Linked Credibility Reviews for Explainable Misinformation Detection

Ronald Denaux[1][0000−0001−5672−9915] and Jose Manuel Gomez-Perez[1][0000−0002−5491−6431]

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

Abstract. In recent years, misinformation on the Web has become increasingly rampant. The research community has responded by proposing systems and challenges, which are beginning to be useful for (various subtasks of) detecting misinformation. However, most proposed systems are based on deep learning techniques which are fine-tuned to specific domains, are difficult to interpret and produce results which are not machine readable. This limits their applicability and adoption as they can only be used by a select expert audience in very specific settings. In this paper we propose an architecture based on a core concept of Credibility Reviews (CRs) that can be used to build networks of distributed bots that collaborate for misinformation detection. The CRs serve as building blocks to compose graphs of (i) web content, (ii) existing credibility signals –fact-checked claims and reputation reviews of websites–, and (iii) automatically computed reviews. We implement this architecture on top of lightweight extensions to Schema.org and services providing generic NLP tasks for semantic similarity and stance detection. Evaluations on existing datasets of social-media posts, fake news and political speeches demonstrates several advantages over existing systems: extensibility, domain-independence, composability, explainability and transparency via provenance. Furthermore, we obtain competitive results without requiring finetuning and establish a new state of the art on the Clef'18 CheckThat! Factuality task.

Keywords: Disinformation Detection · Credibility Signals · Explainability · Composable Semantics

1 Introduction

Although misinformation is not a new problem, the Web –due to the pace of news cycles combined with social media, and the information bubbles it creates– has increasingly evolved into an ecosystem where misinformation can thrive[8] with various societal effects. Tackling misinformation[1] is not something that can be achieved by a single organization –as evidenced by struggling efforts by the major social networks– as it requires decentralisation, common conceptualisations, transparency and collaboration[3].

[1] https://ec.europa.eu/digital-single-market/en/tackling-online-disinformation

Technical solutions for computer-aided misinformation detection and fact-checking have recently been proposed[1,6] and are essential due to the scale of the Web. However, a lack of hand-curated data, maturity and scope of current AI systems, means assessing *veracity*[12] is not feasible. Hence the value of the current systems is not so much their accuracy, but rather their capacity of retrieving potentially relevant information that can help human fact-checkers, who are the main intended users of such systems, and are ultimately responsible for verifying/filtering the results such systems provide. Therefore, a main challenge is developing automated systems which can help the *general public*, and influencers in particular, to assess the credibility of web content, which requires explainable results by AI systems. This points towards the need for hybrid approaches that enable the use of the best of deep learning-based approaches, but also of symbolic knowledge graphs to enable better collaboration between large platforms, fact-checkers, the general public and other stakeholders like policy-makers, journalists, webmasters and influencers.

In this paper, we propose a design on how to use semantic technologies to aid in resolving such challenges. Our contributions are:

- a datamodel and architecture of distributed agents for composable credibility reviews, including a lightweight extension to `schema.org` to support provenance and explainability (section 3)
- an implementation of the architecture demonstrating feasibility and value (section 4)
- an evaluation on various datasets establishing state-of-the-art in one dataset (Clef'18 CheckThat! Factuality task) and demonstrating capabilities and limitations of our approach, as well as paths for improvements (section 5)

2 Related Work

The idea of automating (part of) the fact-checking process is relatively recent[1]. ClaimBuster[6] proposed the first automated fact-checking system and its architecture is mostly still valid, with a database of fact-checks and components for monitoring web sources, spotting claims and matching them to previously fact-checked claims. Other similar services and projects include Truly media[2], invid[3] and CrowdTangle[4]. These systems are mainly intended to be used by professional fact-checkers or journalists, who can evaluate whether the retrieved fact-check article is relevant for the identified claim. These automated systems rarely aim to predict the accuracy of the content; this is (rightly) the job of the journalist or fact-checker who uses the system. Many of these systems provide valuable REST APIs to access their services, but as they use custom schemas, they are difficult to compose and inspect as they are not machine-interpretable or explainable.

[2] https://www.truly.media/ https://www.disinfobservatory.org/
[3] https://invid.weblyzard.com/
[4] https://status.crowdtangle.com/

Besides full-fledged systems for aiding in fact-checking, there are also various strands of research focusing on specific computational tasks needed to identify misinformation or assess the accuracy or veracity of web content based on ground credibility signals. Some low-level NLP tasks include check-worthiness[11] and stance detection[15,14], while others aim to use text classification as a means of detecting deceptive language[13]. Other tasks mix linguistic and social media analysis, for example to detect and classify rumours[20]. Yet others try to assess veracity of a claim or document by finding supporting evidence in (semi)structured data[18]. These systems, and many more, claim to provide important information needed to detect misinformation online, often in some very specific cases. However without a clear conceptual and technical framework to integrate them, the signals such systems provide are likely to go unused and stay out of reach of users who are exposed to misinformation.

The Semantic Web and Linked Data community has also started to contribute ideas and technical solutions to help in this area: perhaps the biggest impact has been the inclusion in Schema.org[5] of the `ClaimReview` markup[5], which enables fact-checkers to publish their work as machine readable structured data. This has enabled aggregation of such data into knowledge graphs like ClaimsKG [17], which also performs much needed normalisation of labels, since each fact-checker uses its own set of labels. A conceptual model and RDF vocabulary was proposed to distinguish between the utterance and propositional aspects of claims [2]. It allows expressing fine-grained provenance (mainly of annotations on the text of the claim), but still relies on `ClaimReview` as the main accuracy describing mechanism. It is unclear whether systems are actually using this RDF model to annotate and represent claims as the model is heavyweight and does not seem to align well with mainstream development practices. In this paper, we build on these ideas to propose a lightweight model which focuses on the introduction of a new type of Schema.org Review that focuses on *credibility* rather than *factuality*[6].

The focus on credibility, defined as an estimation of factuality based on available signals or evidence, is inspired by MisinfoMe[10,9] which borrows from social science, media literacy and journalism research. MisinfoMe focuses on credibility of sources, while we expand this to credibility of any web content and integrate some of MisinfoMe's services in our implementation to demonstrate how our approach enables composition of such services. There is also ongoing work on W3C Credibilty Signals[7], which aims to define a vocabulary to specify *credibility indicators* that may be relevant for assessing the credibility of some web content. To the best of our knowledge, this is still work in progress and no systems are implementing the proposed vocabularies.

[5] https://www.blog.google/products/search/fact-check-now-available-google-search-and-news-around-world/

[6] In our opinion, current AI systems cannot truly assess veracity since this requires human skills to access and interpret new information and relate them to the world.

[7] https://credweb.org/signals-beta/

3 Linked Credibility Reviews

This section presents *Linked Credibility Reviews* (LCR), our proposed linked data model for composable and explainable misinformation detection. As the name implies, *Credibility Review*s (CR) are the main resources and outputs of this architecture. We define a CR as a tuple $\langle d, r, c, p \rangle$, where the CR:

- reviews a *data item* d, this can be any linked-data node but will typically refer to articles, claims, websites, images, social media posts, social media accounts, people, publishers, etc.
- assigns a *credibility rating* r to the *data item* under review and qualifies that rating with a *rating confidence* c.
- *provides provenance information* p about:
  - *credibility signals* used to derive the credibility rating. Credibility Signals (CS) can be either (i) CRs for data items relevant to the data item under review 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, organization or bot. Bots are automated agents that produce CRs for supported data items based on a variety of strategies, discussed below.

The credibility rating is meant to provide a subjective (from the point-of-view of the author) measure of how much the credibility signals support or refute the content in data item. Provenance information is therefore crucial as it allows humans —e.g. end-users, bot developers— to retrace the CRs back to the ground credibility signals and assess the accuracy of the (possibly long) chain of bots (and ultimately humans) that were involved in reviewing the initial data item. It also enables the generation of explanations for each step of the credibility review chain in a composable manner as each bot (or author) can describe its own strategy to derive the credibility rating based on the used credibility signals.

Bot Reviewing Strategies CR bots are developed to be able to produce CRs for specific data item types. We have identified a couple of generic strategies that existing services seem to implement and which can be defined in terms of CRs (these are not exhaustive, though see figure 2 for a depiction of how they can collaborate):

- **ground credibility signal lookup** from some trusted source. CR bots that implement this strategy will (i) generate a query based on d and (ii) convert the retrieved ground credibility signal into a CR;
- **linking** the item-to-review d with n other data items d'_i *of the same type*, for which a $\text{CR}_{d'_i}$ is available. These bots define functions f_r, f_c and f_{agg}. The first two, compute the new values r_i and c_i based on the original values and the relation or similarity between d and d'_i i.e. $r_i = f_r(\text{CR}_{d'_i}, d, d')$. These produce n credibility reviews, CR^i_d, which are then aggregated into $\text{CR}_d = f_{\text{agg}}(\{CR^i_d \mid 0 \leq i < n\})$.

– **decomposing** whereby the bot identifies relevant parts d'_i of the item-to-review d and requests CRs for those parts $CR_{d'_i}$. Like the linking bots, these require deriving new credibility ratings CR_{d_i} and confidences based on the relation between the whole and the parts; and aggregating these into the CR for the whole item. The main difference is that the parts can be items of different types.

Representing and Aggregating Ratings For ease of computation, we opt to represent credibility ratings and their confidences as follows:

– $r \in \Re$, must be in the range of $[-1.0, 1.0]$ where -1.0 means not credible and 1.0 means credible
– $c \in \Re$, must be in the range of $[0.0, 1.0]$ where 0.0 means no confidence at all and 1.0 means full confidence in the accuracy of r, based on the available evidence in p.

This representations makes it possible to define generic, relatively straightforward aggregation functions like:

– $f^{\text{mostConfident}}$ which selects CR_i which has the highest confidence value c
– $f^{\text{leastCredible}}$ which selects the CR_i which has the lowest value r

3.1 Extending `schema.org` for LCR

While reviewing existing ontologies and vocabularies which could be reused to describe the LCR model, we noticed that `schema.org`[5] was an excellent starting point since it already provides suitable schema types for data items on the web for which credibility reviews would be beneficial (essentially any schema type that extends `CreativeWork`). It already provides suitable types for `Review`, `Rating`, as well as properties for expressing basic provenance information and meronymy (`hasPart`). Some basic uses and extensions compliant with the original definitions are:

– Define `CredibilityReview` as an extension of `schema:Review`, whereby the `schema:reviewAspect` is `credibility`[8]
– use `schema:ratingValue` to encode r
– add a `confidence` property to `schema:Rating` which encodes the rating confidence c.
– use `isBasedOn` to record that a CR was computed based on other CRs. We also use this property to describe dependencies between CR Bots, even when those dependencies have not been used as part of a CR.
– use `author` to link the CR with the bot that produced it

The main limitation we encountered with the existing definitions was that `CreativeWork`s (including `Review`s) are expected to be created only by `Person`s or `Organization`s, which excludes reviews created by bots. We therefore propose to extend this definition by:

[8] Note that `ClaimReview` is not suitable since it is overly restrictive: it can only review `Claim`s (and it assumes the review aspect is, implicitly, `accuracy`).

– introducing a type `Bot` which extends `SoftwareApplication`
– allowing `Bots` to be the `authors` of `CreativeWorks`.

Finally, in this paper we focus on *textual* misinformation detection and found we were missing a crucial type of `CreativeWork`, namely `Sentences`. Recently, a `Claim` type was proposed, to represent factually-oriented sentences and to work in tandem with the existing `ClaimReview`, however, automated systems still have trouble determining whether a sentence is a claim or not, therefore, CR bots should be able to review the credibility of `Sentences` and relevant aspects between pairs of sentences such as their stance and similarity. The overall `schema.org` based data model is depicted in figure 1, focused on CRs for textual web content (we leave other modalities as future work).

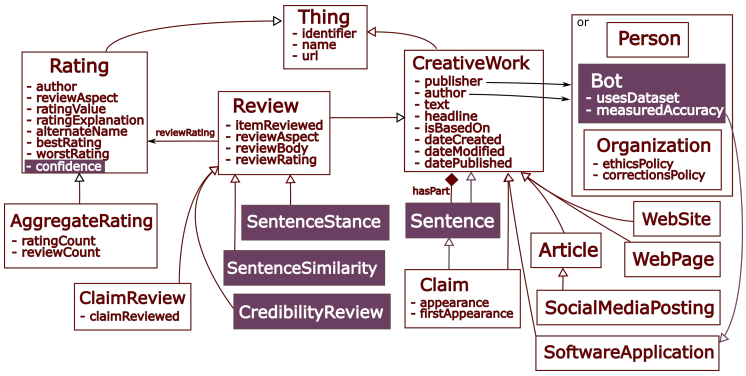


Fig. 1: Linked Credibility Review data model, extending schema.org.

4 acred – Deep Learning-based CR bots

To demonstrate the potential of the Linked Credibility Review architecture, we have implemented a series of CR bots capable of collaborating to review articles, tweets, sentences and websites.[9] We present the conceptual implementation in sections 4.1 to 4.4 and provide further details in section 4.5.

4.1 Ground Credibility Signal Sources

Ultimately we rely on two ground credibility signal sources:

– A database of `ClaimReviews` which provide accuracy ratings for factual claims by a variety of fact-checkers.

[9] The source code is available at https://github.com/rdenaux/acred

– Third-party, well-established services for validating `WebSite`s, such as NewsGuard and Web Of Trust[10], which rely on either expert or community-based ratings.

4.2 GCS Lookup Bots

The two GCS sources are mediated via two GCS lookup bots.

The $\texttt{LookupBot}_{\texttt{ClaimReview}}$ returns a CR for a `Claim` based on a `ClaimReview` from the database. In order to derive a CR from a `ClaimReview`, the accuracy rating in the `ClaimReview` need to be converted into equivalent credibility ratings. The challenge here is that each fact-checker can encode their review rating as they see fit. The final review is typically encoded as a textual `alternateName`, but sometimes also as a numerical `ratingValue`. ClaimsKG already performs this type of normalisation into a set of "veracity" labels, but for other `ClaimReview`s we have developed a list of simple heuristic rules to assign a credibility and confidence score.

The $\texttt{LookupBot}_{\texttt{WebSite}}$ returns a CR for a `WebSite`. This is a simple wrapper around the existing MisinfoMe aggregation service [10], which already produces credibility and confidence values.

4.3 Linking Bots

Although the GCS lookup bots provide access to the basic credibility signals, they can only provide this for a relatively small set of claims and websites. Misinformation online often appears as variations of fact-checked claims and can appear on a wide variety of websites that may not have been reviewed yet by a human. Therefore, to increase the number of sentences which can be reviewed, we developed the following linking bots (see Section 4.5 for further details).

The $\texttt{LinkBot}^{\texttt{PreCrawled}}_{\texttt{Sentence}}$ uses a database of pre-crawled sentences extracted from a variety of websites. A proprietary NLP system[11] extracts the most relevant sentences that may contain factual information in the crawled documents. This is done by identifying sentences that (i) are associated with a topic (e.g. Politics or Health) and (ii) mentions an entity (e.g. a place or person). The CR for the extracted sentence is assigned based on the website where the sentence was found (i.e. by using the $\texttt{LookupBot}_{\texttt{WebSite}}$). Since not all sentences published by a website are as credible as the site, the resulting CR for the sentence has a lower confidence than the CR for the website itself.

The $\texttt{LinkBot}^{\texttt{SemSim}}_{\texttt{Sentence}}$ is able to generate CRs for a wide variety of sentences by linking the input sentence s_i to sentences s_j for which a CR is available (via other bots). This linking is achieved by using a neural sentence encoder $f_{\text{enc}} : S \mapsto \Re^d$ –where S is the set of sentences and $d \in N^+$–, that is optimised to encode semantically similar sentences close to each other in an embedding space. The bot creates an index by generating embeddings for all the sentences reviewed

[10] https://www.newsguardtech.com/, https://www.mywot.com/
[11] http://expert.ai

by the $\texttt{LookupBot}_{\texttt{ClaimReview}}$ and the $\texttt{LinkBot}^{\texttt{PreCrawled}}_{\texttt{Sentence}}$. The incoming sentence, s_i is encoded and a nearest neighbor search produces the closest matches along with a similarity score based on a similarity function $f_{\text{sim}} : \Re^d \times \Re^d \mapsto \Re^{[0,1]}$. Unfortunately, most sentence encoders are not polarity aware so that negations of a phrase are considered similar to the original phrase; therefore we use a second neural model for stance detection $f_{\text{stance}} : S \times S \mapsto \text{SL}$, where SL is a set of stance labels. We then define $f_{\text{polarity}} : \text{SL} \mapsto \{1, -1\}$, which we use to invert the polarity of r_j if s_i disagrees with s_j. We also use the predicted stance to revise the similarity score between s_i and s_j by defining a function $f_{\text{reviseSim}} : \text{SL}, \Re^{[0,1]} \mapsto \Re^{[0,1]}$. For example, stances like *unrelated* or *discuss* may reduce the estimated similarity, which is used to revise the confidence of the original credibility. In summary, the final CR for s_i is selected via $f^{\text{mostConfident}}$ from a pool of $\text{CR}_{i,j}$ for the matching s_j; where the rating and confidences for each $\text{CR}_{i,j}$ are given by:

$$r_i = r_j \times f_{\text{polarity}}(f_{\text{stance}}(s_i, s_j))$$

$$c_i = c_j \times f_{\text{reviseSim}}\Big(f_{\text{stance}}(s_i, s_j), f_{\text{sim}}\big(f_{\text{enc}}(s_i), f_{\text{enc}}(s_j)\big)\Big)$$

4.4 Decomposing Bots

By combining linking and GCS lookup bots we are already capable of reviewing a wide variety of sentences. However, users online encounter misinformation in the form of high-level `CreativeWork`s like social media posts, articles, images, podcasts, etc. Therefore we need bots which are capable of (i) dissecting those CreativeWorks into relevant parts for which CRs can be calculated and (ii) aggregating the CRs for individual parts into an overall CR for the whole `CreativeWork`. In acred, we have defined two main types:

- $\texttt{DecBot}_{\texttt{Article}}$ reviews `Article`s, and other long-form textual `CreativeWork`s
- $\texttt{DecBot}_{\texttt{SocMedia}}$ reviews `SocialMediaPost`s

In both bots, decomposition works by performing NLP and content analysis on the title and textual content of the `CreativeWork` d_i. This results in a set of parts $P = \{d_j\}$ which include `Sentence`s, linked `Article`s or `SocialMediaPost`s and metadata like the `WebSite` where d was published. Each of these can be analysed either recursively or via other bots, which results in a set of reviews $\{\text{CR}_j\}$ for the identified parts. We define a function f_{part} which maps CR_j onto $\text{CR}_{\text{i,j}}$, which takes into account the relation between d_i and d_j as well as the provenance of CR_j to derive the credibility rating and confidence. The final CR_i is selected from all $\text{CR}_{i,j}$ via $f^{\text{leastCredible}}$.

Figure 2 shows a diagram depicting how the various CR bots compose and collaborate to review a tweet.

4.5 Implementation details

Our database of `ClaimReview`s contains 45K claims and was based on public resources such as ClaimsKG [17] (32K), data-commons (9.6K) and our in-house

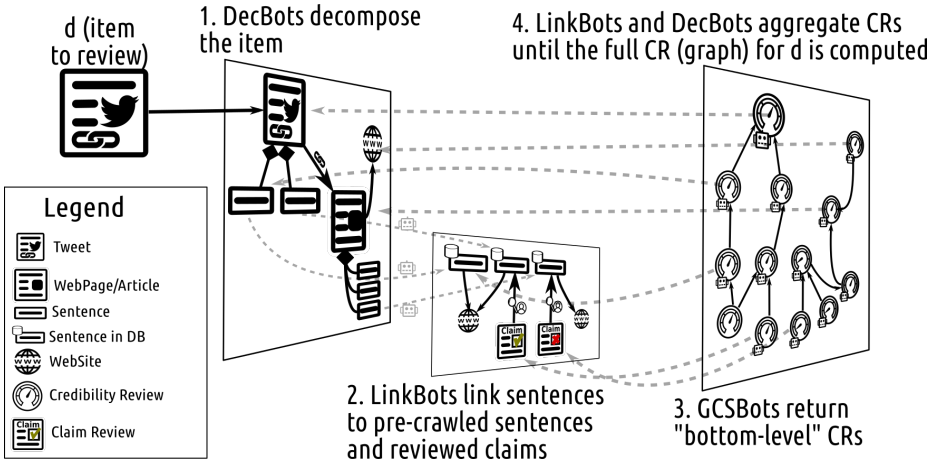


Fig. 2: Depiction of acred bots collaborating to produce a CR for a tweet.

developed crawlers (4K). The database of pre-crawled sentences contained 40K sentences extracted from a variety of generic news sites on-line between april 2019 and april 2020. It consisted primarily in relatively well-regarded news sites like `expressen.se`, `krone.at`, `zdf.de`, which combined for about 35K of the sentences, and a long tail of other sites including `theconversation.com` and `heartland.org`. For reproducibility, we will publish the list of sentences along associated URLs.

Our heuristic rules to normalise `ClaimReviews` are implemented in about 50 lines of python to map numeric `ratingValue`s (in context of specified `bestRating` and `worstRating` values) and 150 lines of python to map about 100 `alternateName` values (e.g "inaccurate", "false and misleading", "this is exaggerated") and about 20 patterns (e.g. "wrong.*", "no, *") into estimated c, r values.

The sentence encoder, f_{enc} in $\texttt{LinkBot}^{\texttt{SemSim}}_{\texttt{Sentence}}$ is implemented as a RoBERTa-base [7] model finetuned on STS-B [4]. We employ a siamese structure as this enables us to perform encoding of the claims off-line (slow) and comparison on-line (fast) at the cost of some accuracy. Our model achieves 83% Pearson correlation on STS-B dev.

The stance detector, f_{stance}, is also a RoBERTa-base model trained on FNC-1 [14], which assigns a stance label (either "agree", "disagree", "discuss" or "unrelated") to pairs of texts: about 50K for training and about 25K for testing. Our model achieves 92% accuracy on the held-out test set. See our GitHub repository for links to model weights and jupyter notebooks replicating our fine-tuning procedure.

Each implemented bot defines a set of templates to generate textual explanations. These reflect processing performed by the bot in a way that can be inspected by a user. Produced CRs use the `schema:ratingExplanation` property to encode the generated explanations and use markdown to take advantage

of hypertext capabilities like linking and formatting of the explanations. Examples are presented in table 1.

Different CR bots are deployed as separate Docker images and expose a REST API accepting and returning JSON-LD formatted requests and responses. They are all deployed on a single server (64GB RAM, Intel i7-8700K CPU @ 3.70GHzn with 12 cores) via docker-compose. The `ClaimReview` and pre-crawled sentence databases are stored in a Solr instance. The index of encoded sentences is generated off-line on a separate server with a GPU by iterating over the claims and sentences in Solr, and loaded into memory on the main server at run-time.

5 Evaluation

One of the main characteristics of the LCR architecture is that CR bots can be distributed across different organizations. This has the main drawback that it can be more difficult to fine-tune bots to specific domains since top-level bots do not have direct control on how lower-level bots are implemented and fine-tuned. Therefore in this paper we first evaluated our `acred` implementation, described above, on a variety of datasets covering social media posts, news articles and political speeches. Our rationale is that existing, non-distributed fact-checking approaches have an edge here as they can fine-tune their systems based on training data and therefore provide strong baselines to compare against. We used the explanations, along with the provenance trace, to perform error analysis[12] on the largest dataset, described in Section 5.2. This showed `acred` was overly confident in some cases. To address this, we introduced a modified version, `acred`$^{+}$ with custom functions to reduce the confidence and rating values of two bots under certain conditions: $\texttt{DecBot}_{\texttt{Article}}$ when based only on a website credibility; $\texttt{LinkBot}^{\texttt{SemSim}}_{\texttt{Sentence}}$ when the stance is "unrelated" or "discuss".

5.1 Datasets

The first dataset we use is the Clef'18 CheckThat! Factuality Task [11] (`clef18`). The task consists in predicting whether a check-worthy claim is either `true`, `half-true` or `false`. The dataset was derived from fact-checked political debates and speeches by `factcheck.org`. For our evaluation we only use the English part of this dataset[13] which contains 74 and 139 claims for training and testing.

FakeNewsNet [16] aims to provide a dataset of fake and real news enriched with social media posts and context sharing those news. In this paper we only use the fragment derived from articles fact-checked by Politifact that have textual content, which consists of 420 *fake* and 528 *real* articles. The articles were retrieved by following the instructions on the Github page[14].

[12] Note that usability evaluation of the generated explanations is not in the scope of this paper.

[13] Our implementation has support for machine translation of sentences, however this adds a confounding factor hence we leave this as future work.

[14] https://github.com/KaiDMML/FakeNewsNet, although we note that text for many of the articles could no longer be retrieved, making a fair comparison difficult.

Table 1: Example explanations generated by our bots.

Bot	Example explanation
$\texttt{LookupBot}_{\texttt{ClaimRev}}$	Claim `'Ford is moving all of their small-car productin to Mexico.'` is *mostly not credible* based on a fact-check by politifact with normalised numeric ratingValue 2 in range [1-5]
$\texttt{LookupBot}_{\texttt{WebSite}}$	Site `www.krone.at` seems *mostly credible* based on 2 review(s) by external rater(s) NewsGuard or Web Of Trust
$\texttt{LinkBot}^{\texttt{PreCrawled}}_{\texttt{Sentence}}$	Sentence `Now we want to invest in the greatest welfare program in modern times.` seems *credible* as it was published in site `www.expressen.se`. (Explanation for WebSite omitted)
$\texttt{LinkBot}^{\texttt{SemSim}}_{\texttt{Sentence}}$	Sentence `When Senator Clinton or President Clinton asserts that I said that the Republicans had had better economic policies since 1980, that is not the case.` seems *not credible* as it agrees with sentence: `Obama said that 'since 1992, the Republicans have had all the good ideas...'` that seems *not credible* based on a fact-check by politifact with textual rating 'false'. Take into account that the sentence appeared in site `www.cnn.com` that seems *credible* based on 2 review(s) by external rater(s) NewsGuard or Web Of Trust
$\texttt{LinkBot}^{\texttt{SemSim}}_{\texttt{Sentence}}$	Sentence `Can we reduce our dependence on foreign oil and by how much in the first term, in four years?` is similar to and discussed by: `Drilling for oil on the Outer Continental Shelf and in parts of Alaska will 'immediately reduce our dangerous dependence on foreign oil.'` that seems *not credible*, based on a fact-check by politifact with textual rating 'false'.
$\texttt{DecBot}_{\texttt{Article}}$	Article "Part 1 of CNN Democratic presidential debate" seems *not credible* based on its least credible sentence. (explanation for sentence CR omitted)
$\texttt{DecBot}_{\texttt{SocMedia}}$	Sentence `Absolutely fantastic, there is know difference between the two facist socialist powers of todays EU in Brussels, and the yesteryears of Nazi Germany` in tweet agrees with: `'You see the Nazi platform from the early 1930s ... look at it compared to the (Democratic Party) platform of today, you're saying, 'Man, those things are awfully similar.''` that seems *not credible* based on a fact-check by politifact with textual claim-review rating 'false'"

Finally, `coinform250`[15] is a dataset of 250 annotated tweets. The tweets and original labels were first collected by parsing and normalising `ClaimReviews` from datacommons and scraping fact-checker sites using the MisinfoMe data collector [9,10]. Note that acred's collection system is **not** based on MisinfoMe[16]. The original fact-checker labels were mapped onto six labels (see table 2) by 7 human raters achieving a Fleiss κ score of 0.52 (moderate agreement). The fine-grained labels make this a challenging but realistic dataset.

For each dataset our prediction procedure consisted in steps to (i) read samples, (ii) convert them to the appropriate `schema.org` data items (`Sentence`, `Article` or `SocialMediaPost`), (iii) request a review from the approriate acred CR bot via its REST API; (iv) map the produced CR onto the dataset labels and (v) optionally store the generated graph of CRs. For `clef18` we set $t = 0.75$, so that $r >= t$ has label `TRUE`, $r <= -0.75$ has label `FALSE` and anything in between is `HALF-TRUE`. See table 2 for `coinform250` threshold definitions.

Table 2: Mapping of credibility `ratingValue` r and `confidence` c for coinform250.

label	r	c
credible	$r \geq 0.5$	$c > 0.7$
mostly credible	$0.5 > r \geq 0.25$	$c > 0.7$
uncertain	$0.25 > r \geq -0.25$	$c > 0.7$
mostly not credible	$-0.25 > r \geq -0.5$	$c > 0.7$
not credible	$-0.5 > r$	$c > 0.7$
not verifiable	any	$c \leq 0.7$

5.2 Results

On `clef18`, `acred` establishes a new state-of-the-art result as shown in table 3, achieving 0.6835 in MAE, the official metric in the competition [11]. This result is noteworthy as, unlike the other systems, `acred` did not use the training set of `clef18` at all to finetune the underlying models. With `acred`$^+$, we further improved our results achieving 0.6475 MAE.

On FakeNewsNet, `acred`$^+$ obtained state of the art results and `acred` obtained competitive results in line with strong baseline systems reported in the original paper [16], shown in table 4. We only consider as baselines systems which only use the article content, since acred does not use credibility reviews based on social context

Table 4: Results on FakeNewsNet Politifact compared to baselines that only use article content.

System	Accuracy	Precision	Recall	F1
`acred`	0.586	0.499	**0.823**	0.622
`acred`$^+$	**0.716**	0.674	0.601	**0.713**
CNN	0.629	**0.807**	0.456	0.583
SAF/S	0.654	0.600	0.789	0.681

[15] https://github.com/co-inform/Datasets

[16] acred's data collector is used to build the `ClaimReview` database described in Sect. 4; it does not store the `itemReviewed` URL values; only the `claimReviewed` strings.

Table 3: Results on `clef18` English test dataset compared to baselines. The bottom rows shows results on the English training set.

system	MAE	Macro MAE	Acc	Macro F1	Macro AvgR
`acred`	0.6835	0.6990	**0.4676**	**0.4247**	0.4367
`acred`$^+$	**0.6475**	**0.6052**	0.3813	0.3741	0.4202
Copenhagen[19]	0.7050	0.6746	0.4317	0.4008	**0.4502**
random[11]	0.8345	0.8139	0.3597	0.3569	0.3589
`acred` "training"	0.6341	0.7092	0.4878	0.4254	0.4286
`acred`$^+$ "training"	0.6585	0.6386	0.4024	0.3943	0.4020

yet. Note that baselines used 80% of the data for training and 20% for testing, while we used the full dataset for testing.

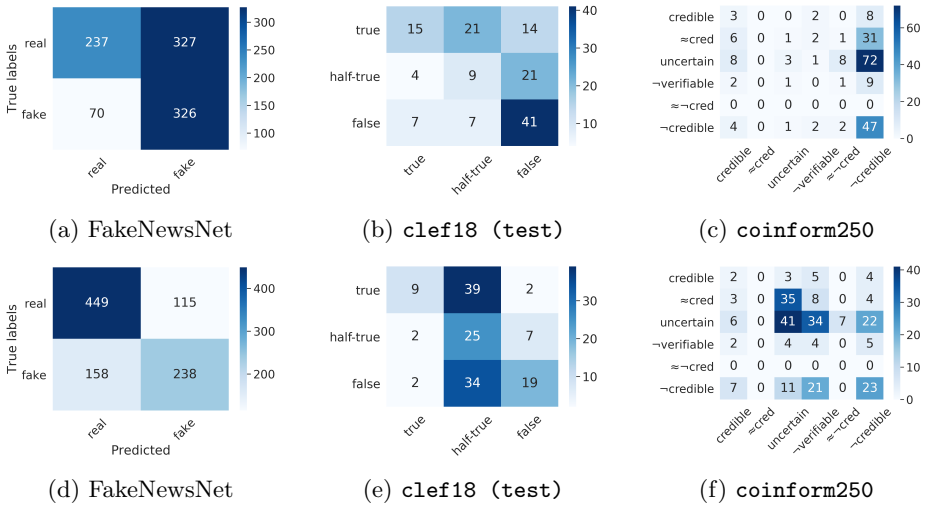
We performed a manual error analysis on the `acred` results for FakeNewsNet[17]:

- 29 errors of fake news predicted as highly credible ($r \geq 0.5$): 16 cases (55%) were due to acred finding pre-crawled sentence matches in that appeared in `snopes.com`, but not `ClaimReview`s for that article. A further 7 cases (24%) were due to finding `unrelated` sentences and using the `WebSiteCR` where those sentences appeared, while being over-confident about those credibilities.
- 41 errors of fake news predicted with low-confidence ($c <= 0.7$). 30 of these cases (73%) are due to the FakeNewsNet crawler as it fails to retrieve valid content for the articles: GDPR or site-for-sale messages instead of the original article content. In these cases, acred is correct in having low-confidence credibility ratings. In the remaining 27% of cases, acred indeed failed to find evidence to decide on whether the article was credible or not.
- 264 real articles were rated as being highly not credible ($r < -0.5$). This is by far the largest source of errors. We manually analysed 26 of these, chosen at random. In 13 cases (50%), the real stance should be *unrelated*, but is predicted as *discussed* or even *agrees*; often the sentences are indeed about a closely related topics, but still about unrelated entities or events. In a further 7 cases (27%) the stance is correctly predicted to be *unrelated*, but the confidence still passess the threshold. Hence 77% of these errors are due to incorrect or overconfident linking by the $\texttt{LinkBot}^{\texttt{SemSim}}_{\texttt{Sentence}}$.
- 48 real articles were rated as being somewhat not credible ($-0.25 < r \leq 0.25$), in a majority of these cases, the r was obtained from $\texttt{LinkBot}^{\texttt{PreCrawled}}_{\texttt{Sentence}}$ rather than from $\texttt{LinkBot}^{\texttt{SemSim}}_{\texttt{Sentence}}$

Finally, for the `coinform250` dataset, `acred`$^+$ obtains 0.279 accuracy which is well above a baseline of random predictions, which obtains 0.167 accuracy. The confusion matrix shown in figure 3f shows that the performance is in line with that shown for FakeNewsNet (fig.3d) and `clef18` (fig.3e). It also shows

[17] As stated above, we used the results of this analysis to inform the changes implemented in `acred`$^+$

that `acred` tends to be overconfident in its predictions, while `acred`$^+$ is more cautious.



(a) FakeNewsNet (b) `clef18 (test)` (c) `coinform250`

(d) FakeNewsNet (e) `clef18 (test)` (f) `coinform250`

Fig. 3: Confusion matrices for `acred` (top) and `acred`$^+$ (bottom row) in evaluation datasets. We use ≈ for *mostly* and ¬ for *not* in the `coinform250` labels.

5.3 Discussion

The Good Our approach obtains competitive results in challenging datasets; these results are especially impressive when you take into account that we do not train or fine-tune our underlying models on these datasets. With `acred`$^+$, we also showed we can substantially improve results by performing some simple optimization of aggregation functions; however doing this in a systematic manner is not in the scope of this paper where we are focusing on validating the LCR design. Since the results were consistent across the different datasets, this shows that our network of bots have certain domain independence and validate our design for composable CR bots as our lookup and linking bots can successfully be reused by the high-level decomposing bots. We think this is largely due to our choice of generic deep-learning models for linguistic tasks like semantic similarity and stance detection where fairly large datasets are available.

We obtain state of the art results in all datasets. This shows that our approach excels at identifying misinformation, which is arguably the primary task of these kinds of systems. Furthermore, in many cases, our system correctly matches the misinforming sentence to a previously fact-checked claim (and article). Even when no exact match is found, often the misinforming sentence is linked to a similar claim that was previously reviewed and that is indicative of a recurring

misinforming narrative. Such cases can still be useful for end-users to improve their awareness of such narratives.

The Bad The `acred` implementation is overly sceptical, which resulted in poor precision for data items which are (mostly) accurate or not verifiable. This is an important type of error preventing real-world use of this technology. As prevalent as misinformation is, it still only represents a fraction of the web content and we expected such errors to undermine confidence in automated systems. The presented error analysis shows that this is largely due to (i) errors in the stance-detection module or (ii) incorrect weighting of predicted stance and semantic similarity. The former was surprising as the stance prediction model obtained 92% accuracy on FNC-1. This seems to be due to the fact that FNC-1 is highly unbalanced, hence errors in under-represented classes are not reflected in the overall accuracy. Note that we addressed this issue to a certain extent with `acred`$^+$, which essentially `compensates` for errors in the underlying semantic similarity and stance prediction modules.

The poor precision on real news is especially apparent in FakeNewsNet and `coinform250`. We think this is due to our naive implementation of our pre-crawled database of sentences extracted from websites. First, the relevant sentence extractor does not ensure that the sentence is a factual claim, therefore introducing significant noise. This suggests that the system can benefit from adding a check-worthiness filter to address this issue. The second source of noise is our selection of pre-crawled article sources which did not seem to provide relevant matches in most cases. We expect that a larger and more balanced database of pre-crawled sentences, coming from a wider range of sources, should provide a better pool of credibility signals and help to improve accuracy.

The Dubious Fact-checking is a recent phenomenon and publishing fact-checked claims as structured data even more so. It is also a laborious process that requires specialist skills. As a result, the pool of machine-readable high-quality reviewed data items is relatively small. Most of the datasets being used to build and evaluate automated misinformation detection systems are therefore ultimately based on this same pool of reviews. A percentage of our results may be based on the fact that we have exact matches in our database of ClaimReviews. On manual inspection, this does not appear to occur very often; we estimate low, single digit, percentage of cases.

Although we did a systematic error analysis, presented in the previous section, we have not yet done a systematic success analysis. Cursory inspection of the successful cases shows that in some cases, we predicted the label correctly, but the explanation itself is incorrect; this tends to happen when a sentence is matched but is deemed to be *unrelated* to a matched sentence. Note that other systems based on machine learning may suffer of the same issue, but unlike our approach they behave as black boxes making it difficult to determine whether the system correctly identified misinforming content for the wrong reasons.

6 Conclusion and Future Work

In this paper we proposed a simple data model and architecture for using semantic technologies (linked data) to implement composable bots which build a graph of Credibility Reviews for web content. We showed that `schema.org` provides most of the building blocks for expressing the necessary linked data, with some crucial extensions. We implemented a basic fact-checking system for sentences, social media posts and long-form web articles using the proposed LCR architecture and validated the approach on various datasets. Despite not using the training sets of the datasets, our implementations obtained state of the art results on the `clef18` and FakeNewsNet datasets.

Our experiments have demonstrated the capabilities and added value of our approach such as human-readable explanations and machine aided navigation of provenance paths to aid in error analysis and pinpointing of sources of errors. We also identified promising areas for improvement and further research. We plan to (i) further improve our stance detection model by fine-tuning and testing on additional datasets[15]; (ii) perform an ablation test on an improved version of acred to understand the impact of including or omitting certain bots and datasets; (iii) perform crowdsourcing to evaluate both the understandability, usefulness and accuracy of the generated explanations. Beside our own plans, it is clear that a single organization or team cannot tackle all the issues which need to be resolved to achieve high-accuracy credibility analysis of web content. This is exactly why we propose the Linked Credibility Review, which should enable the distributed collaboration of fact-checkers, deep-learning service developers, database curators, journalists and citizens in building an ecosystem where more advanced, multi-perspective and accurate credibility analyses are possible.

Acknowledgements Work supported by the European Comission under grant 770302 – Co-Inform – as part of the Horizon 2020 research and innovation programme. Thanks to Co-inform members for discussions which helped shape this research and in particular to Martino Mensio for his work on MisInfoMe. Also thanks to Flavio Merenda and Olga Salas for their help implementing parts of the pipeline.

References

1. Babakar, M., Moy, W.: The State of Automated Factchecking. Tech. rep. (2016)
2. Boland, K., Fafalios, P., Tchechmedjiev, A.: Modeling and Contextualizing Claims. In: 2nd International Workshop on Contextualised Knowledge Graphs (2019)
3. Cazalens, S., Lamarre, P., Leblay, J., Manolescu, I., Tannier, X.: A Content Management Perspective on Fact-Checking. In: The Web Conference (2018)
4. 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)
5. Guha, R.V., Brickley, D., Macbeth, S.: Schema. org: evolution of structured data on the web. Communications of the ACM **59**(2), 44–51 (2016)

6. 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)
7. 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)
8. Marwick, A., Lewis, R.: Media manipulation and disinformation online. New York: Data & Society Research Institute (2017)
9. Mensio, M., Alani, H.: MisinfoMe: Who's Interacting with Misinformation? In: 18th International Semantic Web Conference: Posters & Demonstrations (2019)
10. Mensio, M., Alani, H.: News Source Credibility in the Eyes of Different Assessors. In: Conference for Truth and Trust Online. In Press (2019)
11. Nakov, P., Barrón-Cedeño, A., Suwaileh, R., Arquez, L.M.., Zaghouani, W., Atanasova, P., Kyuchukov, S., Da, G., Martino, S.: Overview of the CLEF-2018 CheckThat! Lab on Automatic Identification and Verification of Political Claims. In: International Conference of the Cross-Language Evaluation Forum for European Languages. pp. 372—-387 (2018)
12. Papadopoulos, S., Bontcheva, K., Jaho, E., Lupu, M., Castillo, C.: Overview of the special issue on trust and veracity of information in social media. ACM Transactions on Information Systems (TOIS) **34**(3), 1–5 (2016)
13. Pérez-Rosas, V., Kleinberg, B., Lefevre, A., Mihalcea, R.: Automatic Detection of Fake News. In: COLING (2018)
14. Pomerleau, D., Rao, D.: The fake news challenge: Exploring how artificial intelligence technologies could be leveraged to combat fake news (2017)
15. Schiller, B., Daxenberger, J., Gurevych, I.: Stance Detection Benchmark: How Robust Is Your Stance Detection? (jan 2020)
16. 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)
17. Tchechmedjiev, A., Fafalios, P., Boland, K., Gasquet, M., Zloch, M., Zapilko, B., Dietze, S., Todorov, K., Zloch, M.: ClaimsKG: A Knowledge Graph of Fact-Checked Claims. International Semantic Web Conference pp. 309–324 (2019)
18. Thorne, J., Vlachos, A., Cocarascu, O., Christodoulopoulos, C., Mittal, A.: The FEVER 2.0 Shared Task. In: Proceedings of the Second Workshop on Fact Extraction and VERification (FEVER). pp. 1–6 (2019)
19. Wang, D., Simonsen, J.G., Larsen, B., Lioma, C.: The Copenhagen Team Participation in the Factuality Task of the Competition of Automatic Identification and Verification of Claims in Political Debates of the CLEF-2018 Fact Checking Lab. CLEF (Working Notes) **2125** (2018)
20. Zubiaga, A., Aker, A., Bontcheva, K., Liakata, M., Procter, R.: Detection and resolution of rumours in social media: A survey. ACM Computing Surveys **51**(2) (2018)