: Suggestions from a study on the relationship between online news advantages and its post-adoption consequences. *Journalism*, *11*(2), 223–241. http://dx.doi.org/10.1177/1464884909355910.

Nie, Y., Chen, H., & Bansal, M. (2019). Combining fact extraction and verification with neural semantic matching networks. In *Proceedings of the AAAI conference on artificial intelligence, Vol. 33* (pp. 6859–6866). http://dx.doi.org/10.1609/aaai.v33i01.33016859.

O'Mahony, K. (2014). As it happens: how live news blogs work and their future. URL http://eprints.lse.ac.uk/56792/.

Oostdijk, N., Reynaert, M., Hoste, V., & Schuurman, I. (2013). The construction of a 500-million-word reference corpus of contemporary written dutch. In P. Spyns, & J. Odijk (Eds.), *Essential speech and language technology for dutch: results by the STEVIN programme* (pp. 219–247). Springer Berlin Heidelberg, http://dx.doi.org/10.1007/978-3-642-30910-6_13.

Pérez-Escoda, A., Pedrero-Esteban, L. M., Rubio-Romero, J., & Jiménez-Narros, C. (2021). Fake news reaching young people on social networks: Distrust challenging media literacy. *Publications*, *9*(2), 24. http://dx.doi.org/10.3390/publications9020024.

Peters, M. E., Ruder, S., & Smith, N. A. (2019). To tune or not to tune? Adapting pretrained representations to diverse tasks. In *Proceedings of the 4th workshop on representation learning for NLP* (pp. 7–14). Florence, Italy: Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/W19-4302.

Pomerleau, D., & Rao, D. (2017). Fake news challenge: exploring how artificial intelligence technologies could be leveraged to combat fake news. URL https://www.fakenewschallenge.org/.

Porlezza, C. (2019). Accuracy in journalism. *Oxford Research Encyclopedia of Communication*, http://dx.doi.org/10.1093/acrefore/9780190228613.013.773.

Porlezza, C., Maier, S. R., & Russ-Mohl, S. (2012). News accuracy in Switzerland and Italy: a transatlantic comparison with the US press. *Journalism Practice*, *6*(4), 530–546. http://dx.doi.org/10.1080/17512786.2011.650923.

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, *1*(1), 81–106. http://dx.doi.org/10.1007/BF00116251.

Rashkin, H., Choi, E., Jang, J. Y., Volkova, S., & Choi, Y. (2017). Truth of varying shades: Analyzing language in fake news and political fact-checking. In *Proceedings of the 2017 conference on empirical methods in natural language processing* (pp. 2931–2937).

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using siamese BERT-networks. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing* (pp. 3982–3992). Hong Kong, China: Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/D19-1410.

Reimers, N., & Gurevych, I. (2020). Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 conference on empirical methods in natural language processing* (pp. 4512–4525). Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/2020.emnlp-main.365.

Saltzis, K. (2012). Breaking news online: How news stories are updated and maintained around-the-clock. *Journalism Practice*, *6*(5–6), 702–710. http://dx.doi.org/10.1080/17512786.2012.667274.

Shaar, S., Babulkov, N., Da San Martino, G., & Nakov, P. (2020). That is a known Lie: Detecting previously fact-checked claims. In *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 3607–3618). http://dx.doi.org/10.18653/v1/2020.acl-main.332.

Shu, K., Sliva, A., Wang, S., Tang, J., & Liu, H. (2017). Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explorations Newsletter*, *19*(1), 22–36. http://dx.doi.org/10.1145/3137597.3137600.

Tandoc, E. C., Jr. (2014). Journalism is twerking? How web analytics is changing the process of gatekeeping. *New Media & Society*, *16*(4), 559–575. http://dx.doi.org/10.1177/1461444814530541.

Theodoridis, S., & Koutroumbas, K. (2006). *Pattern recognition*. Elsevier.

Thorne, J., & Vlachos, A. (2017). An extensible framework for verification of numerical claims. In *Proceedings of the software demonstrations of the 15th conference of the European chapter of the association for computational linguistics* (pp. 37–40). Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/E17-3010.

Thorne, J., & Vlachos, A. (2018). Automated fact checking: task formulations, methods and future directions. In *Proceedings of the 27th International Conference on Computational Linguistics* (pp. 3346–3359). Santa Fe, New Mexico, USA: Association for Computational Linguistics.

Thorne, J., Vlachos, A., Christodoulopoulos, C., & Mittal, A. (2018). FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 conference of the north American chapter of the association for computational linguistics: human language technologies, volume 1 (long papers)* (pp. 809–819). New Orleans, Louisiana: Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/N18-1074.

Thorne, J., Vlachos, A., Cocarascu, O., Christodoulopoulos, C., & Mittal, A. (2018). The fact extraction and verification (FEVER) shared task. In *Proceedings of the first workshop on fact extraction and verification (FEVER)* (pp. 1–9). Brussels, Belgium: Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/W18-5501.

Timmerman, Y., & Bronselaer, A. (2019). Measuring data quality in information systems research. *Decision Support Systems*, *126*, Article 113138. http://dx.doi.org/10.1016/j.dss.2019.113138.

Tsfati, Y., Boomgaarden, H. G., Strömbäck, J., Vliegenthart, R., Damstra, A., & Lindgren, E. (2020). Causes and consequences of mainstream media dissemination of fake news: literature review and synthesis. *Annals of the International Communication Association*, *44*(2), 157–173. http://dx.doi.org/10.1080/23808985.2020.1759443.

Tulkens, S., Emmery, C., & Daelemans, W. (2016). Evaluating unsupervised dutch word embeddings as a linguistic resource. arXiv http://dx.doi.org/10.48550/arXiv.1607.00225.

Usher, N. (2018). Breaking news production processes in US metropolitan newspapers: Immediacy and journalistic authority. *Journalism*, *19*(1), 21–36. http://dx.doi.org/10.1177/1464884916689151.

Vlachos, A., & Riedel, S. (2015). Identification and verification of simple claims about statistical properties. In *Proceedings of the 2015 conference on empirical methods in natural language processing* (pp. 2596–2601). Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/D15-1312.

Vo, N., & Lee, K. (2018). The rise of guardians: Fact-checking url recommendation to combat fake news. In *The 41st international ACM SIGIR conference on research & development in information retrieval* (pp. 275–284). http://dx.doi.org/10.1145/3209978.3210037.

Vosoughi, S., Roy, D., & Aral, S. (2018). The spread of true and false news online. *Science*, *359*(6380), 1146–1151. http://dx.doi.org/10.1126/science.aap9559.

Wang, W. Y. (2017). "Liar, liar pants on fire": A new benchmark dataset for fake news detection. In *Proceedings of the 55th annual meeting of the association for computational linguistics (volume 2: short papers)* (pp. 422–426). Vancouver, Canada: Association for Computational Linguistics, http://dx.doi.org/10.18653/v1/P17-2067.

Welbers, K., Van Atteveldt, W., Kleinnijenhuis, J., Ruigrok, N., & Schaper, J. (2016). News selection criteria in the digital age: Professional norms versus online audience metrics. *Journalism*, *17*(8), 1037–1053. http://dx.doi.org/10.1177/1464884915595474.

Widholm, A. (2016). Tracing online news in motion: Time and duration in the study of liquid journalism. *Digital Journalism*, *4*(1), 24–40. http://dx.doi.org/10.1080/21670811.2015.1096611.

Wilner, T., Wallace, R., Lacasa-Mas, I., & Goldstein, E. (2021). The tragedy of errors: Political ideology, perceived journalistic quality, and media trust. *Journalism Practice*, 1–22. http://dx.doi.org/10.1080/17512786.2021.1873167.

Zamith, R. (2017). Capturing and analyzing liquid content: A computational process for freezing and analyzing mutable documents. *Journalism Studies*, *18*(12), 1489–1504. http://dx.doi.org/10.1080/1461670X.2016.1146083.

Yoram Timmerman received the M.Sc. degree in computer science engineering in 2018 from Ghent University, Ghent, Belgium. Since then, he has been working toward the Ph.D. degree at the "Database, Document, and Content Management" research group under the supervision of Prof. A. Bronselaer. His research interests include data and information quality, data schema quality and data FAIRness.

Antoon Bronselaer received the M.Sc. degree in Computer Science and the Ph.D. degree in Engineering from Ghent University, Ghent, Belgium, in July 2006 and 2010, respectively. Since October 2006, he has been a researcher in the Department of Telecommunications and Information Processing in the research unit "Database, Document, and Content Management" at Ghent University. His research interests include data quality and temporal data management.