

# Towards Crowdsourcing Tasks for Accurate Misinformation Detection

Ronald Denaux<sup>1</sup>[0000-0001-5672-9915], Flavio Merenda<sup>1</sup>, and Jose Manuel Gomez-Perez<sup>1</sup>[0000-0002-5491-6431]\*

Expert System, Madrid, Spain {rdenaux, jmgomez}@expertsystem.com

**Abstract.** For all the recent advancements in Natural Language Processing and deep learning, current systems for misinformation detection are still woefully inaccurate in real-world data. Automated misinformation detection systems —available to the general public and producing explainable ratings— are therefore still an open problem and involvement of domain experts, journalists or fact-checkers is necessary to correct the mistakes such systems currently make. Reliance on such expert feedback imposes a bottleneck and prevents scalability of current approaches. In this paper, we propose a method —based on a recent semantic-based approach for misinformation detection, Credibility Reviews (CR)—, to (i) identify real-world errors of the automatic analysis; (ii) use the semantic links in the CR graphs to identify steps in the misinformation analysis which may have caused the errors and (iii) derive crowdsourcing tasks to pinpoint the source of errors. As a bonus, our approach generates real-world training samples which can improve existing datasets and the accuracy of the overall system.

**Keywords:** Disinformation Detection · Crowdsourcing · Credibility Signals · Explainability

## 1 Introduction

One of the reasons that makes misinformation a hard problem is that verifying a claim requires skills that only a fraction of the population have; typically well-educated domain experts, fact-checkers or journalists who know where to find verifying information for a particular domain. As a consequence fact-checking is a task that cannot easily be performed by crowdsource workers, who have different levels of education and which may lack specific domain knowledge. This bottleneck means in turn that it is difficult to train accurate, domain independent, automated systems to help in the fact-checking process as there is a relatively limited amount of fact-checks available. Furthermore, available fact-checks are highly biased towards claims of specific domains considered more important at the time, i.e. political claims during elections or health claims during pandemics.

---

\* Copyright © 2020 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

Several automated systems have been proposed [2,5,4,11] to help in misinformation detection tasks. However, their accuracy is still quite poor at the overall task of detecting misinforming claims, articles or social media posts in the wild. Ideally, these systems would catch misinformation before it is spread on social media, which means they should be accurate based on the content of the reviewed item. Current content-based systems only achieve about 72% accuracy [4] on datasets like FakeNewsNet, which are relatively easy as they (i) provide plenty of content (news articles), (ii) are simplified into a binary classification (fake or real), and (iii) which have been already reviewed by fact-checkers<sup>1</sup>.

In our previous work on Linked Credibility Reviews (LCRs) [4], we showed that our implementation, called *acred*, obtained state of the art results based on the following steps:

- simple content decomposition: basing the credibility of more complex documents like articles and tweets on its parts like sentences or linked articles and metadata like its publisher website. In our current implementation of *acred*, we have introduced a checkworthiness filter to only take into account sentences which are factual statements.<sup>2</sup>
- linking those sentences to a database of claims already reviewed. This linking was achieved using simple, domain-independent linguistic tasks such as semantic similarity and stance detection for which high accuracy deep learning models can be trained (92% accuracy on stance detection and 83 pearson correlation on semantic similarity, using RoBERTa)
- normalising existing evidence for:
  - *claims* from **ClaimReviews** provided by reputable fact-checkers and
  - *websites* from reputation scores by WebOfTrust, NewsGuard, and others.

Surprisingly, initial error analysis showed that most of the errors could be traced back to the sentence linking steps. One of the advantages of the LCR approach is that it generates a graph of sub-reviews, rather than just producing a single credibility label. In this paper we propose a method for exploiting the traceability of LCRs in order to (i) be able to crowdsource the error analysis process and (ii) derive new training samples for credibility review subtasks like semantic similarity and stance detection.

## 2 Problem and Intuition

Consider the tweet shown in Figure 1a. Using *acred*, we can generate a credibility review for that tweet, which we can show to the users in a couple of ways. The

<sup>1</sup> Social signals (replies, likes, etc.) provide further evidence which can improve accuracy[10,11], but can only be used *after* the content has spread.

<sup>2</sup> This is implemented as a RoBERTa model [6] finetuned on a combination of datasets: CBD [7], Clef’20 Task 1 (see <https://github.com/sshaar/clef2020-factchecking-task1>) and claims extracted from **ClaimReview** metadata. We obtain f1 weighted scores of 0.85 on Clef’19 Task 1 and 0.95 on 2020\_debate (see [https://github.com/idirlab/claimspotter/tree/master/data/two\\_class](https://github.com/idirlab/claimspotter/tree/master/data/two_class))

most concise way is shown as a bar on top of the tweet in Fig. 1a; the bar displays the acred credibility label for the tweet. To the right of the label, we see a couple of buttons that allow users to provide feedback about whether they agree (happy face) or disagree (sad face) with the label assigned by the system. In this case, the numbers indicate there’s a clear majority of users who disagree with the label, which tells us that something has gone wrong in acred’s analysis. The challenge is figuring out which step(s) in the acred analysis introduced errors. Fig. 2a shows the graph of all the evidence gathered and considered by acred in order to produce the “credible” label shown to the user. Each of the “meter” icons is a *sub-review* —e.g. a credibility review of one sentence in the tweet, or a similarity review between that sentence and some other sentence for which a credibility value is known— which contributed to the final rating, therefore any of those steps could have introduced an error, but which ones? Obviously we do not want to generate tasks for all 36 sub-reviews. Instead, we want to select the sub-reviews most likely to have produced the error. The rest of the paper discusses how to do that and what kind of crowdsourcing task could be used to find errors in the graph.

*Intuition for our approach* LCR bots, responsible for contributing the sub-reviews, will tend to apply heuristics to select certain sub-reviews (and discard others). In Figure 1b we see an interface showing a card for the final credibility review for the tweet. In essence, it is summarising the graph shown in Fig. 2a. The generated explanation clearly only uses some of the evidence in the graph. In particular, we see that the explanation hinges on just one of the sentences in the tweet and it *agreeing* with a similar sentence found on a website deemed to be credible. This chain of evidence is shown in Fig 2b, which is a subset of 7 (out of the initial 36) sub-reviews from 2a. In this sub-graph, all the sub-reviews directly contribute to the final label. Since the final label is erroneous, one or more of these evidence nodes must have introduced some error.<sup>3</sup>

### 3 Crowdacred

In this section we formalise the problem and our approach, called Crowdacred.

#### 3.1 Preliminaries

**Schema.org Reviews and Credibility Reviews** *Linked Credibility Reviews* (LCR) [4], is a linked data model for composable and explainable misinformation detection. A *Credibility Review* (CR) is an extension of the generic **Review** data model defined in Schema.org. A Review R can be conceptualised as a tuple  $(d, r, p)$  where R:

- reviews a *data item*  $d$ , via property `itemReviewed`, this can be any linked-data node (e.g. an article, claim or social media post).

<sup>3</sup> Note that some of the discarded sub-reviews may also be erroneous, but those errors did not contribute to the final label, hence we ignore them.



Fig. 1: Example UIs for a (dis)agreement task for a tweet. The user can provide feedback about correct or incorrect labels predicted by acred.

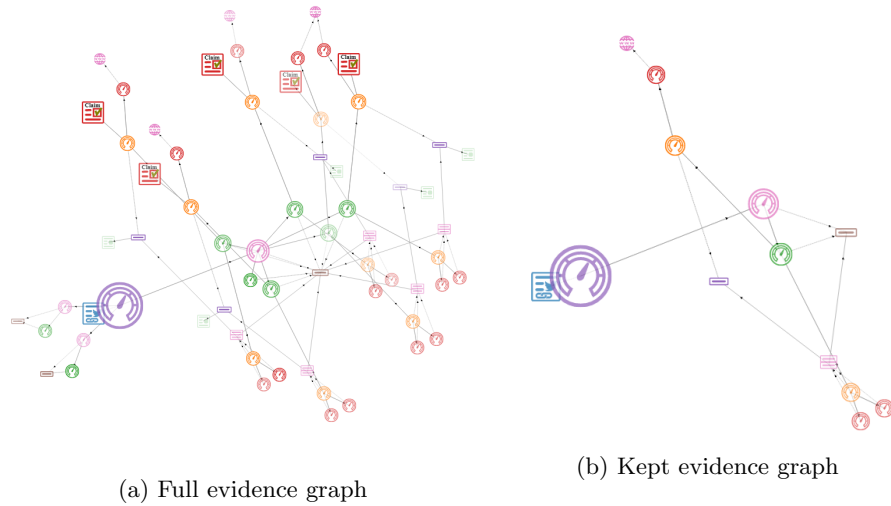


Fig. 2: Evidence graph for the credibility review and tweet shown in Fig. 1. The big “meter” icon represents the main credibility review, next to the icon for the tweet. All the other nodes form the **evidence** gathered by acred and used to determine the credibility of the tweet.

- assigns a numeric or textual *rating*  $r$  to (some, often implicit, `reviewAspect` of)  $d$ , via property `reviewRating`
- *optionally provides provenance information*  $p$ , e.g. via properties `author` and `isBasedOn`.

A Credibility Review (CR) is a subtype of Review, defined as a tuple  $\langle d, r, c, p \rangle$ , where the CR:

- $r$  must have `reviewAspect credibility` and is recommended to be expressed as a numeric value in range  $[-1, 1]$  and is qualified with a *rating confidence*  $c$  (in range  $[0, 1]$ ).
- the provenance  $p$  is mandatory and must include information about:
  - *credibility signals* (CS) used to derive the credibility rating, which can be either (i) Reviews for data items relevant to  $d$  or (ii) *ground credibility signals* (GCS) resources (which are not CRs) in databases curated by a trusted person or organization.
  - the *author* of the review. The author can be a person, organizations or bot. Bots are automated agents that produce CRs.

For this paper, the main thing to take into account is that the CR for a particular data item (e.g. a Tweet) is composed of many “sub reviews” which are available by following the provenance relation  $p$ . For any specific  $CR_i$ , we refer to the overall set of nodes  $V_i$  (Reviews, authors, data items and GCS) and links between them ( $E_i$ ) as the *Evidence Graph*  $G_i = (V_i, E_i)$  for  $CR_i$ .

**Crowdsourcing Review Tasks** A Crowdsourcing Review Task (subsequently simply referred as *task*)  $t$  is defined as a tuple  $\langle d, a, o \rangle$ , where  $d$  is a data item to be reviewed by the user;  $a$  is the aspect of  $d$  that needs to be reviewed; and  $o$  is a set of possible review values. Tasks need to be performed by human users, hence we require a function  $f_{\text{render}}$  which renders the task in a way that a user can inspect. The user performs the task by inspecting the rendering and selecting one of the available options, which produces a review of the form  $\langle d, r_a, p_u \rangle$ ; where  $r_a$  is a rating for aspect  $a$  and the `ratingValue` is one of the options in  $o$ .

### 3.2 Problem Statement and Overview

Given an unlabeled data item  $d$  and an automatically derived credibility review for it,  $CR_d = \langle d, r_d, c_d, p_d \rangle$  —and therefore its corresponding evidence graph  $G_d = (V_d, E_d)$ —, create simple tasks  $t_1, t_2, \dots, t_n$ , which can be performed by un-(or minimally)trained workers and which (i) allows us to decide whether  $r_d$  is accurate and (ii) if  $r_d$  is not accurate, identifies sub-reviews  $R_d^i \in V_d$  which directly caused the error. Furthermore, aim to minimise the number of tasks  $n$ .

In this paper, we propose a two-step method to derive such tasks:

1. collect agreement with overall rating  $r_d$
2. for ratings with high disagreement:
  - identify candidate reviews in the *evidence graph* for  $r_d$  and
  - derive tasks from the identified candidate reviews

### 3.3 Capturing Overall Agreement with Credibility Reviews

In this first step, we generate tasks for users to help us identify CR instances which have an inaccurate credibility rating. For this, we exploit the explainability of credibility ratings. We propose the following task:

Given a user  $u$  and a credibility review  $CR_d$  for data item  $d$ , we define  $t_{\text{agreement}} = \langle CR_d, \text{agreement}, o_{\text{agreement}} \rangle$  as a task where the user is shown a summary of  $CR_d$  (likely including a rendering of  $d$ ), and is asked to produce a rating  $o_{\text{agreement}} = \{\text{agree}, \text{disagree}\}$ . For this task we consider two specific rendering functions:

- **label** maps the values  $r_d$  and  $c_d$  onto a credibility label. For example,  $r_d > 0.5$  and  $c_d > 0.75$  could map to “credible”.
- **explain** generates a more complex textual explanation by following the provenance information  $p_d$  (recursively).

The result of  $t_{\text{agreement}}$  is an instance of a Review:  $(CR_d, r_{\text{agreement}}, p_u)$ . An example of such a task, using both rendering functions, is shown in figure 1.

Although this task is much easier than performing a full fact-check of an article or claim, it can still be cognitively demanding and some users may not have sufficient knowledge about the domain to make an informed decision. Therefore, we expect this to be a challenging task for most crowdsourcing workers. As part of the Co-inform project<sup>4</sup>, instead of relying on crowdsourcing workers, we are asking users of our browser plugin to provide such agreement ratings as an extension of their daily browsing and news consumption habits. As shown in fig. 1a, given sufficient users, a consensus can emerge enabling detection of erroneous reviews.

### 3.4 Finding Candidate Erroneous sub Reviews

Given a credibility review  $CR_d$  which users have rated as erroneous, in this step, we identify sub Reviews  $R_0, R_2, \dots, R_n$  which have directly contributed to the final rating and confidence in  $CR_d$ . Recall that  $p_d$  provides provenance information that can be used. In *acred*, the relevant provenance is implemented by providing a list of sub-reviews via property `isBasedOn`. This list contains references to all the signals *taken into account* to derive the rating but in many cases, the majority of these signals are discarded via aggregation functions (e.g. selecting the subreview with highest confidence or with lowest credibility rating [4]). Therefore, we propose to define two disjoint subproperties of `isBasedOn`: `isBasedOnDiscarded` and `isBasedOnKept`.

Using these new subproperties we can define a subgraph  $G_d^{\text{kept}}$  of  $G_d$ , which contains only those nodes which can be linked to the final  $CR_d$  via `isBasedOnKept` edges. To illustrate this idea, figure 2a shows an example of a full evidence graph, while figure 2b shows only the kept subgraph for the same credibility review. As can be seen from the figures, this step greatly reduces the number of candidate sub reviews, while also ensuring that those reviews directly contributed to the final (presumably erroneous) rating.

<sup>4</sup> <https://coinform.eu/>

### 3.5 Defining Crowdsourcing Tasks

Now that we have identified a small number of sub-reviews which directly influence the final credibility rating, we can use crowdsourcing to identify which steps contributed erroneous evidence. Although we could define user agreement tasks for the individual steps, we can get more actionable information by asking more specific questions to the users. For this, we need to define custom tasks for each step in *acred*. Preliminary error analyses in [4] showed that most of the errors were caused by the linking steps, therefore we discuss three specific types of Reviews used in *acred* and how to derive crowdsourcing tasks for them.

**SentenceCheckworthinessReview** determines whether a Sentence is check-worthy or not. This is the case when the sentence is both factual (i.e. not an opinion or question) and verifiable (someone can, in principle, find out whether the sentence is accurate or not). We derive task  $t_{\text{checkworthy}}$  where  $o_{\text{checkworthy}} = \{\text{checkworthy}, \text{notFactual}, \text{notVerifiable}\}$ . Table 1 shows an example rendering (and expected answer), based on the sub-reviews in Figures 2b and 1b.

Help us to detect if a sentence contains a factual claim
Do you think the following sentence contains a factual claim?
<ul style="list-style-type: none"> <li>– “The vast amounts of money made and stolen by China from the United States, year after year, for decades, will and must STOP.”</li> </ul>
<input checked="" type="checkbox"/> Yes, and the claim can be verified <input type="checkbox"/> Yes, but nobody could verify it <input type="checkbox"/> No

Table 1: Example SentenceCheckworthinessReview task

**SentenceSimilarityReview** assigns a similarity score to a pair of sentences  $\langle s_a, s_b \rangle$ .<sup>5</sup> There are existing crowdsourcing tasks defined for this [1], including instructions and a rating schema, which we can reuse to define  $t_{\text{sentenceSimilarity}} = \langle d, \text{sentenceSimilarity}, o_{\text{sentenceSimilarity}} \rangle$ . The schema,  $o_{\text{sentenceSimilarity}}$  consists of a scale of 6 values ranging from 0 (the two sentences are completely dissimilar) to 5 (the two sentences are completely equivalent, as they mean the same thing). See table 2 for an example.

**SentenceStanceReview** assigns a stance label describing the relation between a pair of sentences.<sup>6</sup> Although there are many existing datasets [9] for this

<sup>5</sup> This is implemented in *acred* via a RoBERTa model that has been fine-tuned on STS-B [3], which has in part been derived from previous semantic similarity tasks [1].

<sup>6</sup> This is implemented in *acred* via another RoBERTa model that has been fine-tuned on FNC-1 [8].

Help us to detect how similar are two sentences
Choose one of the options that describes the semantic similarity grade between the following pair of sentences.
<ul style="list-style-type: none"> <li>– “The vast amounts of money made and stolen by China from the United States, year after year, for decades, will and must STOP.”</li> <li>– ”The US still supplies much more goods from China and the EU than vice versa.“</li> </ul>
The two sentences are:
<input type="checkbox"/> completely equivalent, as they mean the same thing <input type="checkbox"/> mostly equivalent, but some unimportant details differ <input checked="" type="checkbox"/> roughly equivalent, but some important information differs/missing <input type="checkbox"/> not equivalent, but share some details <input type="checkbox"/> not equivalent, but are on the same topic <input type="checkbox"/> on different topics

Table 2: Example SentenceSimilarityReview Task

problem, they differ in their target labels. We find FNC-1[8] labels (*agree*, *disagree*, *discuss* and *unrelated*) provide a good balance as other datasets often are missing a label for the *unrelated* case. Also, the FNC-1 labels have the advantage that they describe symmetric relations (although this is arguable for *discuss*), while other datasets use asymmetric relations like *query*. Therefore we define tasks  $t_{\text{sentenceStance}} = \langle d, \text{sentenceStance}, o_{\text{sentenceStance}} \rangle$  where  $o_{\text{sentenceStance}} = \{\text{agree, disagree, discuss, unrelated}\}$ . Table 3 shows an example of such a task.

Help us to better understand the relation between two sentences
Choose one of the options that describes the relation between the following sentences.
<ul style="list-style-type: none"> <li>– “The vast amounts of money made and stolen by China from the United States, year after year, for decades, will and must STOP.”</li> <li>– ”The US still supplies much more goods from China and the EU than vice versa.“</li> </ul>
The two sentences:
<input type="checkbox"/> agree with each other <input type="checkbox"/> disagree with each other <input checked="" type="checkbox"/> discuss the same issue <input type="checkbox"/> are unrelated


Table 3: Example SentenceStanceReview Task

## 4 Summary and Future Work

In this paper, we presented Crowdacred, a method for extending Linked Credibility Reviews to be able to crowdsource (i) the detection of inaccurate credibility reviews, (ii) the error analysis or erroneous reviews and (iii) generation of realistic sample data for NLP subtasks needed for accurate misinformation detection. We are currently implementing the proposed method on top of acred [4] and plan



to run initial crowdsourcing experiments to validate the approach. The validation study will be based on a core set of (a few dozens) users from Co-inform<sup>7</sup> and a larger pool of crowdsource workers. If successful, we aim to be able to produce new datasets of contents in the wild on specific topics like covid-19.

*Acknowledgements* Work supported by the European Commission under grant 770302 – Co-Inform – as part of the Horizon 2020 research and innovation programme. 

## References

1. Agirre, E., Cer, D., Diab, M., Gonzalez-Agirre, A., Guo, W.: \*SEM 2013 shared task: Semantic textual similarity. In: Second Joint Conference on Lexical and Computational Semantics (\*SEM). pp. 32–43. Association for Computational Linguistics, Atlanta, Georgia, USA (Jun 2013)
2. Babakar, M., Moy, W.: The State of Automated Factchecking. Tech. rep. (2016)
3. Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., Specia, L.: SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Crosslingual Focused Evaluation. In: Proc. of the 10th International Workshop on Semantic Evaluation. pp. 1–14 (2018)
4. Denaux, R., Perez-Gomez, J.M.: Linked Credibility Reviews for Explainable Misinformation Detection. In: 19th International Semantic Web Conference (nov 2020), <https://arxiv.org/abs/2008.12742>
5. Hassan, N., Zhang, G., Arslan, F., Caraballo, J., Jimenez, D., Gawsane, S., Hasan, S., Joseph, M., Kulkarni, A., Nayak, A.K., Sable, V., Li, C., Tremayne, M.: Claim buster: The first-ever end-to-end fact-checking system. In: Proceedings of the VLDB Endowment. vol. 10, pp. 1945–1948 (2017)
6. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: RoBERTa: A Robustly Optimized BERT Pretraining Approach. Tech. rep. (2019)
7. Meng, K., Jimenez, D., Arslan, F., Devasier, J.D., Obembe, D., Li, C.: Gradient-Based Adversarial Training on Transformer Networks for Detecting Check-Worthy Factual Claims (feb 2020), <http://arxiv.org/abs/2002.07725>
8. Pomerleau, D., Rao, D.: The fake news challenge: Exploring how artificial intelligence technologies could be leveraged to combat fake news (2017)
9. Schiller, B., Daxenberger, J., Gurevych, I.: Stance Detection Benchmark: How Robust Is Your Stance Detection? (jan 2020)
10. Shu, K., Mahudeswaran, D., Wang, S., Lee, D., Liu, H.: FakeNewsNet: A Data Repository with News Content, Social Context and Spatialtemporal Information for Studying Fake News on Social Media. Tech. rep. (2018)
11. Shu, K., Zheng, G., Li, Y., Mukherjee, S., Awadallah, A.H., Ruston, S., Liu, H.: Leveraging Multi-Source Weak Social Supervision for Early Detection of Fake News (2020), <http://arxiv.org/abs/2004.01732>

---

<sup>7</sup> <https://coinform.eu/>