Journal of Big Data

METHODOLOGY Open Access

The adaptive community-response (ACR) method for collecting misinformation on social media

Julian Kauk[1*], Helene Kreysa[1], André Scherag[2,3] and Stefan R. Schweinberger[1,3,4]

*Correspondence:
julian.kauk@uni-jena.de

[1] Department of General Psychology and Cognitive Neuroscience, Friedrich Schiller University Jena, Am Steiger 3/1, 07743 Jena, Thuringia, Germany
[2] Institute of Medical Statistics, Computer and Data Sciences, Jena University Hospital, Bachstraße 18, Haus 1, 07743 Jena, Thuringia, Germany
[3] Michael Stifel Center Jena for Data-Driven and Simulation Science, Leutragraben 1, 07743 Jena, Thuringia, Germany
[4] German Center for Mental Health (DZPG), Site Jena-Magdeburg-Halle, Philosophenweg 3, 07743 Jena, Thuringia, Germany

Abstract

Social media can be a major accelerator of the spread of misinformation, thereby potentially compromising both individual well-being and social cohesion. Despite significant recent advances, the study of online misinformation is a relatively young field facing several (methodological) challenges. In this regard, the detection of online misinformation has proven difficult, as online large-scale data streams require (semi-) automated, highly specific and therefore sophisticated methods to separate posts containing misinformation from irrelevant posts. In the present paper, we introduce the adaptive community-response (ACR) method, an unsupervised technique for the large-scale collection of misinformation on Twitter (now known as 'X'). The ACR method is based on previous findings showing that Twitter users occasionally reply to misinformation with fact-checking by referring to specific fact-checking sites (crowdsourced fact-checking). In a first step, we captured such misinforming but fact-checked tweets. These tweets were used in a second step to extract specific linguistic features (keywords), enabling us to collect also those misinforming tweets that were not fact-checked at all as a third step. We initially present a mathematical framework of our method, followed by an explicit algorithmic implementation. We then evaluate ACR on the basis of a comprehensive dataset consisting of > 25 million tweets, belonging to > 300 misinforming stories. Our evaluation shows that ACR is a useful extension to the methods pool of the field, enabling researchers to collect online misinformation more comprehensively. Text similarity measures clearly indicated correspondence between the claims of false stories and the ACR tweets, even though ACR performance was heterogeneously distributed across the stories. A baseline comparison to the fact-checked tweets showed that the ACR method can detect story-related tweets to a comparable degree, while being sensitive to different types of tweets: Fact-checked tweets tend to be driven by high outreach (as indicated by a high number of retweets), whereas the sensitivity of the ACR method extends to tweets exhibiting lower outreach. Taken together, ACR's capacity as a valuable methodological contribution to the field is based on (i) the adoption of prior, pioneering research in the field, (ii) a well-formalized mathematical framework and (iii) an empirical foundation via a comprehensive set of indicators.

Keywords: Misinformation detection, Fake news detection, Twitter, Social media

Introduction

Social media has dramatically changed our information ecosystem, increasing the volume and speed of information flow exponentially. Despite several positive aspects of social media (e.g., connecting individuals across large physical distances, stimulating democracies by facilitating public discourse or serving (local) communities as common ground for communication; see, e.g., [2, 5, 68]), there are also negative side effects. Those range from reduced psychological well-being (e.g., due to cyberbullying or social media addiction; see, e.g., [5, 15, 36, 54]) to more broad, societal effects (e.g., disproportionally powerful social media companies, overaccelerated public discourse or formation of legal vacuums; see, e.g., [27, 39, 73]).

A major concern regarding social media, the spread of misinformation, has become a significant global challenge. Here, we understand misinformation as any kind of false or inaccurate information, regardless of the intent to deceive individuals (see, e.g., [85, 30]). Several events, including the 2016 US presidential election (see, e.g., [3, 9]), the COVID-19 pandemic [34, 63, 76] and the evolving climate crisis [41, 44, 79] have taught us that underestimating this phenomenon may have substantial and unforeseeable consequences. A growing body of research indicates that misinformation can adversely affect social cohesion, adherence to public health measures, integrity of democracies, or trust in elected representatives or the press (see [33, 58, 69, 76, 77]; but see also [83]). Scholars currently conclude that addressing the spread of misinformation requires multidisciplinary, large-scale societal and research efforts (see, e.g., [37, 39]).

Social media play a determining role in the amplification of misinformation (see, e.g., [3, 4, 17, 37, 73]), thus challenging researchers to better understand the dynamics and mechanisms underlying the spread of misinformation, despite its complex and multifaceted nature [39, 40, 64]. Pioneering works on the diffusion of misinformation through social media have helped to improve our understanding of the mechanisms that likely underlie this phenomenon (see, e.g., [4, 25, 31, 37, 65, 82]), but the field is still confronted with significant questions and (methodological) challenges: For instance, there is still no broadly accepted conceptual framework of terms related to the issue (see, e.g., [30, 85]), nor are there sophisticated answers to the relevant question of how common misinformation actually is [37] and what its short- and long-term effects are [83].

One reason for these gaps in knowledge is that there is thus far no broadly accepted pool of methods to collect and analyze online misinformation data. This point was raised by Camargo and Simon [14], who explicitly called for more methodological rigor in the field. Single-case studies and studies with selective and/or small datasets can yield important insights about certain phenomena, but may fail to provide robust and comprehensive evidence with predictive power concerning relevant research questions. Collecting representative and large-scale misinformation datasets can be considered a crucial prerequisite for the generation of knowledge, but this prerequisite is often violated in the field [82].

Several approaches have been proposed to detect misinformation on social media (for overviews, see, e.g., [55, 67, 85]), each focusing on different features of misinformation: techniques may exploit (i) content-based features (e.g., application of fact-checking allows the credibility of a post to be scored; see, e.g., [67, 82]), (ii) style-based features of the posts (e.g., heavy use of adverbs; see, e.g., [1]) or (iii) the social context

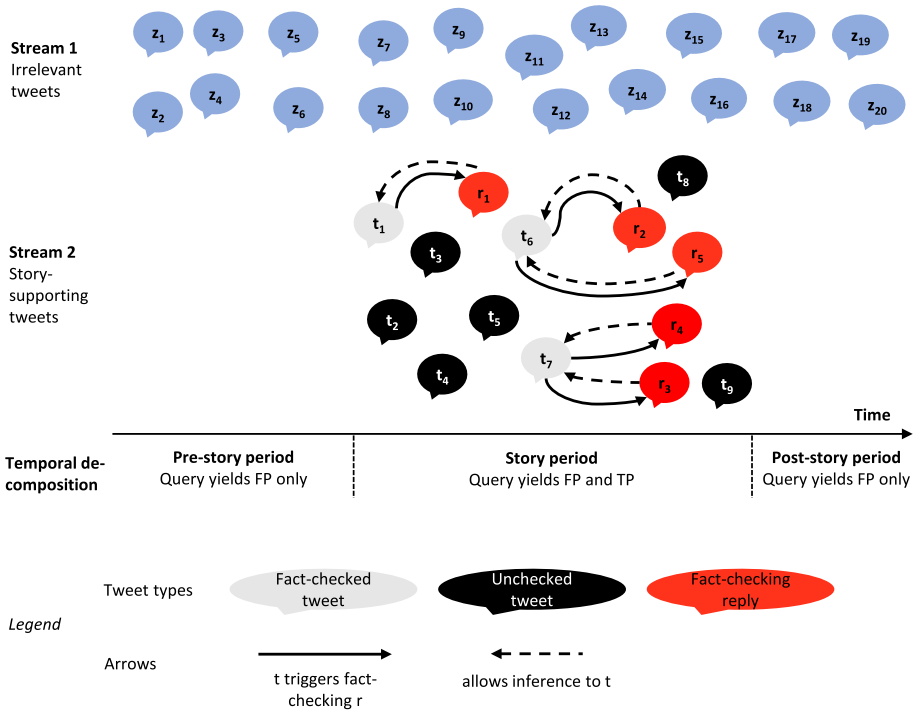

Fig. 1 Our sketch of the (adaptive) community-response approach. Note that t and r stand for (story-supporting) tweet and reply, respectively. Here, tweets t_1, t_6 and t_7 can be identified via fact-checking ('seen' tweets). However, the community-response approach fails for t_2, t_3, t_4, t_5, t_8 and t_9 ('unseen' tweets), because there were no fact-checking responses

(e.g., user profiles or reactions to the posts; see, e.g., [18, 32]). However, each approach is confronted with the challenge of avoiding overload with false positive posts. Overload reduction is particularly important when dealing with social data, as such data streams typically involve millions of posts being posted within days or even hours (in terms of Twitter, more than 200 million tweets per day were posted in some years [38]). Selecting posts supporting a misinformation story is therefore not trivial [67], and requires a sophisticated filtering of story-supporting posts from the immense body of irrelevant posts not concerning or not supporting this story.

Vosoughi et al. [82] proposed an approach to collect misinforming tweets on Twitter by exploiting both content-based features and the social context. This approach may be named the 'community-response approach' and relies on the assumption that story-supporting tweets are subsequently fact-checked by other users (crowdsourced fact-checking) via replying ('reply': comment to a tweet) tweets[1] (Fig. 1, Stream 2, red ovals). Those fact-checking tweets often contain links to fact-checking sites; by searching for such links in replies, the original, story-supporting tweets can be identified (Fig. 1, Stream 2, light gray ovals).

[1] Technically speaking, a tweet can also be fact-checked via a quoting tweet ('quote': comment to a tweet while repeating the tweet). For ease of readability, we will refer to replies only.

Even though the community-response approach represents a valuable method to collect misinforming tweets, it also comes with certain pitfalls, as it fails to detect story-supporting tweets that are *not* fact-checked at all during the dissemination of a story[2] (Fig. 1, Stream 2, black ovals). Thus, the community-response approach probably neglects a substantial proportion of tweets supporting a story due to the absence of crowd fact-checking. Presumably, tweets that are not fact-checked play an important role, as there might be users who post story-supporting tweets without having any (critical) followers to correct them. Furthermore, the findings of Cinelli et al. [17], that Twitter users tend to agglomerate within homogeneous clusters, may also have consequences for the validity of the community-response approach: If there are few or no critical users within a cluster, tweets probably remain in an unchecked state.

To fill this gap in methodology, we propose a method that should capture both crowd fact-checked tweets and tweets without any fact-check reply. More precisely, our approach aims to additionally detect the 'unseen' tweets that were not fact-checked at all (Fig. 1, Stream 2, black ovals). Here, we refer to this novel approach as the adaptive community-response (ACR) method - a research method to automatically collect large-scale datasets of misinforming tweets. The ACR method is inspired by the approach proposed by Vosoughi et al. [82] but extends this approach substantially, as will be explained in the following sections. Initially, we will provide a mathematical framework of the ACR method. Next, we describe the implementation and finally, we evaluate our method on the basis of a large-scale tweet collection, consisting of more than 25 million tweets belonging to more than 300 misinformation stories.

Mathematical framework of the ACR method

As the ACR method is essentially concerned with binary classification, we initially provide the set-theoretical basics of the method. Subsequently, we describe the assumptions on which the method is based, followed by an explication of what is implicated by these assumptions.

Set-theoretical basics

We consider a set of story-supporting tweets

$$T = \{t_1, t_2, \ldots, t_n\}, \tag{1}$$

where t_i stands for the ith story-supporting tweet. We aim to collect as many tweets as possible that truly belong to T, while minimizing the number of irrelevant tweets (tweets that do *not* belong to T). We assume that these irrelevant tweets belong to another set of tweets

$$Z = \{z_1, z_2, \ldots, z_n\}, \tag{2}$$

where z_i is the ith tweet not concerning the story. It holds that $T \cap Z = \emptyset$.

The main problem in the context of tweet retrieval is the enormous number of competing tweets in Z compared to T, because the prevalence of T is (often extremely) low.

[2] Another scenario is that tweets might be fact-checked but without providing links to fact-checking sites. We refrained from taking this scenario into account in the interest of readability.

Consequently, when searching Twitter for T via a query, even a query that seems to have 'high' specificity can yield numerous false positives. An explicit model and example of the low prevalence problem is provided in Additional file 1 which provides supporting information to improve understanding of the upcoming sections.

Any query q_i passed to Twitter will yield a set of tweets, which can be decomposed into story-supportive (true positives; $\mathrm{TP}(q_i)$) or irrelevant (false positives; $\mathrm{FP}(q_i)$) tweets. Tweets that do *not* match the query can also be either story-supportive (false negatives; $\mathrm{FN}(q_i)$) or irrelevant (true negatives; $\mathrm{TN}(q_i)$). It holds that

$$\begin{aligned} &\mathrm{TP}(q_i)\dot{\cup}\mathrm{FN}(q_i) = T \text{ and} \\ &\mathrm{TN}(q_i)\dot{\cup}\mathrm{FP}(q_i) = Z. \end{aligned} \tag{3}$$

The decomposition of matching ($Pos(q_i)$) tweets is therefore given by

$$\mathrm{Pos}(q_i) = \mathrm{TP}(q_i)\dot{\cup}\mathrm{FP}(q_i). \tag{4}$$

The number of tweets yielded by query q_i is

$$\begin{aligned} N(q_i) &= |\mathrm{Pos}(q_i)| \\ &= |\mathrm{TP}(q_i)\dot{\cup}\mathrm{FP}(q_i)| \\ &= |\mathrm{TP}(q_i)| + |\mathrm{FP}(q_i)|. \end{aligned} \tag{5}$$

Evaluating queries' performance

To evaluate the performance of any query q_i, we considered both recall and precision. Recall, reflecting the fraction of story-supporting tweets being detected, is given by

$$\mathrm{Recall}(q_i) = \frac{|\mathrm{TP}(q_i)|}{|T|}. \tag{6}$$

Precision is the fraction of story-supporting tweets yielded by q_i and is given by

$$\mathrm{Precision}(q_i) = \frac{|\mathrm{TP}(q_i)|}{N(q_i)}. \tag{7}$$

To keep data 'clean', it makes sense to define a lower bound of precision. Here, we considered a query to be valid only if it holds that precision is equal to or greater than 0.9. A corresponding indicator function is therefore given by

$$\mathrm{Precise}(q_i) = \begin{cases} 1 & \text{if } \mathrm{Precision}(q_i) \geq 0.9 \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

Thus, we consider data that contain less than 90% true positive tweets as so noisy that analyses would be inappropriate. This strict precision criterion may lead to a loss of recall, due to the precision-recall trade-off (see, e.g., [11]), which describes a negative relationship between recall and precision. At the same time, a decrease in recall may be considered to be 'acceptable', as a story typically involves thousands of tweets, such that even a subset of them should be representative according to the law of large numbers.

Assumptions

Assumption i: Content-representativity

As illustrated in Fig. 1, Vosoughi et al.'s approach can detect only a subset of T, because not every story-supporting tweet is fact-checked. Accordingly, T can be decomposed into two distinct subsets

$$T = T_s \dot{\cup} T_u, \tag{9}$$

where T_s is the part 'seen' by the community-response-approach and T_u is the 'unseen' part. The community-response-approach captures T_s, while T_u remains 'hidden'. The ACR method is based on the assumption that the 'seen' tweets T_s are representative regarding the content of the 'unseen' tweets T_u. This assumption might not hold under all circumstances (e.g., the language used in a homogeneous cluster could differ with regard to the hashtags and keywords that are used when spreading a story).

Assumption ii: Temporal specificity

Temporal specificity means that tweets supporting a story only occur in a specific time window. Accordingly, tweets before and after this time window by definition can *not* belong to the story. Figure 1 illustrates this idea: During the story period, there are both story-supporting tweets and unrelated (irrelevant) tweets (see Streams 1 and 2). In the pre- and post-story period, by definition, only irrelevant tweets can occur. We introduce the following notations:

$$\begin{aligned} \text{Time point when } t_i \text{ was posted} &: \dot{t}_i, \\ \text{Vector of tweets' time points} &: \dot{T} = (\dot{t}_1, \dot{t}_2, \ldots, \dot{t}_n), \\ \text{Onset of the story} &: \dot{T}^o = \min(\dot{T}), \\ \text{End of the story} &: \dot{T}^e = \max(\dot{T}) \text{ and} \\ \text{Lifetime of the story} &: \dot{T}^\Delta = \dot{T}^e - \dot{T}^o. \end{aligned} \tag{10}$$

We assume that $\dot{T}^\Delta$ is nonzero and finite. We therefore explicitly assume that the story is 'closed', meaning that it is no longer present. This assumption may not perfectly resemble real-world relationships, as a story could - in theory - reemerge at any time.

Assumption iii: Temporal homogeneity of irrelevant tweets

This assumption refers to the stream of irrelevant tweets (Stream 1 in Fig. 1) and means that the statistical properties of the irrelevant tweets are time-independent. Essentially, this means that the 'background noise' does not change over time. Note that this is a very strict assumption, which might well be violated, for instance when general aspects of the public discourse change in the long-run or due to unpredicted global events (e.g., the emergence of a pandemic).

Implications of these assumptions

Estimating queries' recall

Under assumption i, it can be inferred that the number of matching tweets in T_s for a query q_i is an estimator for recall of q_i:

$$\begin{aligned}\widehat{\text{Recall}(q_i)} &= f(T_s; q_i) \\ &= \frac{|T_{s_{q_i}}|}{|T_s|},\end{aligned} \tag{11}$$

where f stands for 'relative frequency' and $T_{s_{q_i}}$ refers to all tweets in T_s matching query ($q_i \subseteq T_s$).

Estimating queries' precision

From Eq. 4 we know that the tweets yielded by a query can be decomposed into true positives and false positives. As we assume that the story only occurs in a specific time window (Assumption ii), we can therefore also extend the equation to be bounded to a specific time window:

$$\begin{aligned}\text{Pos}(q_i) &= \text{Pos}(q_i; \dot{T}^o, \dot{T}^e) \\ &= \text{TP}(q_i; \dot{T}^o, \dot{T}^e) \dot{\cup} \text{FP}(q_i; \dot{T}^o, \dot{T}^e).\end{aligned} \tag{12}$$

It therefore also holds that (under consideration of Eq. 5):

$$\begin{aligned}N(q_i) &= N(q_i; \dot{T}^o, \dot{T}^e) \\ &= |\text{TP}(q_i; \dot{T}^o, \dot{T}^e) \dot{\cup} \text{FP}(q_i; \dot{T}^o, \dot{T}^e)| \\ &= |\text{TP}(q_i; \dot{T}^o, \dot{T}^e)| + |\text{FP}(q_i; \dot{T}^o, \dot{T}^e)|.\end{aligned} \tag{13}$$

However, when we consider a baseline period b_j, as defined by two points in time $b_j = (b_j^o, b_j^e)$ that do *not* fall into the story period ($b_j^o, b_j^e \in \{x | x < \dot{T}^o \vee x > \dot{T}^e\}$), and that has the same timespan as the story ($b_j^\Delta = b_j^e - b_j^o = \dot{T}^\Delta$), the results of a query can be decomposed according to:

$$\text{Pos}(q_i; b_j^o, b_j^e) = \text{TP}(q_i; b_j^o, b_j^e) \dot{\cup} \text{FP}(q_i; b_j^o, b_j^e)$$

Under the assumption of temporal specificity, it holds that $\text{TP}(q_i; b_j^o, b_j^e) = \emptyset$, because we assume that the extratemporal occurrence of true positives is impossible. Thus, it holds that

$$\begin{aligned}\text{Pos}(q_i; b_j^o, b_j^e) &= \emptyset \dot{\cup} \text{FP}(q_i; b_j^o, b_j^e) \\ &= \text{FP}(q_i; b_j^o, b_j^e).\end{aligned} \tag{14}$$

As assumption iii implies that the (statistical) properties of Z remain constant over time, the cardinality of $\text{FP}(q_i; b_j^o, b_j^e)$ should be an approximation of $|\text{FP}(q_i; \dot{T}^o, \dot{T}^e)|$:

$$\begin{aligned}\widehat{|\text{FP}(q_i; \dot{T}^o, \dot{T}^e)|} &= |\text{FP}(q_i; b_j^o, b_j^e)| \\ &= |\text{Pos}(q_i; b_j^o, b_j^e)|.\end{aligned} \tag{15}$$

Consequently, after solving Eq. 13 for $|\mathrm{TP}(q_i; \dot{T}^o, \dot{T}^e)|$, we can obtain an estimation for the number of true positives, as $N(q_i; \dot{T}^o, \dot{T}^e)$ is known and $|\mathrm{FP}(q_i; \dot{T}^o, \dot{T}^e)|$ is estimable:

$$\begin{aligned} |\widehat{\mathrm{TP}(q_i)}| &= N(q_i; \dot{T}^o, \dot{T}^e) - |\mathrm{FP}(\widehat{q_i; \dot{T}^o}, \dot{T}^e)| \\ &= N(q_i; \dot{T}^o, \dot{T}^e) - |\mathrm{Pos}(q_i; b_j^o, b_j^e)|. \end{aligned} \tag{16}$$

The precision of query q_i can therefore be estimated via

$$\begin{aligned} \widehat{\mathrm{Precision}}(q_i) &= \frac{\widehat{\mathrm{TP}(q_i)}}{N(q_i)} \\ &= \frac{N(q_i; \dot{T}^o, \dot{T}^e) - |\mathrm{Pos}(q_i; b_j^o, b_j^e)|}{N(q_i; \dot{T}^o, \dot{T}^e)}. \end{aligned} \tag{17}$$

Consequently, we are able to estimate both the recall and precision of a query (under assumptions i-iii). This means that we can evaluate any query in terms of its performance and select the best queries according to Eq. 8.

Implementing the ACR method

The following paragraphs explain in greater detail how we implemented the ACR method. If not indicated otherwise, we used PYTHON (version 3.10.2) for data retrieval and analyses.

General methodological information

Data collection tools

Twitter allowed researchers until March 2023 to collect up to 10 million tweets per month via the Academic Research product track (see [78]). Tweets can be requested via the full history search endpoint, which returns all the tweets (in a specified time period) matching a specific query. These queries are keyword-based, meaning that the query OBAMA WHITE HOUSE will match all tweets containing OBAMA, WHITE and HOUSE. There is, however, also an endpoint option that simply outputs the number of matching tweets for a query (Tweets count endpoint), without retrieving the tweets itself (see [80]). We used the library TWEEPY (version: 4.6; see [62]) to fetch Twitter data via the Twitter application programming interface (API).

We only considered English tweets; thus, when requesting the Twitter API, we always attached the LANG:EN command to the query. We defined a general time window of interest starting from January 01, 2007, until October, 31, 2022. This means that we considered Twitter activity in a time window of almost 15 years, representing a comprehensive time period including various global and local historical events.

Tweet preprocessing

We preprocessed the collected tweets by removing unfavorable characters to obtain standardized, usable tweets. We removed URLs, several irregular characters (regular expression of *allowed* characters: [A-ZA-Z0-9) and transformed the tweet to lowercase. We decided to keep mentions (@) and hashtags (#) in the tweets (but without

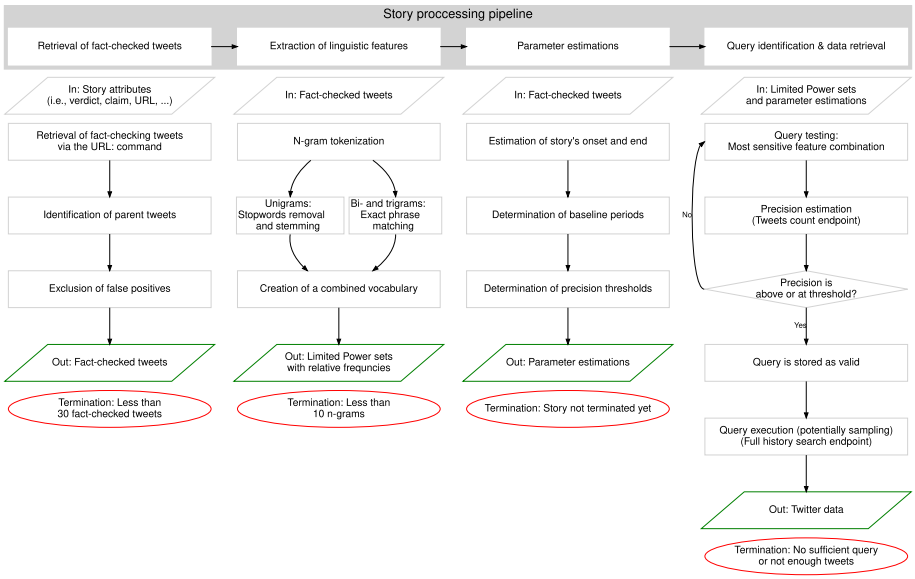

Fig. 2 A simplified story processing pipeline of the ACR method. Please note that processes before (e.g., retrieval of stories and their fact-checks) and after (i.e., Evaluation of the tweets) this pipeline are not shown, but are presented in the main text

@ and #), as they are often integral to the tweets' sentences (e.g., replacing Trump with @realdonaldtrump).

Retrieval of stories and their fact-checks

Twitter users rely on different fact-checking pages to validate tweets, mainly depending on the topic of the respective story. To keep things simple, we only considered two major fact-checking pages (Snopes and PolitiFact; see [57, 71]) We selected these pages because they are known to (i) be highly prevalent on Twitter, (ii) propose clear and quantifiable verdicts about the checked stories [67, 87] and (iii) conduct independent and high-quality fact-checking. Both sites were also used by [82]; they predominantly fact-check stories related to the US [19].

We relied on the ClaimReview system, which is a tagging system for fact-checking organizations to flag their fact-checks for search engines and social networking services (see [20]). Based on this system, Google built the Google Fact Check Tool API (see [24]), which provides access to a relatively comprehensive database of fact-checks of several fact-checking sites (including our sites of interest) in a standardized, JSON-based form. Importantly, this API provides, alongside the link of the fact-checking article, information about (i) the claim that is examined in the fact-check, (ii) the textual verdict of the fact-checkers and (iii) who started the claim. In accordance with Vosoughi et al., we mapped the textual ratings to a five-level Likert scale, ranging from heavily false ($\equiv 1$) to mixed ($\equiv 3$) to completely correct ($\equiv 5$). We only considered false ($\equiv 2$) and heavily false ($\equiv 1$) stories.

Story processing pipeline

The following sections describe how each of the stories retrieved via the ClaimReview system was processed to collect tweets related to the respective story. To better represent this algorithm, the reader is referred to Fig. 2.

Retrieval of fact-checked tweets T_s

Retrieval of fact-checking tweets. Subsequently, we searched Twitter for replies and quotes (command: IS:REPLY OR IS:OUOTE) containing links to the stories evaluated by our fact-checking pages via the URL: command. This allowed us to collect fact-checking tweets and to assign these tweets to specific stories. We resolved links in advance, meaning that we identified other links circulating on Twitter pointing toward fact-checking articles. This was necessary, as one and the same fact-checking article may be reached via different links (due to, e.g., migration of the content management system). We used the GET function of the REQUESTS library for link resolving.

Retrieval of fact-checked tweets. We subsequently used the fact-checking tweets and their REFERENCED_TWEETS field (see [81]) to retrieve the parent tweets, i.e., the fact-checked tweets. In accordance with Vosoughi et al., we excluded parent tweets that (i) were not in English or (ii) themselves contained a link to a fact-checking page. We slightly diverged from Vosoughi et al.'s approach by also taking into account tweets that are not first level, i.e., we also considered replies to replies. This allowed us to capture more data, aiming to increase the accuracy of the ACR method. Analyses were stopped whenever the number of fact-checked tweets fell below $n = 30$.

Feature extraction from T_s and creation of (limited) power sets

The fact-checked tweets T_s were subsequently used to build a vocabulary, enabling us to identify keywords (or keyword combinations) that are sensitive regarding the respective story (according to Eq. 11). We considered uni-, bi- and trigrams, although the preprocessing methods differed slightly between both unigrams and bi- and trigrams (see upcoming sections). N-grams had to occur in at least 5% of the tweets, but at least three times, to remove insignificant features.

Unigrams. We initially removed stopwords (taken from NLTK.CORPUS) and performed word stemming (using the PORTERSTEMMER from NLTK.STEM; for the NLTK library, see [8]). Both steps were mandatory, as (i) stopwords are extremely unspecific and (ii) tweets may contain (many) inflected words with the same word stem. Twitter API, however, does not allow searching for word stems (e.g., GET*), so instead we searched for all the collected inflected words by combining them disjunctively (e.g., GETS OR GETS OR GETTING). We then used the COUNTVECTORIZER function (from SKLEARN.FEATURE_EXTRACTION.TEXT.text; see [53]) with the parameter BINARY = TRUE to build the vocabulary for the unigrams, resulting in a matrix of unigrams' occurrence.

Bi- and trigrams. Different from the procedure for the unigrams, we did not perform (initial) stopword removal or word stemming, because we wanted to maintain the 'genuineness' of the phrases used. Thus, we passed the tweets directly to the COUNTVECTORIZER function (again, BINARY = TRUE) to build the vocabulary

for bi- and trigrams. However, we removed bi- and trigrams that started or ended with a stopword (as suggested by [21]). We performed exact phrase matching for bi- and trigrams, meaning that the queries passed to the Twitter API would only match tweets containing the full phrase (e.g., OBAMA IS INJURED).

Union of vocabularies and creation of (limited) power sets. Finally, we combined the unigrams' vocabulary with the vocabulary for bi- and trigrams. We ordered the n-grams according to their relative frequency and considered only the 50 most frequent n-grams (subsequently denoted as features F) for further analyses. Execution was cancelled if fewer than ten features were extractable.

We subsequently created power sets of limited cardinality $\mathcal{P}_k(F)$ for the extracted features using the COMBINATIONS function of the ITERTOOLS library. We considered subsets up to a cardinality of $k = 4$, meaning that a query passed to the Twitter API could contain up to four (conjunctively combined) features (e.g., "WHITE HOUSE" OBAMA EXPLOSIONS INJURED). A cardinality of $k = 4$ was chosen because creation of subsets with higher cardinality turned out to be computationally demanding (due to the high number of possible combinations), and because queries with more than four features may be too specific to fetch a useful number of tweets. We removed unigram subsets with $k = 1$ (e.g., OBEMA), since single words are rarely specific enough regarding the respective story. We subsequently computed the relative frequency (being an estimator for recall, see Eq. 11) of each subset in the fact-checked tweets T_s and ordered the subsets in descending order.

Estimating stories' lifetime

As stated previously, the ACR method assumes that any story only occurs in a specific time period. Theoretically, the onset of the story $\dot{T}^o$ should correspond to the first tweet spreading the story, while the end of the story $\dot{T}^e$ can be defined as the last tweet supporting the story. As we do *not* precisely know when these tweets were posted, we had to estimate both the onset and the end of the story. For both estimations, we relied on temporal information provided by the set of fact-checked tweets T_s .

Estimation of the onset was performed by using the time point of the first tweet in T_s ($\dot{T}_s^o$) as a first guess for $\dot{T}^o$. It is, however, reasonable to assume that fact-checking requires a certain amount of time to be conducted, implying that there might be a lag between $\dot{T}_s^o$ and $\dot{T}^o$. Consequently, $\dot{T}^o$ was estimated via

$$\widehat{\dot{T}^o} = \dot{T}_s^o - \Delta^{\leftarrow}, \tag{18}$$

while $\Delta^{\leftarrow}$ stands for the backward buffer, as given by $\Delta^{\leftarrow} = 60$ days. We therefore assumed that the true onset of the story could be up to 60 days before the first fact-checked tweet was posted (backward buffer), giving both fact-checkers and users a reasonable amount of time to conduct and spread fact-checking.

Estimating the end of a story is arguably even more difficult. First, a story may never completely 'die out', but becomes so infrequent that it can be considered terminated. Determining this point of *factual* end may be challenging. Second, $\dot{T}_s^e$ might not be a

good estimator for $\dot{T}^e$, as a story may reappear multiple times on Twitter (see [66]), but potentially not implicitly accompanied by community fact-checking responses. To take both points into account, we estimated $\dot{T}^e$ via

$$\widehat{\dot{T}^e} = Q_{0.95}(\dot{T}_s) + \Delta^{\rightarrow}, \tag{19}$$

where $Q_{0.95}(\dot{T}_s)$ stands for the 95^{th} percentile of T_s and $\Delta^{\rightarrow}$ for the forward buffer, as given by $\Delta^{\rightarrow} = Q_{0.95}(\dot{T}_s) - \dot{T}_s^o$. $Q_{0.95}(\dot{T}_s)$ was used to determine the factual end of the fact-checked tweets, therefore also resembling the end of fact-checking efforts. As this time point may not correspond to the factual end of the story, we added a forward buffer by simply doubling the timespan. Equation 20 should therefore account both for the problem of identifying a factual end and for the (potential) problem of reoccurring misinformation (without fact-checking responses). We restricted the maximum observation period to 365 days, as the ACR method may work less efficiently when considering very long time periods (as temporal specificity then may be weakened).

Selecting baseline periods and precision thresholds

To estimate queries' precision, we had to perform the queries in extratemporal time periods. According to Eq. 14 and 16, the cardinality of extratemporal queries should estimate the number of false positives. However, these equations are derived from assumptions ii (temporal specificity) and iii (homogeneity of irrelevant tweets) and may not hold under all circumstances. In particular, assumption iii may prove to be too strict for the 'real' Twitter world, as public discourse may change slowly or rapidly due to societal change or sudden (global) events (e.g., the onset of a pandemic). Such a violation of assumption iii arguably leads to an underestimation of the number of false positives and, vice versa, to precision overestimation. To compensate for this, we used (i) relatively strict precision thresholds, (ii) multiple baseline periods and (iii) penalties for the precision thresholds if not all baseline periods were accessible.

Baseline periods

Using multiple baseline periods may be considered the most effective tool in addressing violation of assumption iii, as they increase the likelihood that at least one of them validly reproduces the background noise during the story period. We considered up to $n = 4$ baseline periods (but at least two), depending on whether these periods were available (a baseline period may be not available because Twitter was, for instance, not launched at this time). A baseline period b_j is defined via two points in time b_j^o and b_j^e, both extratemporal and with the same duration as the story (we used $\widehat{\dot{T}^{\Delta}} = \widehat{\dot{T}^e} - \widehat{\dot{T}^o}$ to estimate the duration). Table 1 shows the relevant baseline periods and explains why they were used.

Precision thresholds and penalties

To keep data 'clean', we used the precision threshold defined in Eq. 8, which means that queries with a precision smaller than 0.9 were not considered. To address potential violations of assumption iii, we decided to differentiate precision thresholds for the single

Table 1 Summary of the four baseline periods and the story period (in chronological order)

j	Name	Description	Onset and end
1	b_1: Four years pre-story period	Since many fact-checks concerned the US (elections), this baseline period should represent a possible election period (a term lasts four years in the US).	$b_1^o = \widehat{t^o} - 4$ years $b_1^e = \widehat{t^e} - 4$ years
2	b_2: Two years pre-story period	This baseline period was used to account for seasonality, meaning that it would compensate for false positives that only occur due to seasonal effects.	$b_2^o = \widehat{t^o} - 2$ years $b_2^e = \widehat{t^e} - 2$ years
3	b_3: Close pre-story period	This baseline period was located as close as possible before the onset of the story (but, ideally, without overlapping). This reduces the likelihood of violating assumption iii, as ample changes in public discourse arguably occur less often in a shorter time period.	$b_3^e = \widehat{t^o} - 30$ days $b_3^o = b_3^e - \widehat{t^\Delta}$
–	Story period s	Time period where the story spread. Everything before and after this period is considered to be extratemporal.	Onset: $\widehat{t^o}$ End: $\widehat{t^e}$ Duration: $\widehat{t^\Delta}$
4	b_4: Post-story period	This baseline period was located after the (estimated) end of the story. It should compensate for very sudden changes in public discourse, which cannot be captured by pre-story periods.	$b_4^o = \widehat{t^e} + \frac{1}{4} \cdot \Delta \rightarrow b_4^e = b_4^o + \widehat{t^\Delta}$

For all baseline periods, it holds that: $b_j^e - b_j^o = \widehat{t^\Delta}$

baseline periods (Precision(q_i; b_j)) as well as for the overall precision, which is defined as the mean of all single baseline periods. Both Precision(q_i; b_j) and the overall precision are estimators for the true precision Precision(q_i).

We defined that queries had to be at least 90% precise for single baseline periods, while the overall precision had to be even greater, with at least 95%. The application of such strict criteria presumably ensures that even if assumption iii is somewhat violated, the queries should nevertheless be sufficiently precise.

Additionally, we also applied precision penalties, depending on the number of available baseline periods. We performed this step because it is reasonable to assume that precision estimations are more likely to be biased if not all baseline periods are available. We used a geometric series to represent the penalty, by halving the distance to the maximum precision of 1 with a missing baseline period[3].

Query identification and data retrieval

We used an iterative procedure to identify queries that could detect story-supporting tweets. Even though we only considered a limited power set of keyword combinations, the number of possible combinations was expected to be high. As certain Twitter API endpoints have limited numbers of requests per time unit (usually 15 min), we tested only a subset of all possible queries. We used the Tweet counts endpoint (as mentioned

[3] The penalty is formally defined as $\text{Penalty}(B) = \sum_{n=1}^{4-|B|} (\frac{1}{2})^n \cdot (1 - \text{Threshold})$. Threshold and B stand for the respective threshold (either 0.9 or 0.95) and the set of available baseline periods (e.g., $B = \{b_1, b_2, b_3\}$), respectively.

in Section 3.1.1) to determine the number of tweets matching our queries. Testing only a subset of queries is reasonable, as it reduces the risk of selecting false positive queries. We tested at maximum 50 queries per story (attempts), ordered by (estimated) recall (decreasing). If a query was considered to be valid (according to Eq. 8, and under consideration of precision penalties), it was stored as a valid query. We then continued to search for more queries satisfying the criteria, but reduced the number of attempts per round (according to a geometric series). This procedure of early stopping was implemented to reduce the risk of false-positive queries. If the algorithm identified multiple valid queries, those were combined disjunctively (OR operator) to increase recall. We performed query concatenation only if it led to measurable increase of query performance, as indicated by $F_{0.5}$[4]. When a query consists of multiple, disjunctively combined queries, we denote them as subqueries.

Finally, we conducted a full-archive search to retrieve matching tweets. Due to a hard limit of 10 million tweets per month, we performed tweet sampling if the number of tweets matching a query exceeded 100k. The number of matching tweets $|\mathrm{Pos}(q_i)|$ was again determined by the Tweet counts endpoint; subsequently, we calculated a downsampling coefficient, as given by $\alpha = \frac{100\mathrm{k}}{|\mathrm{Pos}(q_i)|}$. This coefficient was then used to determine the number of tweets that should be retrieved for every single day of the story period, thereby maintaining representativity.

Evaluation

To evaluate the ACR method, we considered a set of indicators, which will be explained in further detail in the following sections. Those indicators were deducible from the proposed mathematical framework and contribute to validating the ACR method. Initially, however, we present a set of descriptive statistics to objectively characterize our dataset.

Descriptive statistics

Our dataset consisted of 7091 stories that were either fact-checked by Snopes (2097 stories), PolitiFact (4981 stories) or both (13 stories). A majority of these stories ($N = 5488$) had a verdict score of 1, meaning that they were considered to be heavily false, while the remaining stories had a verdict score of 2 (false stories; $N = 1603$). Most stories were concerned with the United States and its politicians (see Additional file 2 Fig. 1), but such a large dataset arguably covers a diverse set of topics.

Replies

It is important to note that probably only a minority of these stories played a significant role on Twitter: Links to the respective fact-checking sites were found in $2.07 \cdot 10^5$ replies, with an average of 29.22 replies per story. Importantly, we found that the distribution of the number of replies was highly right-skewed, which was also expressed by the median $\mathrm{Md} = 4$, as well as by the fact that 1589 stories (22.41%) did not occur

[4] F_β is a measure which takes into account both recall and precision (see, e.g., [29]), where β controls how much weight is given to recall and precision. In the context of tweet retrieval, it makes sense to weight precision more heavily due to the outlined low prevalence problem. A common approach is to weight precision as twice as important as recall, corresponding to $\beta = 0.5$. F_β is then given by $F_{0.5}(q_i) = \frac{5}{4} \cdot \frac{\mathrm{Precision}(q_i) \cdot \mathrm{Recall}(q_i)}{\frac{1}{4} \cdot \mathrm{Precision}(q_i) + \mathrm{Recall}(q_i)}$.

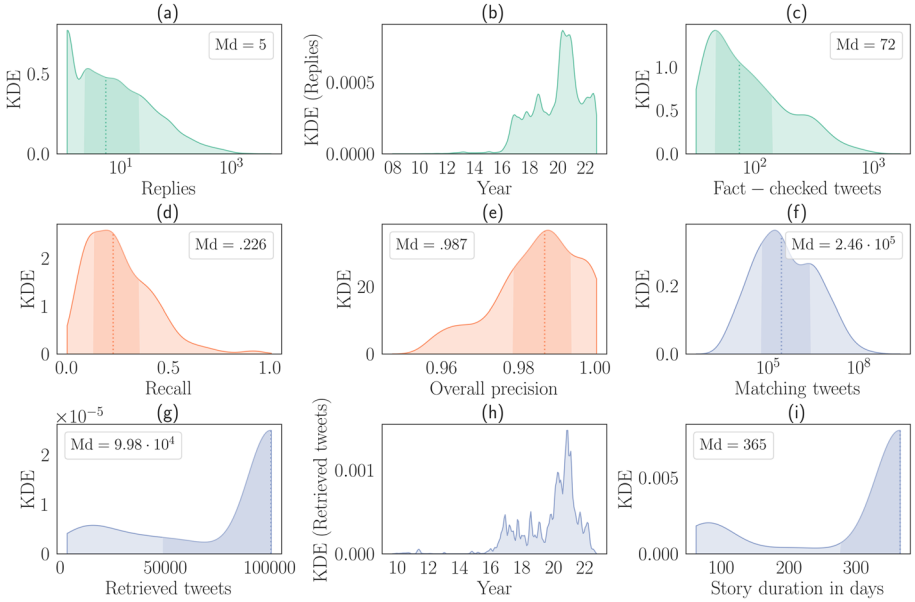

Fig. 3 Relevant distributions of our dataset. Please note the following points: First, we performed kernel density estimations (KDEs (kernel density estimations (KDEs; bandwidth selection according to Scott's rule) to approximate the underlying probability density functions. Second, the dashed line and the colored area reflect the median and the interquartile range (IQR), respectively. Third, the full dataset, i.e., before story exclusion (see Additional file Table view, Fig. 2), was used for panels **a** and **b**. **a** KDE of the number of replies per story; the x-axis was $\log_{10}(x+1)$-transformed. **b** Time series (KDE-approximated) of all replies ($N = 2.07 \cdot 10^5$) **c** KDE of the number of fact-checked tweets per story; the x-axis is $\log_{10}(x+1)$ -transformed. **d**, **e** KDEs of Recall **d** and Overall Precision **e**. Note that these measures are estimates. The density evaluation was restricted to [0, 1]. **f** KDE of the number of matching tweets. These data were accessed via the Tweet count endpoint, and that the number of retrieved tweets for a given story might be lower due to downsampling. **g** Time series (KDE-approximated) of all retrieved tweets ($N = 2.69 \cdot 10^7$). **h** KDE of the estimated story duration. The density evaluation was restricted, as we limited the observation period to 1 year

in any reply. This is also expressed in Fig. 3a, which reflects the log_{10}-transformed distribution of the number of replies per story. Despite this transformation, the distribution remains substantially right-skewed. We also found that stories checked by Snopes were more frequently mentioned (33.18 replies on average) than stories checked by PolitiFact (27.52 replies on average). This was also confirmed by a Mann–Whitney U test ($U = 5493496, p = 5.11 \cdot 10^{-4}$, two − tailed). We also observed that significant initial fact-checking efforts on Twitter coincided with the 2016 US presidential election, as shown in Fig. 3b. This figure also indicates that fact-checking remained prevalent in the following years, with a peak in 2020, possibly due to the emergence of COVID-19 and the 2020 US presidential election.

Story selection

To ensure the quality of the retrieved tweets, we applied a set of exclusion criteria (see Methods section). A key reason for deselecting stories was, as stated previously, that many stories played only a negligible role on Twitter, as indicated by a low prevalence of fact-checking. However, even when a query selection process was initiated, only in 56.58% of the cases the ACR method also identified a valid query. The exclusion process

(see Additional file 2 Fig. 2 for the detailed exclusion flowchart) finally collected tweets of 348 stories (out of 7091), corresponding to a dropout rate of 95.09%. This low success rate of 4.91% indicates that the ACR method is relatively strict in terms of data selection.

It is important to note that a considerable number of the stories likely did not spread on Twitter to a substantial degree, as indicated by the low median number of replies containing fact-checks (see above). We therefore also quantified the success rate of the ACR method under consideration of the effective dissemination of the stories, therefore reflecting the capability of the ACR method to retrieve tweets from stories which presumably attracted public attention. We used the number of fact-checking replies as a proxy for the public attention towards a story, assuming that a higher number corresponds to higher public attention. We found a slight improvement of the success rate ($\frac{348}{5502} = 6.325\%$) when only considering stories which were fact-checked at least once (Replies ≥ 1). However, a story having only a few fact-checking replies may be considered as present, but insignificant, as it likely played only a negligible role on Twitter. We therefore quantified the success rate for stories having at least 30 fact-checking replies, indicating that the stories probably attracted significant public attention. Overall, there were 1296 stories having at least 30 fact-checking replies, corresponding to a ACR success rate of 26.852%. This higher success rate suggests that the ACR method can fetch a considerable proportion of stories that have gained significant traction on Twitter.

Please note that the upcoming sections only present results where the ACR method was *not* terminated.

Fact-checked tweets

In total, we retrieved $4.08 \cdot 10^4$ fact-checked tweets[5]. The fact-checked tweets showed a similar pattern as the replies in terms of the distribution shape (see Fig. 3c): After $\log_{10}$ -transformation, the distributions remained right-skewed, indicating that strong fact-checking of misinforming tweets is relatively rare on Twitter. This is also confirmed by the discrepancy of the median (72.5) and the mean (117.21).

Query selection and performance metrics

We evaluated up to 50 (sub)queries per story, indicating that a substantial number of (sub)queries were tested for each story (for the distribution, see Additional file 2 Fig. 3a). The averaged number of subqueries per story is 2.08 (max. 6), but, notably, the distribution of the valid number of queries decreased exponentially (see Additional file 2 Fig. 3b), indicating that our early stopping approach successfully restricted the number of queries. Concerning the number of available baseline periods, we found that most stories had three baselines, followed by four baselines (average: 3.39 baselines). Notably, baseline b_4 (post-story) was most likely to be missing due to time constraints. The distribution of the number of available baselines is shown in Additional file 2, Fig. 3c.

[5] Note that we did not retrieve a parent fact-checked tweet for every reply for three reasons. First, one and the same fact-checked tweet may be checked by multiple replies (containing the same link to a fact-checking article). Second, the Twitter API may not return the parent tweet after the request due to, e.g., tweet deletion. Third, the fact-checked tweets may not be in English. On average, 2.03 replies yielded one fact-checked tweet.

With respect to the performance metrics, we found that Recall followed a right-skewed distribution (see Fig. 3c) with an average and a median of .259 and .228, respectively. This pattern of low average recall and positive skewness can be explained by the relatively strict precision thresholds: Queries that are both sensitive and precise might occur relatively rarely due to the precision-recall tradeoff. Thus, the observed distribution can be considered a signature of the precision-recall tradeoff. Despite the relatively low average recall, it is nevertheless reasonable to assume that we were able to collect representative sets of tweets according to the law of large numbers.

Precision (see Fig. 3d), on the other hand, also behaved as expected: Overall Precision was kept above 0.95, with an average and a median of .985 and .987, respectively. The skew of the distribution is arguably due to ceiling effects and stricter precision thresholds due to penalties for fewer baseline periods.

Retrieved tweets

Overall, we retrieved $2.69 \cdot 10^7$ tweets belonging to 348 stories. On average, we collected $7.72 \cdot 10^4$ tweets per story (median: $9.98 \cdot 10^4$, see Fig. 3g). It is, however, important to note that we had to perform tweet sampling for 235 (67.53%) stories, as the respective number of matching tweets exceeded the threshold of 10^5. The distribution of the number of matching tweets (see Fig. 3f) was highly right-skewed, making a log_{10}-transformation necessary. This pattern of (extremely) skewed distributions is very common for social data (see, e.g., [23]) and may partially reflect the 'viral' nature of social media or biases introduced by algorithms used by the social networking services [7]. However, the median of $2.59 \cdot 10^5$, as well as the interquartile range of $5.78 \cdot 10^4$ to $2.29 \cdot 10^6$, support the idea that most stories are small-scaled relative to the entirety of tweets.

With respect to the temporal features of our dataset, we found that most stories emerged beginning in 2016 (see Fig. 3h) with a peak in 2020/21. In general, the time series seems to resemble the time series of the replies, as shown in Fig. 3b. This close correspondence, although to some extent logical (as fact-checking replies are a precondition for the ACR method), restricts the interpretability of the time series shown in Fig. 3h: The conclusion that the prevalence of misinformation on Twitter increased substantially over the past years might be inappropriate, whereas the conclusion that misinformation tends be more fact-checked and countered might hold. The peak in 2020/21, however, may be attributed to the COVID-19 pandemic, which was accompanied by a sizeable body of misinformation, but also significant fact-checking efforts (see, e.g., [10]). The estimated duration of the stories (Fig. 3i) indicates that most were estimated to last for one year or longer. Note that estimated duration may not always correspond to true duration, as we used a relatively liberal estimator to avoid exclusion of time periods where the story was (still) spreading. However, the relatively large temporal extent of many stories is not implausible: Shin et al. [66] presented preliminary evidence that misinformation tended to reappear multiple times after the initial peak, potentially explaining the observed pattern.

Main indicators of the ACR method

We subsequently present a set of indicators that provide robust and direct evidence regarding the validity of the ACR method. For ease of readability, we focus here on important indicators only; additional indicators can be found in Additional file 3.

In order to objectivize the efficacy of the ACR method, we mainly rely on a text similarity measure, which we used to determine the semantic similarity between individual tweets and the claim of a story. Specifically, we used a sentence-transformers model (see [60]), a state-of-the-art Python framework for text embeddings. We used the best-performing model provided by Reimers and Gurevych [60] (all-mpnet-base-v2; see [61]), showing good model performance when fine-tuned on a set of natural language processing (NLP) tasks. Model performance was evaluated on the basis of the Sentence Embedding Benchmark, which uses a set of clustering, binary classification, retrieval, reranking, and semantic textual similarity tasks (see [60]). This model allowed us to map sentences to vectors capturing their semantic information. These vectors can then be used to compute measures of similarity. We used cosine similarity ($S_c(a, b) \in [-1, 1]$; see, e.g., [70]) to quantify semantic similarity between tweets (a) and the respective claim (b).

Comparing text similarity across baselines and story period

Under consideration of Eq. 12 and Eq. 14, we assumed that story-supporting tweets can be only retrieved during the story period, while only irrelevant tweets can occur during baseline periods. We therefore expected lower text similarities for the baselines compared to the story period. We collected up to 10k tweets per baseline, depending upon how precise the respective query was for a given baseline. We used both linear mixed-effects modelling (LMEM) and receiver operating characteristic (ROC) analyses to quantify the differences in text similarity between baselines and story period.

LMEM of mean text similarity across baselines and story period. Unlike analysis of variance, LMEM can accommodate missing data under the missing at random assumption (see, e.g., [45, 46]) and can also take into account that the performance of the ACR method may vary across stories (random effects). The ACR method may (intentionally) yield missing data for some baselines, as (i) baselines may not be available due to time constraints (see Methods section) or (ii) the identified query might be so selective that no false positives were present for some baselines.

The LMEM was performed with the STATSMODEL package (version: 0.13.2), using the Limited-memory Broyden-Fletcher-Goldfarb-Shanno algorithm or Powell's method (see [42, 56]) combined with maximum likelihood estimation. We considered mean text similarity for each time period as our dependent variable; this measure, however, was only determined when at least 100 tweets were available for the respective time period. We specified our model according to the formula `sim ~ t + (1 + t|sid)`, where `sim`, `t` and `sid` stand for mean text similarity, time point of measurement (b_1, b_2, b_3, b_4 and story period s as reference period) and story ID, respectively. In such a random effects model, both the intercepts and slopes are allowed to vary stochastically. We also tested a fixed effects model (formula: `sim ~ t + (1|sid)`) in contrast (restricting the slopes to be invariant), but found that the random effects model showed better model fit, as indicated by both Akaike information criterion (AIC) and BIC (Bayesian

Table 2 Fixed effects of the random effects model for mean text similarity

Time period	Coefficient	SE	*z*	$p > \|z\|$	95% confidence interval	
					0.025	0.975
Intercept s	0.47	$7.11 \cdot 10^{-3}$	66.13	≈ 0	0.456	0.484
Baseline b_1	−0.099	$7.42 \cdot 10^{-3}$	−13.33	≈ 0	−0.113	−0.084
Baseline b_2	−0.115	$6.93 \cdot 10^{-3}$	−16.54	≈ 0	−0.128	−0.101
Baseline b_3	−0.099	$6.54 \cdot 10^{-3}$	−15.06	≈ 0	−0.111	−0.086
Baseline b_4	−0.088	$7.64 \cdot 10^{-3}$	−11.49	≈ 0	−0.103	−0.073

Note: *SE* stands for standard error and reported P -values reflect one-tail P-values

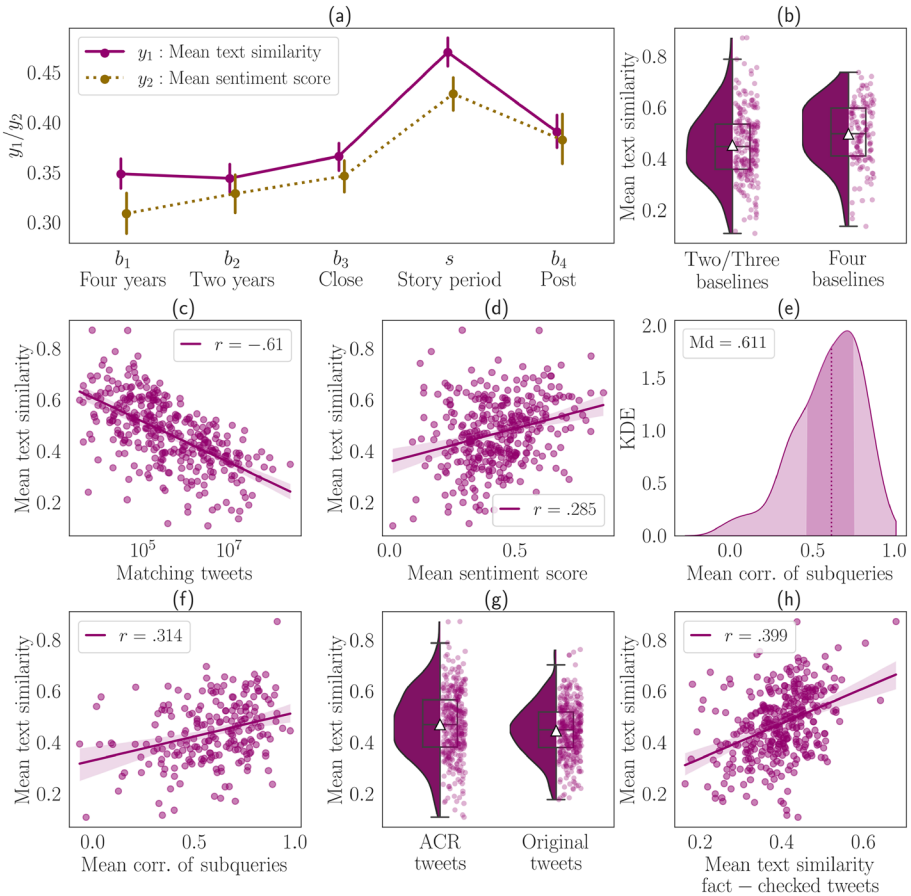

Fig. 4 Main indicators of the ACR method. Regression lines reflect ordinary least squares regression, with 95% confidence intervals (estimation by bootstrapping). In panels **b** and **g**, colored areas and white triangles reflect KDEs and means, respectively. **a** Mean text similarity and sentiment score across baselines and story period. Error bars reflect 95% confidence intervals. Both measures showed the expected pattern of a peak during the story period. **b** Mean text similarity was significantly higher for stories with four baselines, indicating that the ACR method is more reliable when more baselines are available. **c** A strong association between the number of matching tweets and mean text similarity is a signature of occasional tweet overload. **d** A weak association between mean text similarity and sentiment score indicates that the more reliably a story is measured, the more negative the tweets are. **e** KDE of the mean correlation between subqueries. A majority of the subqueries are highly correlated. **f** Association between mean correlation between subqueries and text similarity; the more the subqueries converge, the better the story is measured. **g** A text similarity comparison of the ACR-retrieved and original tweets indicates that ACR-retrieved may outperform original tweets. **h** An association between text similarity of the ACR-retrieved and fact-checked tweets indicates that the reliability of the ACR depends upon a good training set

information criterion; see, e.g., [12, 72]). For full model outputs, absolute and subtracted AIC and BIC values, see Additional file 4 (output 1 and 2).

For improved readability, we report only the fixed effects of each baseline (see Table 2). Mean text similarity was significantly reduced for all baselines compared to the story period, as indicated by negative baselines coefficients (Fig. 4a), and an expected pattern of mean text similarity across baselines: While the pre-story baselines b_1, b_2 and b_3 showed the lowest text similarity, text similarity of post-story baseline b_4 was slightly less reduced. This b_4-effect attenuation can be explained by the potential presence of residual story-supporting tweets, as a story may continue to reemerge on Twitter even after long periods of time [66]. We observed substantial effect heterogeneity across stories (see Additional file 5), with a minority of stories (40.52%) not showing the expected pattern of a similarity peak during the story period at all, arguably reflecting failures of the ACR method. Welch's t-tests confirmed for 207 (59.48%) stories that the similarity during the story period was significantly higher than all other baselines.

Text similarity during story periods was significantly higher for stories having four baselines (average: .497) compared to stories with two or three baselines (average: .454), as indicated by Fig. 4b and confirmed by Welch's t-test ($t(307.48) = 3.05, p = 1.22 \cdot 10^{-3}, \text{one} - \text{tailed}$). We expected this, as more baselines should increase the precision of the respective query, thereby increasing the text similarity of the tweets with the respective claim. A control analysis (see Additional file 6 confirmed that this effect was *not* driven by potential confounding variables (such as recall and number of subqueries).

ROC analysis of text similarity between baselines and story period. We performed a ROC analysis (see, e.g., [22]) by considering tweets during story period s as positive tweets, while tweets during baselines were considered to be negative.[6] ROC analysis was performed using the SCIKIT-LEARN package (version: 1.1.1; see [53]) using the METEICS.ROC_CURVE function; we merged tweets from all available baselines and determined ROC curves, as well as areas under the ROC curves (AUCs; for interpretation rules, see, e.g., [28]), if at least 100 tweets were available for the baselines.

The ROC analysis confirmed the substantial performance heterogeneity of the ACR method: While a mean AUC of .66 (SD = .175) indicates fair but not excellent classification performance, we observed good ($\geq$ .7), very good ($\geq$ .8) and even excellent ($\geq$ .9) performance for 133 (40.67%), 73 (22.32%) and 37 (11.31%) stories, respectively. At the same time, we observed ACR failures for 133 (40.67%) stories, as indicated by AUCs being equal to or smaller than .6, corresponding to poor or even uninformative classification. However, ROC analysis confirmed that (complete) ACR failures are rare. For individual ROC curves and AUCs, see Additional file 7.

We also observed a relatively strong association between AUC and mean text similarity during story periods ($r(325) = .577, p \approx 0, \text{two} - \text{tailed}$), indicating that better discrimination between tweets of the story and baseline periods is accompanied by an

[6] Strictly speaking, the tweets during story period should also contain a small proportion of negative tweets (as implicated by the defined precision thresholds, see Methods section). As the average precision of our queries was relatively close to 1 (see Fig. 3d), the presence of negative tweets during the story period can be considered negligible.

absolute increase in mean text similarity. This association can be interpreted as evidence for the discriminative power of the ACR method to separate story-supporting tweets from irrelevant tweets.

Again, we observed that more baselines improve ACR performance, as indicated by higher AUC scores for four stories (mean: .689) with four baselines relative to stories with two or three baselines (mean: .641). A Welch's t-test confirmed this difference ($t(297.67) = 2.51, p = 6.28 \cdot 10^{-3}$, one − tailed), which is in line with our expectations, as more baselines should improve the discriminative power of the ACR performance.

Validating the ACR method through manual annotation of tweets

To gain a deeper understanding of the ability of the ACR method to collect story-related tweets, we conducted a validation analysis where we manually annotated a sample of tweets (for a full description of the procedure, the results and the conclusions, see Additional file 9). We used 37 stories with excellent automated classification levels (AUC $\geq$ .9), expecting to observe similarly unambiguous classification by human raters. To obtain robust estimates of task performance for each of these stories, we randomly sampled 30 tweets per story, which corresponds to a total number of annotated tweets $N = 37 \cdot 30 = 1110$. The tweets were annotated by at least two independent raters.

The raters performed two tasks: They determined whether a tweet was related to the story or not (Task I: Relatedness) and if so, whether it was supportive, neutral, or contradictive for the story (Task II: natural language inference [NLI]). We found that the inter-rater reliability, as measured by Cohen's κ, was acceptable for both Task I ($\kappa = .637$) and Task II ($\kappa = .537$). We observed that 984 of the 1110 annotated tweets were related to the stories, corresponding to an overall performance of 88.65% (95% [86.63%, 90.46%]). The performance varied from story to story, ranging from 50.0% to 100.0% success rate (median: 93.33%). Most stories (22 $\widehat{=}$ 59.46%) showed performances $\geq$ .9, and 10 stories (27.03%) even showed perfect performance, meaning that all annotated tweets in the sample were considered to be related. The observed overall performance approximates closely the intended precision of .9, indicating that the ACR method is capable of detecting story-related tweets.

Among the related tweets, we found that 76.3% (95% CI [73.51%, 78.92%]) of the tweets were also supportive, meaning that a majority of the tweets in fact support the stories. On the level of the individual stories, we observed that most of the stories showed high proportions of supportive tweets, as indicated by a (i) median proportion of 86.21% and (ii) low number of stories having less than 50% supportive tweets ($N = 6$ [16.22%]). We also observed a considerable amount of tweets being contradictive to the stories (16.58%; 95% CI [14.31%, 19.06%]), whereas neutral tweets were less prevalent (7.12%; 95% CI [5.59%, 8.91%]). We found that a majority of the stories had less than 10% contradicting tweets ($N = 24$ [64.86%]), which was paralleled by a relatively low median proportion of contradicting tweets (6.67%). These results indicate that the ACR primarily gathers tweets that support the stories, yet it intermittently retrieves tweets that exhibit neutrality or contradict the stories.

Table 3 Fixed effects of the random effects model for mean sentiment score

Time period	Coefficient	SE	*z*	$p > \|z\|$	95% confidence interval	
					0.025	0.975
Intercept s	0.429	$7.84 \cdot 10^{-3}$	54.69	≈ 0	0.413	0.444
Baseline b_1	−0.111	$9.45 \cdot 10^{-3}$	−11.73	≈ 0	−0.129	−0.092
Baseline b_2	−0.096	$9.49 \cdot 10^{-3}$	−10.13	≈ 0	−0.115	−0.078
Baseline b_3	−0.083	$8.41 \cdot 10^{-3}$	−9.87	≈ 0	−0.099	−0.067
Baseline b_4	−0.042	0.011	−3.76	$8.57 \cdot 10^{-5}$	−0.064	−0.02

SE stands for standard error and reported P-values reflect one-tail P-values

Tweet overload phenomenon

We observed large numbers of matching tweets (119 (34.2%) stories exceeded 10^6), which could be related to at least two different factors. First, stories may actually disseminate broadly, meaning that there are many true positives among the tweets. Second, the ACR method may fail to identify a sufficiently specific query (leading to tweet overload with false positives). A substantial negative correlation between the number of matching tweets and text similarity ($r(346) = -.61, p \approx 0, \text{two} - \text{tailed}$; see Fig. 4c) indicates that the second explanation may be more likely: stories with many matching tweets showed poor correspondence with the respective claims, indicating an overload with false positives. Conversely, stories with few matching tweets showed high similarity with the respective claims. This finding was confirmed by a robust association between the number of matching tweets and AUC ($r(325) = -.394, p \approx 0, \text{two} - \text{tailed}$), indicating that tweet overload also led to reduced classification performance.

Comparing sentiment scores across baselines and story period

Sentiment analysis is another NLP tool to assess whether false stories tend to elicit negative affective responses. A growing body of research indicates that misinformation (especially when disseminated with the intent to deceive) tends to use negative emotional language to fuel uncertainty and conflict (see, e.g., [16, 47, 67]). We used the sentiment analysis provided in the TWEETNLP library (see [13]), which is a dedicated state-of-the-art language model, to differentiate three affective categories (negative, neutral, positive) of tweets. The model provides both tweet categories and category probabilities. We used the mean probability that a tweet was classified as negative $p(\text{catagory} = \text{negative})$ as our dependent measure, which we denoted 'mean sentiment score'.

We used the same mixed-effects model specification used for the text similarity to quantify mean sentiment scores during the different time periods (see Table 3). We observed a similar pattern as for the text similarity: While mean sentiment scores were reduced during baselines, they peaked during story periods (see also Fig. 4a). Even though the effects were slightly smaller than for mean text similarity, they were highly significant for all baselines, indicating that false stories indeed contained more negative affective information. Welch's t-tests confirmed that in 41.38% of the stories, the sentiment score during the story period was significantly higher than all other baselines. It is conceivable that these effects should be interpreted in the context of a general tendency

of the public discourse to become more 'hostile' (see, e.g., [39]) over the past years. Note however that sentiment reduction during baseline b_4 (post-story) shows that elevated sentiment scores during story periods are specific, and do not simply result from a general tendency over time.

We also found a weak to moderate association between mean text similarity and mean sentiment score ($r(346) = .285, p = 3.22 \cdot 10^{-8}$, one − tailed; see Fig. 4d), indicating that the more validly a story is measured, the stronger the inherent negative affective information. Again, this correlation might be confounded by the inherent 'negativity' of the stories' claims; if so, it would not truly reflect an increase in affective negativity. We excluded this possibility by a semipartial correlation, using the sentiment score of stories' claims as a covariate, which did not confirm a confounding effect as the correlation remained virtually unchanged ($r(345) = .274, p = 1.06 \cdot 10^{-7}$, one − tailed). We consider this association as evidence that the ACR method indeed collects tweets belonging to the respective story.

Time series correlation between subqueries

For a majority of the stories (61.78%), multiple subqueries were identified. As each subquery should detect tweets that capture the same story, the time series of the subqueries should be positively correlated. We retrieved the time series for each subquery via the Tweet counts endpoint (granularity: days; see Methods section). We performed $log_{10}(x + 1)$-transformations to the time series, as recommended by Gonzales [23]. For any story having three or more subqueries, we calculated the mean of all correlation coefficients.

On average, the time series of the subqueries were robustly correlated ($\bar{r} = 0.586, \text{SD} = 0.205$; see Fig. 4e), suggesting that these time series reflect the same underlying process. However, this correlation between time series of subqueries could be spurious if different subqueries match the very same tweets, thus artificially raising the correlation coefficients. To address this, we conducted a control analysis where we considered the correlation among the subqueries where these intersections were eliminated (relative complements). Despite a relatively small decrease, the analysis confirmed the robustness of the average correlation ($\bar{r} = 0.504, \text{SD} = 0.2$).

The substantial average correlation between subqueries' time series allows us to infer that the ACR method can reliably measure an underlying process. Whether this process also reflects the process of interest (i.e., the respective story), remains an open question. We therefore considered the correlation between text similarity and subqueries' time series correlation. We expected a positive correlation, as we assumed that when the underlying process is measured more reliably, the validity (as indicated by text similarity) should increase as well. We observed a moderate correlation between both variables ($r(213) = 0.314, p = 1.27 \cdot 10^{-6}$, one − tailed; see Fig. 4f), indicating that queries that are assumed to be more reliable also reflect the respective story more accurately.

Comparing fact-checked and ACR tweets

To assess whether the ACR method yields tweets that represent the respective story to a comparable degree as the original tweets do, we compared the mean text similarity between ACR tweets and original tweets. Here, we defined original tweets following

Vosoughi et al., meaning that only first-level tweets were considered (replies to replies were not considered). In fact, we found that the ACR tweets showed higher mean text similarity (average: 0.47) relative to the original tweets (average: 0.446), as confirmed by a paired t-test ($t(347) = -3.5, p = 5.2 \cdot 10^{-4}$, two $-$ tailed; see also Fig. 4g).

We additionally conducted a ROC analysis of text similarity for the original tweets, in order to better compare the performance of both methods. So far, we have only conducted a ROC analysis for the ROC method (see Sect.), where we considered tweets during the story period as positive tweets, while tweets during the baselines were considered to be negative. We performed the same analysis for the approach of Vosoughi et al., whereas the retrieved original tweets were considered as positive, while the negative tweets still corresponded to the baseline tweets.

We found that original tweets also showed a fair, but slightly reduced performance for the given task: While the ACR tweets achieved a mean AUC of 0.66 (SD: 0.175), the original tweets had a mean AUC of 0.616 (SD: 0.193). This difference was confirmed by a paired t-test ($t(325) = 3.79, p = 1.77 \cdot 10^{-4}$, two $-$ tailed). In accordance with the ROC results of the ACR tweets, we found the performance to be heterogeneously distributed for the approach of Vosoughi et al. too. We observed good (≥ 0.7), very good (≥ 0.8) and excellent (≥ 0.9) performance for 118 (36.2%), 62 (19.02%) and 20 (6.13%) stories, respectively. At the same time, we observed poor classification performance (AUC < 0.6) for 145 (44.48%) stories.

Even though such difference should be interpreted with caution (and may not necessarily mean superiority of the ACR method), it further provides evidence that the ACR method is an efficient tool to identify specific linguistic features, and one that may even outperform the approach proposed by Vosoughi et al. in certain domains.

Interestingly, we observed a different pattern for the mean sentiment score: original tweets were more emotional compared to the ACR tweets, as shown by a paired t-test ($t(347) = 7.18, p \approx 0$, two $-$ tailed). We suspect that the increased emotional valence triggered greater outreach (via retweets), thereby increasing the probability that tweets were fact-checked. If this is true, then the ACR method may additionally detect a 'blind spot' of less emotional, less distributed tweets. We performed a control analysis to prove whether fact-checked tweets had in fact higher retweet frequencies than ACR tweets: for each story, we compared the retweet frequencies of both methods via a Mann–Whitney U test (two-tailed). In 304 (87.36%) of the stories, the retweet frequency was significantly higher for the fact-checked tweets than for the ACR tweets. This difference is also expressed in the much higher proportion of tweets exceeding particular retweet frequencies: 69.65% of the fact-checked tweets exceeded the threshold of 10 retweets, whereas only 27.16% of the ACR tweets achieved the same level of engagement. This difference remained substantial also for higher levels of retweet activity (100 retweets: 51.99% vs. 12.12%; 1000 retweets: 28.04% vs. 2.78%). We interpret this higher retweet frequency among fact-checked tweets as evidence that the ACR method is more suitable to detect tweets 'below radar level' (meaning that it also detects tweets which are not heavily distributed), while the original fact-checking approach seems to perform well for detecting large-scale tweet cascades having substantial outreach via many retweets.

We also investigated whether both methods are capable of tracking narrative dynamics of stories: The (dominant) narrative of a story may change over time, a phenomenon which has been described for (false) rumors in previous studies (see, e.g., [6, 66]). For each story and method, we partitioned the dataset into three equally sized segments: the first, second, and third segments were assumed to encapsulate the initial, intermediate, and final narratives of each story, respectively. We performed a 2 x 3 repeated-measures analysis of variance (rmANOVA) with factors data collection method (ACR versus fact-checked tweets) and time segment (initial, intermediate and final). In line with our previous findings, we observed a main effect of the method ($F(1, 321) = 15.02, p = 1.29 \cdot 10^{-4}, \eta^2 = 0.011$), indicating that ACR tweets had higher text similarity with the respective claims than fact-checked tweets (see also Fig. 4g). We also observed a main effect of the time segment ($F(2, 642) = 61.64, p \approx 0, \eta^2 = 0.019$) meaning that text similarity evolved over time negatively (thus, initial tweets corresponded to the story more closely than later ones). We also observed a significant interaction between both factors ($F(2, 642) = 7.47, p = 6.2 \cdot 10^{-4}, \eta^2 = 2.23 \cdot 10^{-3}$), indicating that the effect of the time segments depended on the data collection method. Post-hoc tests revealed that the differences between the data collection methods were more pronounced for later time segments than for the initial one, while there was no such difference between the intermediate and final segment. We suspect that the ACR method is profound in collecting one (dominant) 'line' of story, which can be attributed to the static nature of queries which are based on linguistic features.

We also found that the 'quality' of the fact-checked tweets was predictive of the ability of the ACR method to detect story-related tweets, as indicated by a moderate association between mean text similarity of fact-checked and ACR tweets ($r(346) = 0.399, p \approx 0, \text{two} - \text{tailed}$; see also Fig. 4h). This correlation underpins the dependency of the ACR method on a 'good' training dataset for feature identification.

Discussion

The proposed ACR method represents an efficient tool for large-scale and fully automatized collection of misinforming tweets. Its efficacy is based upon three pillars: (i) the adoption of prior, pioneering research in the field, (ii) a well-formalized mathematical framework and (iii) an extensive empirical proof. More precisely, it builds upon literature by extending the approach proposed by Vosoughi et al., thereby benefiting from prior research in the field. The ACR method also provides a formal mathematical framework with well-defined assumptions and implications, which distinguishes it from other approaches in terms of clarity and structure. In addition, a comprehensive set of indicators has established substantial evidence that the ACR method can, despite significant performance heterogeneity, reliably collect misinforming tweets.

Despite its strengths, the ACR method also has limitations. We repeatedly observed that the number of false positives was underestimated by the ACR method, which can probably be linked to violations of assumption iii (temporal homogeneity of irrelevant tweets). The consequence of such biased estimations was tweet overload with false positives, resulting in poor data quality and an unfavorable signal-to-noise ratio. To mitigate this issue, future revisions of the ACR method may (i) refine parameter values (e.g., improve estimates of the onset and end of a story), (ii) implement overload thresholds that, when exceeded, lead

to rejection of a query or (iii) avoid the use of multiple subqueries but instead use only the most accurate query according to specific performance measures (e.g., $F_{0.5}$; see, e.g., [29]).

When comparing the ACR method to the approach proposed by Vosoughi et al., we found a strong difference in the sensitivity to highly influential ('viral') tweets: While fact-checked tweets showed high retweet frequencies, the tweets yielded by the ACR method were much less retweeted, therefore being 'nonviral'. Previous research (see [43]) has shown that the frequency distribution of retweets can be described via a power law distribution. This implies that nonviral tweets (e.g., tweets having less than 10 retweets) constitute a substantial proportion of the tweets, while viral tweets with high outreach (e.g., > 1000 retweets), on the other hand, are relatively rare. Our results show that the ACR method is more sensitive to nonviral tweets, while the original fact-checking approach predominantly catches viral ones. Retrieving data from multiple inputs (whereas the methods may have different strengths and limitations) is crucial for drawing robust inferences, as such multi-method approaches allow validating research results. We therefore believe that researchers in the field of social media research who are interested in diversifying their data inputs could benefit from this approach. Apart from its relevance for users among the scientific community, ACR might also contribute to efficient automatic pre-screening of social data by fact-checking organizations, thus ultimately promoting faster responses.

We also consider the keyword-based nature of the ACR method as a limitation, for two reasons. First, the narrative of a misinformation story may change over time (see [66]; but see also [6]), making dynamic tracking of such changes necessary. The ACR method is assumed to be less adaptive to such narrative changes compared to the original fact-checking approach. It is reasonable to assume that this reduced adaptivity of the ACR method can be linked to the static nature of (keyword-based) queries used to collect the tweets: When a narrative changes over time (e.g., by exchanging a specific term), the ACR method may fail to detect tweets belonging to this second narrative, as the query is based on the initial term. This might be particularly important for stories resurfacing multiple times over a long time period, as it has been shown for climate change misinformation [79].

Second, a substantial proportion of misinformation in social media is distributed via videos, photos, or audio files (see, e.g., [75]). These media increasingly use "deepfakes" that contain forged identities created via artificial intelligence (AI) to enhance perceivers' trust in misinformation [86]. The APIs of social networking services typically do not have dedicated operators to search for specific videos, photos, or audio (which would be challenging, as the same photo/video/audio may exist with different formats and filters). Further research is needed to address this shift from text-based misinformation towards photo-, video-, or audio-based material. Recent developments in AI (such as ChatGPT or DALL-E; see [51, 52]) may facilitate our everyday life and work substantially, but may *also* facilitate the creation of false media content. We appeal to the research community to address this issue, as the presence of these new technologies may dramatically affect the volume and format of online misinformation. The proposed ACR method could constitute an important tool in addressing these new forms of misinformation: While it was not designed to recognize AI-generated content, it arguably represents an important

'preselector' of potentially relevant material by identifying specific keywords that might be attached to the respective AI-generated media.

We explicitly suggest that the ACR method *should* be combined with other methods for misinformation detection to improve data quality. An initial approach could be to use a text similarity threshold, while it is assumed that text similarity has a (positive) probabilistic relationship to the (true) labels of tweets (thus, the higher the text similarity, the higher is the chance that a tweet indeed relates to the respective story). However, there are other challenges regarding the valid classification of tweets. For instance, the query OBAMA INJURED aiming to detect a misinformation story that Obama was injured during an attack on the White House (see Additional file 1) potentially also yields results that contradict the claim (e.g., 'Obama was *not* injured!') or which use irony to debunk it (e.g., 'Obama was injured? then I will move into the White House!'). Our manual annotation results confirmed that the ACR method successfully retrieves a large majority of story-supporting tweets. At the same time, these results suggest that, in cases in which optimal performance at the level of specific stories is a priority, it can be beneficial to combine automated ACR-based processing with manual checks of tweets to achieve the best outcome. Large language models may help to better distinguish posts that truly support the story of interest from the other posts by using strict, ROC-based thresholds, NLI or stance or irony detection (see, e.g., [13, 48]). The performance of such large language models on social data may currently be limited, as posts on social media typically 'suffer' from a fragmented syntactic structure as well as the use of unusual, online-specific words (e.g., lol or fyi; see, e.g., [35, 50]). The correspondence of the training data of these models to social data is often poor, highlighting the need to incorporate appropriate training data before applying such models to social media.

At this point, the ACR method is a tool to collect misinformation on Twitter. The degree to which it can be applied to other social networking services remains to be tested. We expect that the ACR method will also prove a valid tool for other services, but its efficacy may vary due to differences in architectures and user populations, thus calling for adaptations to achieve optimal performance. In general, the outcome of the ACR method depend on a complex interplay of the parameters (and optimal parameter settings may vary from story to story), explaining its heterogeneously distributed performance. This includes, among others, the dependency on the fact-checked tweets. Besides a threshold of the minimum number of fact-checked tweets of $n = 30$ (which makes ACR less sensitive to relatively small stories), we observed that the accuracy of the ACR tweets was contingent on the quality of the entered fact-checked tweets. Diversifying the data inputs and further fine-tuning of other parameters will likely improve ACR's performance significantly. For instance, additional baselines are presumed to be beneficial for the outcome of the ACR method. Moreover, more precise estimations of the onset and end of a story may also be advantageous, due to higher temporal specificity (see section). We hope that future research will promote better understanding of such parameter effects and thus optimization of ACR.

Unfortunately, the research community is witnessing significant aspirations of social networking services to monetize their data: Twitter announced in February 2023 that it will increase the cost of its API dramatically (see, e.g., [84]), making it economically unfeasible for researchers to further collect data via the Twitter API. Similarly, Reddit

announced in April 2023 that it will no longer provide an API free of charge (see [59]), triggering disapproval of the Reddit community. These profit-seeking steps of social networking services drastically constrain the availability of social data, thereby rendering future research in the field highly challenging, but not impossible (see, e.g., [26, 49]).

While these adverse developments promote fundamental questions regarding the ownership of data that have been provided by large user communities, it seems clear that future research in the field is important and potentially vital for societies: Being equipped with knowledge about misinformation and tools to detect and counter it may be an important precondition for the integrity of societies and democracies (see, e.g., [33, 39, 58, 74]; but see also [83]). In this context, we believe it is essential that social networking services provide free or low-cost APIs to enable researchers to generate robust evidence regarding pressing research questions.

We consider the ACR method proposed here as a valuable tool extending the present research equipment. Despite its theoretical and empirical foundation, it comes with a set of advantages: It (i) is fully automated, (ii) yields large-scale datasets, (iii) allows estimates of how reliable the data are and (iv) probably also works for mixed and true stories, i.e., stories which were fact-checked and rated as mixed/true (see Additional file 8 view for an evaluation of the ACR method for true stories). In sum, the ACR method represents a useful extension to the methods pool of the field, as it can be used for automated large-scale collection of misinformation on Twitter, with promising potential for other social networking services and different kinds of social data.

Conclusion

The ACR method proposed here represents a sophisticated technique for automatized, large-scale collection of misinforming tweets for three reasons. First, it relies on the current literature by adopting and extending an elaborated approach proposed by Vosoughi et al. Second, and unlike many other methods, it provides a clear and formal mathematical framework with well-defined assumptions and implications. Third, its validity has been linked to a comprehensive set of indicators (including manual checks of tweets), which showed that the ACR method is, despite significant performance heterogeneity, a useful extension to the methods pool of the field, as it allows to better collect nonviral tweets.

Abbreviations

ACR	Adaptive community-response
AI	Artificial intelligence
AIC	Akaike information criterion
API	Application programming interface
AUC	Area under (ROC) curve
BIC	Bayesian information criterion
KDE	Kernel density estimation
LMEM	Linear mixed-effects modelling
NLI	Natural language inference
NLP	Natural language processing
rmANOVA	Repeated-measures analysis of variance.
ROC	Receiver operating characteristic

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s40537-024-00894-w.

Additional file 1. Model of the low prevalence problem on Twitter.

Additional file 2. Additional figures concerning the ACR method.

Additional file 3. Additional indicators validating the ACR method.

Additional file 4. Outputs of the LMEM.

Additional file 5. Text similarity on story level over story period and baselines.

Additional file 6. Control analysis conforming that text similarity is reduced for four baselines.

Additional file 7. ROC curves and AUC on story level.

Additional file 8. Evaluation of the ACR method for true stories.

Additional file 9. Validating the ACR method through manual annotation of tweets.

Acknowledgements
We thank Linda Ficco for her suggestion regarding the figures. We thank Clara Marie Breier for her suggestions regarding the manuscript. We thank Maria Zangemeister and Bernadette Knoepffler for manually annotating the validation sample of tweets.

Author contributions
JK designed the method, created the software, collected and interpreted the data and drafted the work HK designed the method, interpreted the data and substantively revised the work SRS designed the method, interpreted the data and substantively revised the work AS interpreted the data and substantively revised the work

Funding
Open Access funding enabled and organized by Projekt DEAL. JK is funded by a Doctorate scholarship (Landesgraduiertenstipendium; funded by the free State of Thuringia, Germany).

Availability of data and materials
All the data that do not conflict with Twitter's terms of service (see https://developer.twitter.com/en/developer-terms/policy) are available on OSF (https://osf.io/6waby/).

Code availability
All the code underlying this research paper is available on OSF (https://osf.io/6waby/).

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Received: 29 August 2023 Accepted: 11 February 2024
Published online: 24 February 2024

References

1. Afroz S, Brennan M. Greenstadt R Detecting hoaxes, frauds, and deception in writing style online. Proc IEEE Sympos Secur Privacy. 2012. https://doi.org/10.1109/SP.2012.34.
2. Akram W. A study on positive and negative effects of social media on society. Int J Computer Sci Eng. 2017. https://doi.org/10.26438/ijcse/v5i10.351354.
3. Allcott H, Gentzkow M. Social media and fake news in the 2016 election. J Econom Perspect. 2017;31(2):211–36. https://doi.org/10.1257/JEP.31.2.211.
4. Allcott H, Gentzkow M, Yu C. Trends in the diffusion of misinformation on social media. Res Polit. 2019. https://doi.org/10.1177/2053168019848554.
5. Allcott H, Braghieri L, Eichmeyer S, et al. The welfare effects of social media†. Am Econ Rev. 2020. https://doi.org/10.1257/aer.20190658.
6. Allport GW, Postman L. An analysis of rumor. Publ Opin Quart. 1946. https://doi.org/10.1093/poq/10.4.501.
7. Baym NK. Data not seen: the uses and shortcomings of social media metrics. First Monday. 2013. https://doi.org/10.5210/fm.v18i10.4873.
8. Bird S, Loper E, Klein E. Natural language processing with python. O'Reilly Media Inc. 2009
9. Bovet A, Makse HA, Influence of fake news in Twitter during the,. US presidential election. Nature Commun. 2016;2019. https://doi.org/10.1038/s41467-018-07761-2.
10. Brennen JS, Simon FM, Howard PN, et al. Types, sources, and claims of COVID-19 misinformation Tech Rep 2020. Oxford: University of Oxford; 2020.
11. Buckland M, Gey F. The relationship between recall and precision. J Am Soc Inform Sc. 1994. https://doi.org/10.1002/(SICI)1097-4571(199401)45:1$<$12::AID-ASI2$>$3.0.CO;2-L.

12. Burnham KP, Anderson DR. Understanding AIC and BIC in model selection. Multimodel Inference. 2004. https://doi.org/10.1177/0049124104268644.
13. Camacho-collados J, Rezaee K, Riahi T, et al. TweetNLP: Cutting-Edge Natural Language Processing for Social Media. In: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. Association for Computational Linguistics, Abu Dhabi, UAE, pp 38–49, 2022 arXiv:2206.14774
14. Camargo CQ, Simon FM. Mis- and disinformation studies are too big to fail: six suggestions for the field's future. Harvard Kennedy School Misinform Rev. 2022. https://doi.org/10.7016/mr-2020-106.
15. Chen IH, Strong C, Lin YC, et al. Time invariance of three ultra-brief internet-related instruments: smartphone application-based addiction scale (SABAS), Bergen social media addiction scale (BSMAS), and the nine-item internet gaming disorder scale- short Form (IGDS-SF9) (Study Part B). Addictive Behav. 2020. https://doi.org/10.1016/j.addbeh.2019.04.018.
16. Chuai Y, Zhao J. Anger can make fake news viral online. Front Phys. 2022. https://doi.org/10.3389/fphy.2022.970174.
17. Cinelli M, de Francisci Morales G, Galeazzi A, et al. The echo chamber effect on social media. Proc Natl Acad Sci United States Am. 2021. https://doi.org/10.1073/pnas.2023301118.
18. Davis CA, Varol O, Ferrara E, et al. BotOrNot. In: Proceedings of the 25th international conference companion on world wide web, 273–274, 2016 https://doi.org/10.1145/2872518.2889302
19. Duke Reporters' Lab Fact- Checking - Duke Reporters' Lab. 2021 https://reporterslab.org/fact-checking/, Accessed 29 Aug2023
20. Duke Reporters' Lab The Facts About ClaimReview - The ClaimReview Project. 2023 https://www.claimreviewproject.com/the-facts-about-claimreview, Accessed 29 Aug2023
21. ewz93 machine learning - Should I remove stopwords before generating n-grams? - Cross Validated. 2022 https://stats.stackexchange.com/questions/570698/should-i-remove-stopwords-before-generating-n-grams, Accessed 29 Aug 2023
22. Fawcett T. An introduction to ROC analysis. Pattern Recogn Lett. 2006. https://doi.org/10.1016/j.patrec.2005.10.010.
23. Gonzales A. Evaluating time series and identifying trends. 2018 https://twitterdev.github.io/do_more_with_twitter_data/timeseries.html, Accessed 29 Aug 2023
24. Google LLC. Google Fact Check Tool APIs. 2023 https://toolbox.google.com/factcheck/apis, Accessed 29 Aug 2023
25. Guess A, Nagler J, Tucker J. Less than you think: prevalence and predictors of fake news dissemination on Facebook. Sci Adv. 2019. https://doi.org/10.1126/sciadv.aau4586.
26. Guess AM, Malhotra N, Pan J, et al. Reshares on social media amplify political news but do not detectably affect beliefs or opinions. Science. 2023;381(6656):404–8. https://doi.org/10.1126/science.add8424.
27. Hemsley J, Jacobson J, Gruzd A, et al. Social media for social good or evil: an introduction. Soc Media Soc. 2018. https://doi.org/10.1177/2056305118786719.
28. de Hond AA, Steyerberg EW, van Calster B. Interpreting area under the receiver operating characteristic curve. Lancet Digit Health. 2022. https://doi.org/10.1016/S2589-7500(22)00188-1.
29. Hripcsak G, Rothschild AS. Agreement, the F-measure, and reliability in information retrieval. J Am Med Inform Assoc. 2005. https://doi.org/10.1197/jamia.M1733.
30. Jack C. Lexicon of Lies: Terms for Problematic Information. Data & Society Research Institute 2017 https://datasociety.net/pubs/oh/DataAndSociety_LexiconofLies.pdf
31. Jin F, Dougherty E, Saraf P, et al. Epidemiological modeling of news and rumors on Twitter. In: Proceedings of the 7th Workshop on Social Network Mining and Analysis, SNA-KDD 2013, 2013 https://doi.org/10.1145/2501025.2501027
32. Jin Z, Cao J, Zhang Y, et al. News verification by exploiting conflicting social viewpoints in microblogs. In: 30th AAAI Conference on Artificial Intelligence, AAAI 2016, 2016 https://doi.org/10.1609/aaai.v30i1.10382
33. Karic T, Mededovic J. Covid-19 conspiracy beliefs and containment-related behaviour: the role of political trust. Personality Individual Differ. 2021. https://doi.org/10.1016/j.paid.2021.110697.
34. Kauk J, Kreysa H, Schweinberger SR. Understanding and countering the spread of conspiracy theories in social networks: evidence from epidemiological models of Twitter data. PLOS ONE. 2021;16(8): e0256179. https://doi.org/10.1371/JOURNAL.PONE.0256179.
35. Kim AE, Hansen HM, Murphy J, et al. Methodological considerations in analyzing twitter data. J Natl Cancer Institute - Monographs. 2013. https://doi.org/10.1093/jncimonographs/lgt026.
36. Kowalski RM, Giumetti GW, Schroeder AN, et al. Bullying in the digital age: a critical review and meta-analysis of cyberbullying research among youth. Psychol Bull. 2014. https://doi.org/10.1037/a0035618.
37. Lazer DMJ, Baum MA, Benkler Y, et al. The science of fake news. Science. 2018. https://doi.org/10.1126/science.aao2998.
38. Leetaru K. Visualizing Seven Years Of Twitter's Evolution: 2012-2018. 2019 https://www.forbes.com/sites/kalevleetaru/2019/03/04/visualizing-seven-years-of-twitters-evolution-2012-2018/?sh=274356017ccf, Accessed 29 Aug 2023
39. Lewandowsky S, Ecker UK, Cook J. Beyond misinformation: uderstanding and coping the "Post-Truth" Era. J Appl Res Memory Cogn. 2017. https://doi.org/10.1016/j.jarmac.2017.07.008.
40. van der Linden S. Misinformation: susceptibility, spread, and interventions to immunize the public. 2022 https://doi.org/10.1038/s41591-022-01713-6
41. van der Linden S, Leiserowitz A, Rosenthal S, et al. Inoculating the Publ Misinform limate change. Global Challenges. 2017. https://doi.org/10.1002/gch2.201600008.
42. Liu DC, Nocedal J. On the limited memory BFGS method for large scale optimization. Mathematical Program. 1989. https://doi.org/10.1007/BF01589116.
43. Lu Y, Zhang P, Cao Y, et al. On the frequency distribution of retweets. Proc Computer Sci. 2014. https://doi.org/10.1016/j.procs.2014.05.323.
44. Maertens R, Anseel F, van der Linden S. Combatting climate change misinformation: evidence for longevity inoculation and consensus messaging effects. J Environ Psychol. 2020. https://doi.org/10.1016/j.jenvp.2020.101455.

45. Magezi DA. Linear mixed-effects models for within-participant psychology experiments: An introductory tutorial and free, graphical user interface (LMMgui). 2015 https://doi.org/10.3389/fpsyg.2015.00002
46. Magezi DA. Corrigendum: Linear mixed-effects models for within-participant psychology experiments: An introductory tutorial and free, graphical user interface (LMMgui). 2019 https://doi.org/10.3389/fpsyg.2019.00489
47. Martel C, Pennycook G, Rand DG. Reliance on emotion promotes belief in fake news. Cognitive Res Principl Impl. 2020. https://doi.org/10.1186/s41235-020-00252-3.
48. Nie Y, Williams A, Dinan E, et al. Adversarial NLI: A New Benchmark for Natural Language Understanding. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics. 2020. https://doi.org/10.8653/v1/2020.acl-main.441
49. Nyhan B, Settle J, Thorson E, et al. Like-minded sources on Facebook are prevalent but not polarizing. Nature. 2023;620(7972):137–44. https://doi.org/10.1038/s41586-023-06297-w.
50. Olteanu A, Castillo C, Diaz F, et al. Social data Biases. Methodol Pitfalls Ethical Boundaries. 2019. https://doi.org/10.3389/fdata.2019.00013.
51. OpenAI DALL-E 2. 2023a https://openai.com/dall-e-2, Accessed Aug 29 2023
52. OpenAI (2023b) GPT-4. https://openai.com/gpt-4, Accessed Aug 29 2023
53. Pedregosa F, Varoquaux G, Gramfort A, et al. Scikit-learn: Machine learning in python. J Machine Learn Res. 2011;12:2825–30.
54. Pellegrino A, Stasi A, Bhatiasevi V. Research trends in social media addiction and problematic social media use: a biometric. Front Psychiatry. 2022. https://doi.org/10.3389/fpsyt.2022.1017506.
55. Potthast M, Kiesel J, Reinartz K, et al. A stylometric inquiry into hyperpartisan and fake news. In: ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers), 2018 https://doi.org/10.18653/v1/p18-1022
56. Powell MJD. An efficient method for finding the minimum of a function of several variables without calculating derivatives. The Computer J. 1964. https://doi.org/10.1093/comjnl/7.2.155.
57. Poynter Institute 2020 PolitiFact. 2023 https://www.politifact.com/, Accessed 29 Aug 2023
58. Pummerer L, Böhm R, Lilleholt L, et al. Conspiracy theories and their societal effects during the COVID-19 pandemic. Soc Psychol Person Sci. 2021. https://doi.org/10.1177/19485506211000217.
59. Reddit Staff. Creating a healthy ecosystem for reddit data and reddit data API access. 2023 https://www.redditinc.com/blog/2023apiupdates, Accessed Aug 29 2023
60. Reimers N, Gurevych I. Sentence-BERT: Sentence embeddings using siamese BERT-networks. In: EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference, 2019 https://doi.org/10.18653/v1/d19-1410
61. Reimers N, Gurevych I. sentence-transformers/all-mpnet-base-v2. Hugging Face. 2021 https://huggingface.co/sentence-transformers/all-mpnet-base-v2, Accessed Aug 29 2023
62. Roesslein J. Tweepy. 2022 https://www.tweepy.org/, Accessed Aug 29 2023
63. Roozenbeek J, van der Linden S, Nygren T. Prebunking interventions based on the psychological theory of "inoculation" can reduce susceptibility to misinformation across cultures. Harvard Kennedy School Misinform Rev. 2020. https://doi.org/10.37016//mr-2020-008.
64. Sample C, Jensen MJ, Scott K, et al. Interdisciplinary lessons learned while researching fake news. Front Psychol. 2020. https://doi.org/10.3389/fpsyg.2020.537612.
65. Shao C, Ciampaglia GL, Varol O, et al. The spread of low-credibility content by social bots. Nature Commun. 2018. https://doi.org/10.1038/s41467-018-06930-7.
66. Shin J, Jian L, Driscoll K, et al. The diffusion of misinformation on social media: temporal pattern and source. Computer Human Behav. 2018. https://doi.org/10.1016/j.chb.2018.02.008.
67. Shu K, Sliva A, Wang S, et al. Fake news detection on social media. ACM SIGKDD Explorations Newsletter. 2017. https://doi.org/10.1145/3137597.3137600.
68. Siddiqui S, Singh T. Social media its impact with positive and negative aspects. Int J Computer Appl Technol Res. 2016. https://doi.org/10.7753/ijcatr0502.1006.
69. Simonov A, Sacher S, Dube JPH, et al. The persuasive effect of fox news: non-compliance with social distancing during the covid-19 pandemic. SSRN Electron J. 2020. https://doi.org/10.2139/ssrn.3600088.
70. Singhal A. Modern information retrieval: a brief overview. Bull IEEE Computer Soc Technical Committee Data Eng. 2001;24(4):35.
71. Snopes Media Group Inc. Snopes.com | The definitive fact-checking site and reference source for urban legends, folklore, myths, rumors, and misinformation. 2023 https://www.snopes.com/, Accessed Aug 29 2023
72. Stoica P, Selen Y. Model-order selection: a review of information criterion rules. IEEE Signal Process Magazine. 2004. https://doi.org/10.1109/MSP.2004.1311138.
73. Stremlau N, Gagliardone I, Price M, et al. World Trends in Freedom of Expression and Media Development: 2017/2018 Global Report. Paris: UNESCO; 2018.
74. Swift A. Americans' Trust in Mass Media Sinks to New Low. Tech. rep., Gallup, 2016 https://news.gallup.com/poll/195542/americans-trust-mass-media-sinks-new-low.aspx, Accessed Aug 29 2023
75. Tandoc EC, Lim ZW, Ling R. Defining "Fake News": A typology of scholarly definitions. 2018 https://doi.org/10.1080/21670811.2017.1360143
76. Tasnim S, Hossain M, Mazumder H. Impact of rumors and misinformation on COVID-19 in Social Media. 2020 https://doi.org/10.3961/JPMPH.20.094
77. Teovanović P, Lukić P, Zupan Z, et al. Irrational beliefs differentially predict adherence to guidelines and pseudoscientific practices during the COVID-19 pandemic. Appl Cogn Psychol. 2021. https://doi.org/10.1002/acp.3770.
78. Tornes A. Enabling the future of academic research with the Twitter API. 2021 https://developer.twitter.com/en/blog/product-news/2021/enabling-the-future-of-academic-research-with-the-twitter-api, Accessed Aug 29 2023
79. Treen KM, Williams HT, O'Neill SJ. Online misinformation about climate change. WIREs Climate Change 2020;11(5). https://doi.org/10.1002/wcc.665

80. Twitter Inc. GET /2/tweets/counts/all | Docs | Twitter Developer Platform. 2023a https://developer.twitter.com/en/docs/twitter-api/tweets/counts/api-reference/get-tweets-counts-all, Accessed Aug 29 2023
81. Twitter Inc. Tweet object | Docs | Twitter Developer Platform. 2023b https://developer.twitter.com/en/docs/twitter-api/data-dictionary/object-model/tweet, Accessed 29 August 2023
82. Vosoughi S, Roy D, Aral S. The spread of true and false news online. Science. 2018. https://doi.org/10.1126/science.aap9559.
83. Watts DJ, Rothschild DM, Mobius M. Measuring the news and its impact on democracy. Proc Natl Acad Sci United States of Am. 2021. https://doi.org/10.1073/pnas.1912443118.
84. Willingham A. Why Twitter users are upset about the platform's latest change. 2023 https://edition.cnn.com/2023/02/03/tech/twitter-api-what-is-pricing-change-cec/index.html
85. Wu L, Morstatter F, Carley KM, et al. Misinformation in social media: definition, manipulation, and detection. ACM SIGKDD Explorations Newsletter. 2019. https://doi.org/10.1145/3373464.3373475.
86. Zhang L, Wang X, Cooper E, et al. The PartialSpoof database and countermeasures for the detection of short fake speech segments embedded in an uterance. IEEE/ACM Trans Audio Speech Language Process. 2023. https://doi.org/10.1109/TASLP.2022.3233236.
87. Zhang X, Ghorbani AA. An overview of online fake news: characterization, detection, and discussion. Inform Process Manag. 2020. https://doi.org/10.1016/j.ipm.2019.03.004.

Publisher's Note