ClaimEval: Integrated and Flexible Framework for Claim Evaluation Using Credibility of Sources

Mehdi Samadi
Carnegie Mellon University
msamadi@cs.cmu.edu

Partha Talukdar
Indian Institute of Science
ppt@serc.iisc.in

Manuela Veloso
Carnegie Mellon University
veloso@cs.cmu.edu

Manuel Blum
Carnegie Mellon University
mblum@cs.cmu.edu

Abstract

The World Wide Web (WWW) has become a rapidly growing platform consisting of numerous sources which provide supporting or contradictory information about claims (e.g., “Chicken meat is healthy”). In order to decide whether a claim is true or false, one needs to analyze content of different sources of information on the Web, measure credibility of information sources, and aggregate all these information. This is a tedious process and the Web search engines address only part of the overall problem, viz., producing only a list of relevant sources. In this paper, we present ClaimEval, a novel and integrated approach which gives a set of claims to validate, extracts a set of pro and con arguments from the Web information sources, and jointly estimates credibility of sources and correctness of claims. ClaimEval uses Probabilistic Soft Logic (PSL), resulting in a flexible and principled framework which makes it easy to state and incorporate different forms of prior knowledge. Through extensive experiments on real-world datasets, we demonstrate ClaimEval’s capability in determining validity of a set of claims, resulting in improved accuracy compared to state-of-the-art baselines.

Introduction

The World Wide Web (WWW) and the Web search engines, such as Google, Bing, etc., that operate over millions of documents have made information readily available to everyone. Over the last few years, several large knowledge bases (KBs), such as Freebase, Yago, Google Knowledge Graph, etc., have also been developed which makes it possible to readily evaluate factoid claims (e.g., “Paris is the capital of France”). In spite of this democratization of information, evaluating correctness of non-factoid claims (e.g., “Turkey meat is healthy”), is still an open challenge. This is particularly challenging as two different webpages may contain conflicting evidences even related to a single claim. For example, while an animal rights website might not support meat-eating and thus term turkey meat as unhealthy, the website of a grocery store might claim otherwise. Additionally, a scientific paper focusing on this question might provide the most authoritative answer. So ideally, one would want to trust evidences contained in the credible source (the scientific paper) and ignore the other two. Hence, given a set of claims, one needs to identify relevant sources on the Web, extract supporting and contradictory evidences from those sources, estimate source credibility, and finally aggregate all those information to evaluate the given set of claims. This is a tedious and time consuming process, and current Web search engines only address the first aspect of this big challenge. This is a growing need for an integrated approach for automatic extraction of relevant evidences and sources, estimation of information source credibility and utilizing those credibility estimates in claim evaluation.

Moreover, estimating information source credibility may be subjective (Bhattacharya, Devinney, and Pillutla 1998;
1. We propose a novel and fully-integrated technique called ClaimEval. Input to ClaimEval is the prior credibility assessment knowledge, and a set of claims, where the truth values of a few of them is known. ClaimEval then evaluates the truth of a set of unlabeled claims by automatically crawling the relevant information from the Web, building the CA graph, calculating the credibility of sources, and incorporating the calculated credibility scores to validate the truth of the claims.

2. ClaimEval uses Probabilistic Soft Logic (PSL) (Kimmig et al. 2012; Broecheler, Mihalkova, and Getoor 2010), resulting in a flexible and principled framework for joint credibility estimation and claim evaluation. In contrast to prior approaches, ClaimEval has the following three properties: (a) ease of incorporation of prior knowledge, (b) guaranteed convergence, and (c) interpretable credibility scores. To the best of our knowledge, ClaimEval is the first such integrated system of its kind.

3. We present experimental results on real-world data demonstrating effectiveness of ClaimEval compared to the state-of-the-art approaches.

Related Work

Given a few labeled nodes and a graph where the edge weight represents the degree of similarity between the two connected nodes, Graph-based Semi-Supervised Learning (GSSL) algorithms classify the initially unlabeled nodes in the graph (Subramanya and Talukdar 2014). Since GSSL algorithms can handle only a specific edge type (i.e., node similarity), they are not applicable to the CA graph (Figure 1) and the setting of this paper as a CA graph consists of multiple edge types with differing semantics.

To overcome the limitations of GSSL algorithms, several fact-finding algorithms have been developed which consider a bipartite version of a graph similar to that of CA, and propagate information between nodes in a non-symmetric way. Algorithms such as Sums (Kleinberg 1999), TruthFinder (Yin, Han, and Yu 2007), Generalized-Investment (Pasternack and Roth 2011), Pooled-Investment (Pasternack and Roth 2010), Accu-Vote (Dong, Berti-Equille, and Srivastava 2009), Average.Log (Pasternack and Roth 2010), and 3-Estimates (Galland et al. 2010) have been developed that use different propagation update functions which specify how the information flows between the nodes in such a graph. Each of these fact-finding algorithms suffers from at least one of the following main disadvantages. First, the score assigned to claims are biased toward favoring potentially (non-credible) sources that assert many evidences. Second, the scores assigned to the nodes are not interpretable. Third, the convergence of these iterative algorithms are not guaranteed. Finally, and most importantly, incorporating additional prior-knowledge to guide how the credibility information should flow over the CA-like graph is not intuitive and easy. Prior assumptions usually are incorporated through a set of cumbersome and usually non-intuitive update functions.

Some of these limitations are partially addressed by some other previous work. Pasternack and Roth (Pasternack and Roth 2010) introduced a pipelined approach which takes beliefs output by a fact-finder as an input, and “corrects” those beliefs based on some prior-knowledge defined by the user. Unlike ClaimEval, (Pasternack and Roth 2010) cannot incorporate additional prior-knowledge such as how credibility should propagate over the graph. Additionally, and in contrast to ClaimEval, the semantics of belief scores in (Pasternack and Roth 2010) is specific to the external fact-finder that is iteratively used to assign scores to the nodes in the graph, and convergence of the algorithm is also not guaranteed. To address bias and interpretability limitations of fact-finders, probabilistic fact-finding approaches are introduced that model the joint probability of the sources and the claims (Pasternack and Roth 2013; Zhao et al. 2012; Wang et al. 2011). Although these approaches provide a
transient and interpretable model, the prior knowledge can be provided only in a node-centric manner. Incorporating more sophisticated types of prior-knowledge, such as adding different types of nodes or how information should propagate over the graph, requires non-trivial modeling changes. In contrast, ClaimEval offers a flexible framework which makes it possible to add such prior-knowledge using first-order logic rules and without requiring any changes in the model.

Moreover, the above approaches do not provide an integrated system that asserts the truthfulness of claims by extracting evidences from the unstructured sources of information on the Web. OpenEval (Samadi, Veloso, and Blum 2013) addresses this issue but it doesn’t take source credibility into account. Similarly, Defacto (Lehmann et al. 2012) only uses PageRank as an estimation for the credibility of source. In contrast, ClaimEval identifies relevant sources, extracts evidences from them, estimates source credibility and uses those credibility scores for improved claim evaluation, all in an integrated system.

The notion of trust has been also addressed in other areas. Reputation-based systems are based on the notion of transitive trust and are used to measure credibility of entities such as webpages (Page et al.), people (Levien et al. 1998), and peers in network (Kamvar, Schlosser, and Garcia-Molina 2003). In computational trust, variations of probabilistic logic frameworks have been used to present a formal definition of trust and trust propagation, which can potentially be used in artificial agents to make trust-based decisions (Marsh 1994; Manchala 1998; Jøsang, Marsh, and Pope 2006). For data integration, different approaches are built to integrate the most complete and accurate records from diverse set of sources where the data sources are extremely heterogeneous (Zhao et al. 2012; Dong and Srivastava 2013; Li et al. 2014). In crowdsourcing, various techniques have been developed to automatically validate the quality of crowd answers, by identifying the faulty workers (Venanzi, Rogers, and Jennings ; Richardson and Domingos ; Wang et al. 2013; Nguyen et al. 2015). In social networks, Probabilistic Soft Logic (PSL) is used to model the trust in the social interactions (Huang et al. 2012; 2013). The truthfulness of deep web data has been studied by (Li et al. 2013), which surprisingly shows a large amount of inconsistency between the data provided by different sources. We point out that, while relevant, none of these prior work directly address the problem of claim evaluation in its full complexity as considered in this paper.

Our Approach: ClaimEval

ClaimEval performs joint estimation of source credibility and claim evaluation in one single model using Probabilistic Soft Logic (PSL). ClaimEval consists of the following two steps: (i) Credibility Assessment (CA) Graph Construction (Section ), and (ii) Joint Source Credibility Estimation and Claim Evaluation (Section ). In the next two sections, we describe each one of these two steps in greater detail. Algorithm 1 summarizes the process that ClaimEval follows for evaluating a claim.

```
Algorithm 1 ClaimEval - Evaluating Correctness of a Claim

Input: \( \{U, L, r, K\} \) /* U is a set of unlabeled claims that should be evaluated, L is a set of labeled claims (label is either true or false), r is the category that U and L belong to, and K is a set of first-order logic rules defining the prior credibility knowledge. */

Output: \( \langle \text{label (True or False), confidence} \rangle \) for claims in U

1: \( W \leftarrow \text{Search the Web using Bing and extract webpages for claims in } L \cup U \)
2: \( \text{CAG} \leftarrow \text{Construct layers of CA graph by parsing content of webpages in } W \)
3: \( \text{Connect nodes between different layers of CAG} \)
4: \( \text{Label edges that are connecting Evidence and Claim layers, using the evidence classifier trained for category } r \)
5: \( \text{Label nodes in the Claim layer for which we know the truth label (i.e., claims that exist in } L) \)
6: \( R \leftarrow \text{For all the nodes and edges in CAG, instantiate variables of the rules that are defined in prior knowledge } K, \text{ Equation 4, and Equation 5} \)
7: \( \text{R} \leftarrow \text{Relax logical operators in } R \text{ using Lukasiewicz real-values operators} \)
8: \( \langle \text{label, confidence} \rangle \leftarrow \text{Convert } R \text{ to an optimization problem using Probabilistic Soft Logic (PSL) and find the label (True or False) and confidence value of claims in } U \)
```

Credibility Assessment (CA) Graph Construction

Let us consider the two claims: “Turkey is healthy” (C1) and “Chicken is healthy” (C2) as shown in Figure 1. In practice, in order to evaluate these claims, one would probably first try to identify sources (e.g., webpages) which are likely to contain evidence either in favor or against such claims. An evidence may be considered as a text snippet (e.g., a sentence or paragraph in a webpage) expressing opinion about the claim in a webpage. Overall, a claim is substantiated by one or more evidences which are contained in webpages, with webpages in turn contained within domains. We can put all this together in a multi-relational graph which we shall call a Credibility Assessment (CA) graph. A CA graph has four types of nodes: Domain, Webpage, Evidence, and Claim. Nodes are connected by the following types of edges: (1) HasWebPage(Domain, WebPage), (2) ContainsEvidence(WebPage, Evidence), (3) DoesNotSupportClaim(Evidence, Claim), and (4) SupportsClaim(Evidence, Claim). For example, in Figure 1, evidence node “All meats are the same...” (E1) DoesNotSupportClaim C1. Evidence E1 is found in webpage represented by webpage node W1. This particular webpage is from the peta.org domain which is represented by node D1 in the CA graph.

As an input, we assume that a set of categories of claims is given to our system. For each category, a set of true and false claims are provided (i.e., category instances). For example, healthy food is an example of a category, and \{apple, broccoli\} and \{mayonnaise, soda\} are, respectively, true and false claims provided for this category. For each category, we assume that a set of labeled and unlabeled claims are provided (e.g., a set of true and false claims). Our goal is to classify the set of unlabeled claims to either true or false classes, with a confidence value attached to the label.
Training Evidence Classifier: To build the CA graph, ClaimEval first learns an evidence classifier for each category of claims, which is later used to classifying evidences, for a given claim, as either SupportsClaim (pro) or DoesNotSupportClaim (con) classes. This step is performed only once during training for each category. The trained classifier is then used during the test time. To build the training data for the classifier, ClaimEval first iterates over all the claims for which the label is known. ClaimEval then converts each claim to a search query, where the search query is built from the claim and the name of the category. ClaimEval then searches the query on the Web (e.g., using Bing) and downloads the set of the highest ranked webpages. In each of the returned webpages, ClaimEval extracts all the paragraphs that contain the claim phrase. All the words in each paragraph is saved as an evidence. The extracted evidences get the same labels as the labels of claims. By assuming that the training data extracted from the search query are ground-truth, ClaimEval trains a classifier (e.g., SVM) for each category of claims. For the input to the classifier, each evidence is represented using standard bag-of-words model (bigrams and unigrams).

Constructing CA Graph: After training the evidence classifier, ClaimEval iterates over all the labeled and unlabeled claims, similar to the training process, and extracts a set of evidences for each claim. Using the trained evidence classifier, each of the extracted evidences is then assigned to either the pro or the con classes. The evidence layer in the CA graph is built using all the evidences extracted from the Web. Each evidence is connected to the claim for which it is extracted. The edge that connects an evidence to a claim is classified as SupportsClaim (solid green line in Figure 1), if the evidence is supporting the claim, and otherwise is classified as DoesNotSupportClaim (red dotted line). The webpage layer is constructed from the webpages that are returned by the search engine. Each webpage is connected to the set of evidences extracted from the webpage. For each webpage, we extract the domain name (e.g., whfoods.org) and create a corresponding node in the domain layer. Each domain node is connected to its webpages in the webpage layer.

Initially, all the nodes in the graph do not have any value, except for the nodes whose true labels (i.e., user assessments) are known. Each node in the CA graph takes a value in the range [0,1]. The semantic of the values assigned to the nodes varies across the layers in the graph. For example, the value of a node in the webpage layer is interpreted as the degree of the credibility of the webpage, and the value of a node in the claim layer is interpreted as the confidence in the truth of a claim. Edges in the graph can be seen as transforming the meaning of values of nodes between the layers.

Joint Source Credibility Estimation & Claim Evaluation

When computing the credibility scores in the CA graph, we propagate the credibility and the correctness information across the different layers based on some prior knowledge, which is defined as a set of rules. We first list a set of such rules that specify how the credibility inference should be performed across the CA graph. We later explain how these rules are incorporated in our credibility assessment model.

Prior Knowledge for Credibility Assessment

- Evidence ⇒ Claim: Inferring correctness of a claim based on the credibility of an evidence:
  - EC1: Evidence is credible & evidence supports claim ⇒ claim is true.
  - EC2: Evidence is credible & evidence doesn’t support claim ⇒ claim is false.
  - EC3: Evidence is not credible, then the evidence has no effect on the correctness of the claim.

- Claim ⇒ Evidence: Inferring credibility of an evidence based on the correctness of a claim:
  - CE1: Claim is true & evidence supports claim ⇒ evidence is credible.
  - CE2: Claim is true & evidence doesn’t support claim ⇒ evidence is not credible.
  - CE3: Claim is false & evidence supports claim ⇒ evidence is not credible.
  - CE4: Claim is false & evidence doesn’t support claim ⇒ evidence is credible.

- Webpage ⇔ Evidence: Inferring credibility of an evidence based on the credibility of a webpage, and vice versa:
  - WE1: Webpage is credible ⇔ evidence is credible.
  - WE2: Webpage is not credible ⇔ evidence is not credible.

- Domain ⇔ Webpage: Inferring credibility of a webpage from the credibility of a domain, and vice versa:
  - DW1: Domain is credible ⇔ webpage is credible.
  - DW2: Domain is not credible ⇔ webpage is not credible.

Encoding Prior Knowledge using First-Order Logic

We use first-order logic (FOL) to formally define a set of rules, based on the prior knowledge that we defined in the previous section. Each layer of the CA graph is represented by a logical predicate, and each node in the graph is an instance of the predicate. For example, the predicate Domain(x) is used to define nodes in the domain layer. In this case, predicate instance Domain(peta.org) has value 1, since peta.org is a domain in the domains layer, otherwise it would have taken the value 0. Similarly, predicates Webpage(x), Evidence(x), and Claim(x) are defined to represent nodes in the other layers of the graph.

In addition to the predicates that represent different types of the nodes in the graph, we use the following predicates to define edges: HasWebpage(Domain, Webpage), ContainsEvidence(Webpage, Evidence), SupportsClaim(Evidence, Claim), DoesNotSupportClaim(Evidence, Claim). Values of the two predicates connecting an Evidence with a Claim is computed by the evidence classifier as described in the previous section.

We represent each of the rules that are defined as part of the prior knowledge in first-order logic. For example, the EC1 prior knowledge rule from above may be represented using the following rule:

\[ \text{Evidence}(e) \land \text{Claim}(c) \land \text{Credible}(e) \land \text{SupportsClaim}(e, c) \Rightarrow \text{Correct}(c) \]
In this case, if the evidence is credible (Credible(e) has value ‘true’ or 1.0), e supports claim c (i.e., SupportsClaim(e, c) = 1), then c is labeled true (i.e., Correct(c) = 1). Other rules are defined similarly.

Although the logical rules precisely define the flow of information in the graph, their applicability is limited on real world examples, as the nodes in the graph can only take values 1 (true) or 0 (false). For example, assume that a claim c is supported by two evidences a and b, both of which are credible. If a is in favor of c and b is against c, then we can neither say c is true nor that c is false. Ideally, the value of claim c is between 0 and 1 depending on the relative credibility of evidences a and b. In the next section, we explain how to address this issue.

**Going Beyond Binary Logical Operators using Probabilistic Soft Logic (PSL)** Probabilistic Soft Logic (PSL) is a general purpose logical system that uses First-Order Logic (FOL) as its underlying logical language. We overview the PSL technique briefly in this section; for a more detailed exposition of PSL, we refer the reader to (Bach et al. 2012; Broecheler, Mihalkova, and Getoor 2010; Kimmig et al. 2012).

In PSL, the value of each ground atom is relaxed to take a soft truth-value in the interval [0,1], where 0 is interpreted as absolute false and 1.0 as absolute true. The soft-truth assignment allows us to define the degree of correctness and credibility for the nodes in the graph. For example, if we have two sources s₁ and s₂, where Credible(s₁) > Credible(s₂), then we can infer that s₁ is more credible compared to s₂.

To handle the continuous truth values in the variables, PSL relaxes conjunction, disjunction, and negation logical operators by using Lukasiewicz real-valued logic operators, which are defined as follows:

\[
\begin{align*}
    a \land b &= \max\{0, a + b - 1\} \\
    a \lor b &= \max\{1, a + b\} \\
    \neg a &= 1 - a
\end{align*}
\]

Lukasiewicz t-norm operators, compared to other non-classical logic operators such as product t-norm, are suitable for our credibility assessment model since they linearly combine the values that they take. For example, consider this rule:

\[
\text{Credible(e)} \land \text{SupportsClaim(e, c)} \Rightarrow \text{Correct(c)}
\]

Using the Lukasiewicz disjunction operation, we can write the body of this rule as:

\[
\max\{0, \text{Credible(e)} + \text{SupportsClaim(e, c)} - 1\}
\]

which evaluates to true when the resulting value is greater than certain threshold, say 0.5. We can roughly interpret it as: claim c is correct only when e is credible and supports claim c, each at least by a degree of 0.5 (sum should be greater than 1.5).

Given a set of rules, we first instantiate all the variables in the rules with respect to the CA graph that is constructed. Given these instantiated rules, our goal is to find values of the different nodes in the graph such that the total number of satisfied rules is maximized. For solving this optimization problem, PSL defines the satisfaction distance for each rule in the domain. Given a ground rule r, such as \( r := P \Rightarrow Q \), r is satisfied if and only if \( \hat{V}(Q) \geq \hat{V}(P) \), where \( \hat{V}(X) \) is defined as the value of the ground logical expression X using Lukasiewicz operators (i.e., \( Q \) is at least as truthful as \( P \)). The rule’s distance to satisfaction (denoted by \( d(r) \)) measures the satisfaction degree of each rule r := \( P \Rightarrow Q \):

\[
d(r) = \max\{0, \hat{V}(P) - \hat{V}(Q)\}
\]

Given the distance function in Equation 2 and a set of ground rules, PSL finds values for all the ground predicate instances in order to minimize the total satisfaction degrees of all the ground rules. To do this, assume that \( I \) is the assignment of values to predicate instances, and \( d_I(r) \) is the satisfaction degree of rule r given assignment I. Thus, given the set of ground rules \( R \), the optimal assignment \( I^* \) may be obtained as follows,

\[
I^* \leftarrow \arg\max_I \frac{1}{Z} \exp[- \sum_{r \in R} \lambda_r (d_I(r))^2]
\]

where \( Z \) is the normalization factor, and \( \lambda_r \) is the weight for each ground rule \( r \). Most Probable Explanation (MPE) inference algorithm may be used to optimize Equation 3.

When optimizing Equation 3, we allow only the values of Credible and Correct predicate instances to change, and the values of the rest of predicate instances remain fixed. Also, in order to make sure that the labeled claims will preserve their values during the optimization (i.e., constraints), we define the following rules:

\[
\forall \text{ labeled claims } c, \text{ PosClaim}(c) \rightarrow \text{Correct}(c)
\]

\[
\forall \text{ labeled claims } c, \text{ NegClaim}(c) \rightarrow \neg \text{Correct}(c)
\]

Predicates PosClaims and NegClaims respectively define the positive and negative set of labeled claims. Ideally, the above rules should have higher weight compared to the other rules defined as part of the prior knowledge, in order to make sure that during the optimization, the correct labels are assigned to the labeled data. The weight of rules can be tuned manually or can be learnt by using using the Maximum Likelihood Estimation (MLE) (Broecheler, Mihalkova, and Getoor 2010).

**Experimental Evaluation**

**Setup:** ClaimEval is tested on nine different sets of categories. Table 1 lists all the categories in the leftmost column. For each category, about 40 seed examples are used to train a SVM-based evidence classifier (used to annotate evidences with pro or con labels), and about 40 examples are used as test data. All the data sets, including the exact number of training examples for each category, and the methodology used to obtain the train and test data for each category are available at: www.ClaimEval.com. To train the evidence classifier, we use the top ten pages returned by Bing, and all the evidences extracted from the returned search pages. 0.5 is used as the confidence threshold for all experiments.

**Baselines:** We compare the accuracy of ClaimEval to the following Fact-Finding approaches: Majority Vote
Table 1: The accuracy of Majority Vote (MV), Generalized Sums (GS), TruthFinder (TF), Average-Log (AL), Generalized Investment (GI), Pooled-Investment (PI), and ClaimEval (CE) techniques in predicting the truth values for different categories of claims. Maximum value of each row shown in bold. ClaimEval, the proposed system, achieves the best overall performance.

<table>
<thead>
<tr>
<th>Category</th>
<th>MV</th>
<th>GS</th>
<th>TF</th>
<th>AL</th>
<th>GI</th>
<th>PI</th>
<th>CE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Healthy Food</td>
<td>0.89</td>
<td>0.89</td>
<td>0.69</td>
<td>0.86</td>
<td>0.89</td>
<td>0.89</td>
<td>0.91</td>
</tr>
<tr>
<td>Company with Stock Growth</td>
<td>0.62</td>
<td>0.65</td>
<td>0.65</td>
<td>0.62</td>
<td>0.62</td>
<td>0.60</td>
<td>0.72</td>
</tr>
<tr>
<td>High Ranked Universities</td>
<td>0.80</td>
<td>0.82</td>
<td>0.73</td>
<td>0.85</td>
<td>0.80</td>
<td>0.80</td>
<td>0.85</td>
</tr>
<tr>
<td>Top CS Journals</td>
<td>0.74</td>
<td>0.71</td>
<td>0.82</td>
<td>0.67</td>
<td>0.71</td>
<td>0.71</td>
<td>0.79</td>
</tr>
<tr>
<td>Top CS Conferences</td>
<td>0.53</td>
<td>0.55</td>
<td>0.58</td>
<td>0.57</td>
<td>0.53</td>
<td>0.53</td>
<td>0.68</td>
</tr>
<tr>
<td>High GDP Growth</td>
<td>0.60</td>
<td>0.70</td>
<td>0.50</td>
<td>0.70</td>
<td>0.6</td>
<td>0.50</td>
<td>0.60</td>
</tr>
<tr>
<td>High HDI Growth</td>
<td>0.81</td>
<td>0.53</td>
<td>0.65</td>
<td>0.63</td>
<td>0.81</td>
<td>0.86</td>
<td>0.76</td>
</tr>
<tr>
<td>High Crime Rate Cities</td>
<td>0.67</td>
<td>0.63</td>
<td>0.67</td>
<td>0.60</td>
<td>0.80</td>
<td>0.80</td>
<td>0.80</td>
</tr>
<tr>
<td>Top Soccer Club Teams</td>
<td>0.65</td>
<td>0.62</td>
<td>0.65</td>
<td>0.65</td>
<td>0.69</td>
<td>0.62</td>
<td>0.69</td>
</tr>
<tr>
<td>Average</td>
<td>0.71</td>
<td>0.68</td>
<td>0.66</td>
<td>0.69</td>
<td>0.72</td>
<td>0.71</td>
<td>0.76</td>
</tr>
</tbody>
</table>

Main Result: Table 1 shows the accuracy of baselines and ClaimEval for nine different categories. The experiments are obtained when 60 evidences are randomly extracted for each claim from the first 10 pages returned by Bing.

From Table 1 we observe that the performance of fact-finding algorithms may be category-specific, which is consistent with similar observations in (Pasternack and Roth 2011). Overall, ClaimEval, the proposed system, achieves best performance in 6 out of the 9 categories, and on average outperforming all other state-of-the-art baselines.

Change in Performance with Increasing Evidence Size: Figure 2 shows accuracy of different methods when increasing number of evidences is available. ClaimEval, the proposed system (top-most plot), is best able to exploit additional evidence achieving best overall performance.

Qualitative Example: As described in the introduction, one of the main disadvantages of most of the fact-finding algorithms (including the baselines considered in this paper) is their bias towards increased credibility for sources that assert many claims.

Figure 3 shows an example of a CA graph, where a domain node 1 (through webpage node 3) is asserting overwhelmingly many evidences against claim node 93 (negative evidence is marked by dotted red line in the figure, as in Figure 1). Claim node 92 with concentric circles is known to be correct a-priori. Correctness of the other claim node 93 needs to be evaluated. Based on these input information, judgments of a human annotator is shown on top of each node. Most of the fact-finding algorithms (including MV, AL, and GS) are biased towards favoring domain nodes with many evidences. In this example, these algorithms overestimate the credibility of nodes 1 and 3, and incorrectly classify claim 93 as incorrect (as all evidences from source node 3 oppose claim 93). In contrast, ClaimEval is able to infer that nodes 1 and 3 have low credibility since they provide contradictory information regarding claim 92. ClaimEval then uses this reduced credibility to correctly classify claim 93 as true. This examples provides qualitative evidence of how ClaimEval is able to overcome bias of many existing fact-finding algorithms.
Conclusion

In this paper, we proposed ClaimEval, an integrated, flexible, and principled system for claim evaluation. In contrast to previous approaches for claim evaluation, given a set of claims, ClaimEval identifies a set of relevant sources and evidences within them which might support or refute the claims, estimates credibility of those sources, and uses those credibility estimates to evaluate correctness of the claims – all in a single integrated system. ClaimEval uses Probabilistic Soft Logic (PSL) to flexibly incorporate various types of prior-knowledge from the user. Through extensive experiments on real-world datasets, we demonstrated ClaimEval’s effectiveness over other state-of-the-art baselines. As part of future work, we hope to exploit the flexibility offered by ClaimEval and incorporate other types of prior-knowledge and user preferences and evaluate their effect on claim evaluation.

Acknowledgments

The authors would like to thank Uday Saini and Ankit Sharma for carefully reading a draft of this paper. This work is supported in part by research grants from Google, Accenture, Pradeep Sindhu Fellowship, and the National Science Foundation (under Grants CCF-1101668 and CCR-0122581).

References


Li, X.; Dong, X. L.; Lyons, K.; Meng, W.; and Srivastava, D. 2013. Truth finding on the deep web: is the problem solved? In VLDB.

Li, Q.; Li, Y.; Gao, J.; Zhao, B.; Fan, W.; and Han, J. 2014. Resolving conflicts in heterogeneous data by truth discovery and source reliability estimation. In SIGMOD.


Pasternack, J., and Roth, D. 2010. Knowing what to believe (when you already know something). In COLING.


Samadi, M.; Veloso, M.; and Blum, M. 2013. Openeval: Web information query evaluation. In AAAI.


Venanzi, M.; Rogers, A.; and Jennings, N. R. Trust-based fusion of untrustworthy information in crowdsourcing applications. In AAMAS 2013.


Yin, X.; Han, J.; and Yu, P. S. 2007. Truth discovery with multiple conflicting information providers on the web. In KDD.

Zhao, B.; Rubinstein, B. I. P.; Gemmell, J.; and Han, J. 2012. A bayesian approach to discovering truth from conflicting sources for data integration. Proc. VLDB Endow. 5(6).