Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence

Making Better Informed Trust Decisions with Generalized Fact-Finding

Jeff Pasternack **Dan Roth**
University of Illinois, Urbana-Champaign
{jpaster2, danr}@uiuc.edu

Abstract

Information retrieval may suggest a document, and information extraction may tell us what it says, but which information sources do we trust and which assertions do we believe when different authors make conflicting claims? Trust algorithms known as *fact-finders* attempt to answer these questions, but consider only which source makes which claim, ignoring a wealth of background knowledge and contextual detail such as the uncertainty in the information extraction of claims from documents, attributes of the sources, the degree of similarity among claims, and the degree of certainty expressed by the sources. We introduce a new, *generalized* fact-finding framework able to incorporate this additional information into the fact-finding process. Experiments using several state-of-the-art fact-finding algorithms demonstrate that generalized fact-finders achieve significantly better performance than their original variants on both semi-synthetic and real-world problems.

1 Introduction

The Information Age has created an explosion of data, but has also lowered the barriers of entry for producers of that data. Databases were once ledgers written by hand by a single person; today they can be vast stores of information agglomerated from a myriad of disparate sources, much of it of uncertain veracity. The mass media, formerly limited to newspapers and television programs held to journalistic standards, has expanded to include collaborative content such as blogs, wikis, and message boards. Documents covering nearly every topic abound on the Internet, but the motives of the authors and accuracy of the content are frequently unknown to the reader. Even personal communication has transformed from traceable and (relatively) costly mail and telephone services to cheap, anonymous email and VoIP. Regardless of whether the consumer is a human or an algorithm, the vast quantity of information now at hand is tempered by a newfound ignorance as to its quality.

Information *sources* each make one or more *claims*. Since sources often make claims that are contradictory ("Obama was born in Hawaii" and "Obama was born in Kenya"), which claims do we believe, and which sources do we trust? The simplest method, *voting*, simply chooses the claim made by the most sources. However, not all sources are equally trustworthy; consequently, a class of algorithms called *fact-finders* instead iteratively calculate the trustworthiness of each source given the belief in its claims, and the belief in each claim given the trustworthiness of its sources.

However, fact-finders can only consider "who claims what" in their trust decision, despite the plethora of additional background knowledge and contextual detail available. For example, information extraction may give us a *distribution* over possible claims made by a source instead of a simple binary prediction (the claim was certainly made or it certainly wasn't). Furthermore, we often face ambiguities in both a document's semantics (does "Barack Obama was born in Kenya" refer to President Obama or his father, Barack Obama, Sr.?) and its attribution (e.g. if "Barack Obama" is listed as the author). Alternatively, the sources themselves may not be entirely sure of their claim ("I'm 90% sure Obama was born in Hawaii"). We can also judge the similarity among competing claims to determine, for example, that a source claiming "Kenya" implicitly prefers neighboring "Uganda" over "Hawaii". Or we may know the group membership of a source (a source belonging to the Tea Party movement can hardly be considered impartial) and other background knowledge of its attributes, such as (depending on the type of source) age, educational attainment, number of employees, sophistication of a document's layout, and so on.

Building upon our preliminary work in [Pasternack and Roth, 2011], our principal contribution is a framework for incorporating such diverse, commonly-available information by generalizing fact-finding algorithms, allowing them to consider important new factors in their trust decisions. In doing so, we are able to achieve better results than were previously possible, as demonstrated by our experiments utilizing state-of-the-art fact-finding algorithms.

2 Related Work

Fact-finders consider a set of sources, each of which makes a set of claims. Often, sets of claims are *mutually exclusive* with one another (e.g. putative Obama birthplaces), and the goal of the fact-finder is to determine which of these alternatives is correct. They do this by iteratively calculating the trustworthiness of each source given the belief in the claims it makes, and the belief in each claim given the trustworthiness of the sources asserting it, in the same manner of [Kleinberg, 1999]'s Hubs and Authorities. TruthFinder [Yin *et al.*, 2008], for example, calculates the trustworthiness of a source as the mean belief in its claims. Further, [Pasternack and Roth, 2010] introduces the Investment, PooledInvestment, and Average·Log algorithms, and demonstrates that these perform markedly better than the previous state-of-the-art on a number of datasets.

Other fact-finders enhance this basic formula. AccuVote [Dong *et al.*, 2009] computes source dependence (where one source copies another) and gives greater credence to more "independent" sources. 3-Estimates [Galland *et al.*, 2010] estimates the "difficulty" of claims in its calculation, and correctly asserting a difficult claim (for which there is a high degree of disagreement) confers more trustworthiness for a source than asserting something that is "obvious". Finally, [Pasternack and Roth, 2010] presents a method for incorporating prior knowledge (especially common-sense knowledge) about the relationships among claims (such as "Obama was born in Hawaii ⇒ Obama is a U.S. citizen") into fact-finding to improve performance. These approaches are orthogonal to our work; e.g. we can generalize the 3-Estimates algorithm, and can apply common-sense to a generalized fact-finder just as we can to a standard one.

3 Fact-Finding Algorithms

Before we discuss generalized fact-finding, we describe the standard fact-finding algorithm. We have a set of sources, S, a set of claims C, the claims C_s asserted by each source $s \in S$, and the set of sources S_c asserting each claim $c \in C$. The sources and claims can be viewed as a bipartite graph, where an edge exists between each s and c if $c \in C_s$. In each iteration i, we estimate the trustworthiness $T^i(s)$ of each source s given $B^{i-1}(C_s)$, the belief in the claims it asserts, and estimates the belief $B^i(c)$ in each claim c given $T^i(S_c)$, the trustworthiness of the sources asserting it, continuing until convergence or a stop condition. An initial set of beliefs, $B^0(C)$, serve as priors for each algorithm; our experiments use those given by [Pasternack and Roth, 2010]. As a concrete example, consider the simple **Sums** fact-finder based on Hubs and Authorities [Kleinberg, 1999]; here $T^i(s) = \sum_{c \in C_s} B^{i-1}(c)$ and $B^i(c) = \sum_{s \in S_c} T^i(s)$, with $B^0(c) = 1$ for all $c \in C$.

The *mutual exclusion set* $M_c \subseteq C$ is the set of claims that are mutually exclusive to one another (e.g. Obama's birthplaces) to which c belongs; if c is not mutually exclusive to any other claims, $M_c = \{c\}$. For each mutual exclusion set M containing true claim $\bar{c}$, the goal of the fact-finder is to ensure $\operatorname{argmax}_{c \in M_c} B^f(c) = \bar{c}$ at the final iteration f; the reported accuracies in our experiments are thus the percentage of mutual exclusion sets we correctly predict over, discounting cases where this is trivial ($|M| = 1$) or no correct claim is present ($\bar{c} \notin M$).

Notice that a fact-finder can be specified with just three things: a trustworthiness function $T(s)$, a belief function $B(c)$, and the set of priors $B^0(C)$. We use a number of fact-finders in our experiments: the aforementioned Sums, 3-Estimates [Galland *et al.*, 2010], TruthFinder [Yin *et al.*, 2008], and Average · Log, Investment, and PooledInvestment [Pasternack and Roth, 2010].

4 Generalized Fact-Finding

The key technical idea behind generalized fact-finding is that we can quite elegantly encode the relevant background knowledge and contextual detail by replacing the bipartite graph of standard fact-finders with a new weighted k-partite graph, transitioning from binary assertions to weighted ones ("source s claims c with weight x") and adding additional "layers" of nodes to the graph to represent source groups and attributes. We then need only modify the fact-finding algorithms to function on this new graph.

4.1 Encoding Information in Weighted Assertions

Weighted assertions, where each source s asserts a claim c with weight $\omega(s, c) = [0, 1]$, allow us to incorporate a variety of factors into the model:

- Uncertainty in information extraction: we have a $[0, 1]$ probability that source s asserted claim c.
- Uncertainty of the source: a source may qualify his assertion ("I'm 90% certain that...")
- Similarity between claims: a source asserting one claim also implicitly asserts (to a degree) similar claims.
- Group membership: the other members of the groups to which a source belongs implicitly support (to a degree) his claims.

We separately calculate ω_u for uncertainty in information in extraction, ω_p for uncertainty expressed by the source, ω_σ for the source's implicit assertion of similar claims, and ω_g for a source's implicit assertion of claims made by the other members of the groups to which he belongs. These are orthogonal, allowing us to calculate the final assertion weight $\omega(s, c)$ as: $\omega_u(s, c) \times \omega_p(s, c) + \omega_\sigma(s, c) + \omega_g(s, c)$. Here, $\omega_u(s, c) \times \omega_p(s, c)$ can be seen as our expectation of the $[0, 1]$ belief the source s has in claim c given the possibility of an error in information extraction, while $\omega_\sigma(s, c)$ and $\omega_g(s, c)$ redistribute weight based on claim similarity and source group membership, respectively.

Uncertainty in Information Extraction

The information extractor may be uncertain whether an assertion occurs in a document due to intrinsic ambiguities in the document or error from the information extractor itself (e.g. an optical character recognition mistake, an unknown verb, etc.); in either case, the weight is given by the probability $\omega_u(s, c) = P(s \to c)$.

Uncertainty of the Source

Alternatively, the source himself may be unsure. This may be specific ("I am 60% certain that Obama was born in...") or vague ("I am pretty sure that..."); in the latter case, we assume that the information extractor will assign a numerical certainty for us, so that in either event we have $\omega_p(s, c) = P_s(c)$, where $P_s(c)$ is the estimate provided by source s of the probability of claim c.

Similarity Between Claims

Oftentimes a meaningful similarity function exists among the claims in a mutual exclusion set. For example, when comparing two possible birthdays for Obama, we can calculate their similarity as the inverse of the time between them, e.g. $|days(date1) - days(date2)|^{-1}$ (where $days$ measures the number of days relative to an arbitrary reference date). A source claiming $date1$ then also claims $date2$ with a weight proportional to this degree of similarity, the idea being that while $date2$ is not what he claimed, he will prefer it over other dates that are even *more* dissimilar. Given a $[0,1]$ similarity function $\sigma(c_1, c_2)$, we can calculate:

$$\omega_\sigma(s,c) = \sum_{d \in M_c, d \neq c} \omega_u(s,d)\omega_p(s,d)\sigma(c,d)$$

Notice that a self-consistent source will not assert multiple claims in mutual exclusion set M with $\sum_{c \in M} \omega_u(s,c)\omega_p(s,c) > 1$, and thus the addition of $\omega_\sigma(s,c)$ to $\omega(s,c)$ will never result in $\omega(s,c) > 1$; it is possible, however, that $\sum_{c \in M} \omega(s,c) > 1$ for a given source s. One way to avoid this is to redistribute weight rather than add it; we introduce the parameter α to control the degree of redistribution and obtain:

$$\omega_\sigma^\alpha(s,c) = \sum_{d \in M_c, d \neq c} \left(\frac{\alpha\omega_u(s,d)\omega_p(s,d)\sigma(c,d)}{\sum_{e \in M_d, e \neq d} \sigma(d,e)} \right) - \alpha\omega_u(s,c)\omega_p(s,c)$$

This function ensures that only a portion α of the source's expected belief in the claim, $\omega_u(s,c)\omega_p(s,c)$, is redistributed among other claims in M_c (proportional to their similarity with c), at a cost of $\alpha\omega_u(s,c)\omega_p(s,c)$.

[Yin *et al.*, 2008] previously used a form of additive similarity as "Implication" functions in TruthFinder; however, the our formalization generalizes this idea and allows us to apply it to other fact-finders as well.

Group Membership via Weighted Assertions

Oftentimes a source belongs to one or more groups; for example, a journalist may be a member of professional associations and an employee of one or more publishers. Our assumption is that these groups are *meaningful*, that is, sources belonging to the same group tend to have similar degrees of trustworthiness. A prestigious, well-known group (e.g. the group of administrators in Wikipedia) will presumably have more trustworthy members (in general) than a discredited group (e.g. the group of blocked Wikipedia editors). The approach discussed in this section encodes these groups using ω_g; a more flexible approach, discussed later, is to use additional "layers" of groups and attributes instead.

Let G_s be the set of groups to which a source s belongs. If a source s and source u are both members of the same group g, we interpret this as an implicit assertion by u in C_s, and by s in C_u—that is, group members mutually assert each others' claims to a degree. We use a redistribution parameter β such that the original weight of a member's assertion is split between the member (proportional to $1-\beta$) and the other members of the groups to which he belongs (proportional to β). This gives us:

$$\omega_g^\beta(s,c) = \beta \sum_{g \in G_s} \sum_{u \in g} \frac{\omega_u(u,c)\omega_p(u,c) + \omega_\sigma(u,c)}{|G_u| \cdot |G_s| \cdot \sum_{v \in g} |G_v|^{-1}} - \beta(\omega_u(s,c)\omega_p(s,c) + \omega_\sigma(s,c))$$

$\sum_{v \in g} |G_v|^{-1}$ in the denominator gives greater credence to "small" groups (where members belonging to many other groups weigh less heavily), with the intuition that smaller groups have more similar members. Note that in the worst case (where all sources belong to a single group and each assert a unique set of k claims) this can effectively create as many as $(k \cdot |S|)^2 - k \cdot |S|$ new assertions, with a corresponding increase in computational cost when running the fact-finder.

4.2 Rewriting Fact-Finders for Assertion Weights

After calculating the weight functions $\omega(s,c)$ for all $s \in S$ and $c \in C$, we need to rewrite each fact-finder's $T(s)$, $B(c)$ and $B^0(c)$ functions to use these weights in the generalized fact-finding process by qualifying previously "whole" assertions as "partial", weighted assertions. We start by redefining S_c as $\{s : s \in S, \omega(s,c) > 0\}$, and C_s as $\{c : c \in C, \omega(s,c) > 0\}$. The basic rewriting rules are:

- Replace $|S_c|$ with $\sum_{s \in S_c} \omega(s,c)$
- Replace $|C_s|$ with $\sum_{c \in C_s} \omega(s,c)$
- In $T^i(s)$, replace $B^{i-1}(c)$ with $\omega(s,c)B^{i-1}(c)$
- In $B^i(c)$, replace $T^i(s)$ with $\omega(s,c)T^i(s)$

These rules suffice for all the linear fact-finders we encountered; one, TruthFinder, is instead log-linear, so an exponent rather than a coefficient is applied, but such exceptions are straightforward. For brevity, we list only three of the rewritten fact-finders here as examples.

Generalized Sums (Hubs and Authorities)

$$T^i(s) = \sum_{c \in C_s} \omega(s,c)B^{i-1}(c) \quad B^i(c) = \sum_{s \in S_c} \omega(s,c)T^i(s)$$

Generalized Average·Log

Average·Log employs the same B function as Sums, so we provide only the trustworthiness function:

$$T^i(s) = \log\left(\sum_{c \in C_s} \omega(s,c)\right) \cdot \frac{\sum_{c \in C_s} \omega(s,c)B^{i-1}(c)}{\sum_{c \in C_s} \omega(s,c)}$$

Generalized Investment

The Investment algorithm requires sources to "invest" their trust uniformly in their claims; we rewrite this such that these investments are weighted by ω. As per [Pasternack and Roth, 2010], we used the same $\mathcal{G}(x) = x^{1.2}$ in our experiments.

$$T^i(s) = \sum_{c \in C_s} \frac{\omega(s,c)B^{i-1}(c)T^{i-1}(s)}{\sum_{c \in C_s} \omega(s,c) \cdot \sum_{r \in S_c} \frac{\omega(r,c)T^{i-1}(r)}{\sum_{b \in C_r} \omega(r,b)}}$$

$$B^i(c) = \mathcal{G}\left(\sum_{s \in S_c} \frac{\omega(s,c)T^i(s)}{\sum_{c \in C_s} \omega(s,c)}\right)$$

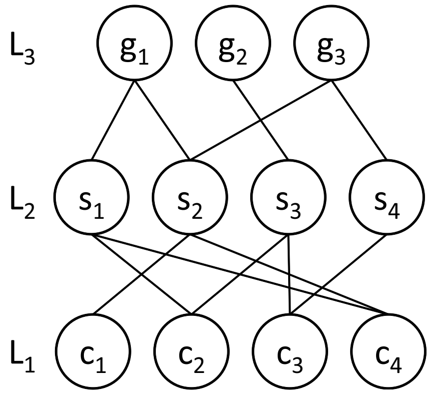

Figure 1: A fact-finding problem with a single group layer. Edges between sources and groups denote membership.

4.3 Groups and Attributes as Layers

Instead of using weighted assertions, we can add additional "layers" to represent groups and attributes directly. Each node in these layers will represent a group or attribute, with edges linking to its adjoining layers (either the sources or other groups/attributes), creating a k-partite graph (with $k > 3$ used to encode meta-groups and meta-attributes.) An standard fact-finder iteratively alternates between calculating the first layer (the claims) and the second layer (the sources), using the B and T functions, respectively. Now we replace these with generic "up" and "down" functions for each layer. For a k-partite graph with layers $L_{1...k}$, we define $U_j^i(L_j)$ over $j = 2...k$ and $D_j^i(L_j)$ over $j = 1...k-1$, with special cases $U_1^i(L_1) = D_1^{i-1}(L_1)$ and $D_k^i(L_k) = U_k^i(L_k)$. The U_j and D_j functions may differ for each j, or they may be the same over all layers. In each iteration i, we calculate the values $U_j^i(L_j)$ for layers $j = 2$ to k, and then calculate $D_j^i(L_j)$ for layers $j = k-1$ to 1. For example, to extend Sums to k layers, we calculate $U_j(e)$ and $D_j(e)$ as follows for $e \in L_j$:

$$U_j^i(e) = \sum_{f \in L_{j-1}} \omega(e,f) U_{j-1}^i(f)$$

$$D_j^i(e) = \sum_{f \in L_{j+1}} \omega(e,f) D_{j+1}^i(f)$$

Where $\omega(e,f) = \omega(f,e)$ is the edge weight between nodes e and f; if e or f is a group or attribute, $\omega(e,f)$ is 1 if e has attribute or group f or vice-versa, and 0 otherwise. In many cases, though, we may benefit from using an existing fact-finder over the claim and source layers, while using a different set of functions to mediate the interaction between the source and group/attribute layers. In particular, an information bottleneck often exists when calculating trustworthiness of a source in the "down phase", as it will be wholly dependent upon the trustworthiness of the groups to which it belongs: a source belonging to one overall-mediocre group may make many correct claims, but still be assigned a low trustworthiness score by the D function because of its group membership. This type of problem can be resolved by incorporating both the layer below *and* the layer above in each calculation; for example, for a given $D_j(e)$, we can define $\omega_{children} = \sum_{f \in L_{j-1}} \omega(e,f)$ and $D_j^{smooth}(e) = (1+\omega_{children})^{-1} D_j(e) + \omega_{children}(1+\omega_{children})^{-1} U_j(e)$, which returns a mixture of the value derived from e's ancestors, $D_j(e)$ and the value derived from e's descendants, $U_j(e)$, according to the (weighted) number of children e possesses, the intuition being that with more children $U_j(e)$ is more certain and should be weighted more highly, whereas with fewer children we should depend more upon our ancestors. We will use $D_j^{smooth}(e)$ in our experiments.

5 Experiments

5.1 Datasets

We use [Pasternack and Roth, 2010]'s Population dataset, extracted from Wikipedia infoboxes [Wu and Weld, 2007] (semi-structured tables with various fields within Wikipedia articles), and [Yin *et al.*, 2008]'s Books dataset, extracted from online bookstore websites. The Population dataset is a collection of 44,761 claims of the populations of cities in a particular year (e.g. triples such as (New York, 1400000, 2005)) from 171,171 sources ("editors", in Wikipedia parlance), with a test set of 308 true claims taken from census data (omitting the many cases where editors did not contest the population, or where all claims in Wikipedia were wrong). The Books dataset is a collection of 14,287 claims of the authorship of various books by 894 websites, where a website asserts that a person was an author of a book (e.g. (Bronte, "Jane Eyre")) explicitly by including them in the list of authors, or implicitly asserts a person was *not* an author (e.g. (¬Bronte, "Jane Eyre")) by omitting them from the list (when at least one other website lists that person as an author of the book—if nobody lists a person as an author, his non-authorship is not disputed and can be ignored). The test set is 605 true claims collected by examining the books' covers.

5.2 Tuned Assertion Certainty

A user modifying a field of interest in an infobox (e.g. the *population_total* field) is clearly asserting the corresponding claim ("population = x"), but what if he edits another part of the infobox, or somewhere else on the page? Did he also read and approve the fields containing the claims we are interested in, implicitly asserting them? [Pasternack and Roth, 2010] opted to simply consider only direct edits of a field containing a claim to be an assertion of that claim, but this ignores the large number of potential assertions that may be implicit in an editor's decision to *not* change the field.

This may be considered either uncertainty in information extraction (since we are not able to extract the author's true intent) or uncertainty of the part of the authors (an editor leaves a field unaltered because he believes it is "probably" true). In either case, we can weight the assertions to model this uncertainty in the generalized fact-finder. The information extractor provides a list of all edits and their type (editing the field of interest, another field in the infobox, or elsewhere in the document), and each type of edit implies a different certainty (a user editing another field in the infobox is more likely to have read and approved the neighboring field of interest than a user editing a different portion of the document), although

Data	Weights	Vote	Sum	3Est	TF	A·L	Inv$^{1.2}$	Pool$^{1.4}$
Pop	Unweighted	81.49	81.82	81.49	84.42	80.84	87.99	80.19
Pop	Tuned	81.90	82.90	82.20	87.20	83.90	90.00	80.60
Pop	Best	81.82	83.44	82.47	87.66	86.04	90.26	81.49

Table 1: Experimental Results for Tuned Assertion Certainty. All values are percent accuracy.

we do not know what those levels of certainty are. These can be discovered by tuning with a subset of the test set (and evaluating on the remainder), varying the relative weights of the "infobox", "elsewhere", and "field of interest" assertions. Our results are shown in Table 1; the algorithms we compare on are Vote (weighted voting, where belief in a claim c is given by $\sum_{s \in S_c} \omega(s,c)$), Sum (Hubs and Authorities), 3Est (3-Estimates), TF (TruthFinder), A·L (Average·Log), Inv$^{1.2}$ (Investment with $\mathcal{G} = x^{1.2}$) and Pool$^{1.4}$ (PooledInvestment with $\mathcal{G} = x^{1.4}$). Each fact-finder was run for 20 iterations. Note that in the "unweighted" case only direct edits to the "field of interest" are considered, and "infobox" and "elsewhere" edits are ignored (giving all edits equal weight fares much worse).

We tuned over 208 randomly-chosen examples and evaluated on the remaining 100, repeating the experiment ten times. We also tuned (and tested) with all 308 labeled examples to get the "Best" results, only slightly better than those from legitimate tuning. As expected, assigning a smaller weight to the "infobox" assertions (relative to the "field of interest") and a much lower weight to the "elsewhere" assertions yielded the greatest results, confirming our common-sense assumption that edits close to a field of interest confer more supervision and implicit approval than those elsewhere on the page. We found a significant gain across all fact-finders, notably improving the top Investment result to 90.00%, a greater improvement than what [Pasternack and Roth, 2010] achieved using common-sense knowledge about the claims and demonstrating that generalized fact-finders can dramatically increase performance.

5.3 Uncertainty in Information Extraction

We next consider the case where the information extractor is uncertain about the putative claims, but provides an (accurate) estimate of $\omega_u(s,c) = P(s \to c)$, the probability that source s made a given claim c.

For the Population dataset, we augment each mutual exclusion set M with an additional (incorrect) claim, ensuring $|M| \geq 2$. For each assertion $s \to c$ we select another $c' \in M_c$, and draw a p from a Beta(4,1) distribution ($\mathbb{E}(p) = 0.8 \Rightarrow 20\%$ chance of error). We then set $\omega_u(s,c) = p$ and $\omega_u(s,c') = 1 - p$. In the unweighted case (where edge weights must be 0 or 1), we keep the edge between s and c if $p \geq 0.5$, and replace that edge with one between s and c' if $p < 0.5$.

For the Books dataset, each mutual exclusion set had exactly two claims (a person is either an author of a book or he is not) and thus did not require augmentation. Here we drew p from a Beta(2,1) distribution ($\mathbb{E}(p) = 2/3$), corresponding to a greater (33%) chance of error. Our results are shown in Table 2; on both datasets, generalized fact-finders easily outperform their standard counterparts.

5.4 Groups as Weighted Assertions

Using the Population data we considered three groups of editors: administrators, blocked users, and regular users with at least one template on their user page (intended to capture more serious editors). To keep things simple, we allowed each user to belong to at most one of these groups—if an administrator had been blocked, he nonetheless belonged to the administrator group; if an otherwise "regular" user were blocked, he (of course) belonged to the blocked group. Given that administrators are promoted to that position by being trusted by other Wikipedia editors, and that blocked users are blocked by trusted administrators for (presumable) misbehavior, we expected that administrators will be relatively trustworthy on the whole, while blocked users will be more untrustworthy, with serious editors somewhere in between. We then encoded these groups as weighted assertions, using ω_g with arbitrarily chosen β parameters, as shown in Table 3. We see improved performance with all β values tested, with the exception of the Investment algorithm, which requires a much lower β; we can conclude from this that β should be tuned independently on each fact-finder for best results.

5.5 Groups as Additional Layers

We next took the same three groupings of editors (administrators, blocked users, and regular users) and added them as a third layer in our generalized fact-finders, continuing to use the same Population dataset as before. For most fact-finders, we can directly adapt the T and B functions as U and D functions, respectively, though this excludes PooledInvestment (which depends on mutual exclusion sets) and 3-Estimates (whose "claim difficulty" parameters are not readily extended to groups). In the former case, we can calculate the trustworthiness of the groups in the third layer as a weighted average of the trustworthiness of its members, giving us $U_3^i(g) = \sum_{s \in g} U_2^i(s)/|g|$, where g is a group and $|g|$ is the number of sources it contains. Likewise, we can calculate the trustworthiness a source inherits from its groups as the weighted average of the groups' trustworthiness, giving $D_2^i(s) = \sum_{g \in G_s} D_3^i(g)/|G_s|$, where G_s is the set of groups to which source s belongs (recall that, since there are three layers, $D_3^i(g) = U_3^i(g)$). We can use these new U_3 and D_2 functions to handle the interaction between the group layer and the source layer, while continuing to use an existing fact-finder to mediate the interaction between the source layer and claim layer. We apply this hybrid approach to two fact-finders, giving us Inv$^{1.2}$/Avg, and Pool$^{1.4}$/Avg. Finally, note that regardless of the choice of D_2, we are discarding the

Data	Assertions	Vote	Sum	3Est	TF	A·L	$\text{Inv}^{1.2}$	$\text{Pool}^{1.4}$
Pop	Unweighted	71.10	77.92	71.10	78.57	76.95	78.25	74.35
Pop	Generalized (Weighted)	**76.95**	**78.25**	**76.95**	**80.19**	**78.90**	**84.09**	**78.25**
Books	Unweighted	80.63	77.93	80.74	80.56	79.21	77.83	81.20
Books	Generalized (Weighted)	**81.88**	**81.13**	**81.88**	**82.90**	**81.96**	**80.50**	**81.93**

Table 2: Experimental Results for Uncertainty in Information Extraction

β	Vote	Sum	3Est	TF	A·L	$\text{Inv}^{1.2}$	$\text{Pool}^{1.4}$
(No groups)	81.49	81.82	81.49	84.42	80.84	87.99	80.19
0.7	**84.09**	**84.09**	**84.42**	**85.71**	**84.74**	84.74	**83.44**
0.5	83.77	**84.09**	**84.42**	85.06	84.09	87.01	82.79
0.3	82.47	83.77	83.77	84.74	83.77	87.01	82.79
0.00001	83.44	82.14	83.44	84.42	81.49	**88.96**	80.51

Table 3: Experimental Results for Groups using Weighted Assertions.

Description	Sum	TF	A·L	$\text{Inv}^{1.2}$	$\text{Inv}^{1.2}$/Avg	$\text{Pool}^{1.4}$/Avg
No Groups	81.82	**84.42**	80.84	87.99	87.99	80.19
Group Layer	83.77	83.44	**84.42**	52.92	88.64	64.94
Group Layer with D_2^{smooth}	**84.74**	84.09	82.79	**88.96**	**89.61**	**84.74**
Tuned + Group Layer	**86.10**	83.30	**87.00**	**88.50**	**90.00**	77.90
Tuned + Group Layer with D_2^{smooth}	83.20	**85.30**	84.20	87.40	**90.00**	**83.50**

Table 4: Experimental Results for Groups as an Additional Layer.

trustworthiness of each source as established by its claims in favor of the collective trustworthiness of its groups, an information bottleneck. When we have ample claims for a source, its group membership is less important; however, when there are few claims, group membership becomes much more important due to the lack of other "evidence". The previously described D_j^{smooth} captures this idea by scaling the impact of groups on a source by the (weighted) number of claims made by that source. We show results both with and without this smoothing in Table 4.

Except for TruthFinder, groups always improve the results, although "smoothing" may be required. We also tuned the assertion certainty as we did in Table 1 in conjunction with the use of groups; here we find no relative improvement for Investment or TruthFinder, but gain over both tuning and groups alone for all other fact-finders.

6 Conclusion

Generalized fact-finding allow us to incorporate new factors into our trust decisions, such as information extraction and source uncertainty, similarity between the claims (previously evaluated with success by [Yin *et al.*, 2008], but applicable only to TruthFinder), and source groupings and attributes. As our experiments have shown, this additional information provides a substantial performance advantage across a broad range of fact-finders.

Acknowledgements

This research sponsored by the Army Research Laboratory (agreement W911NF-09-2-0053), and the DHS CCICADA Multimodal Information Access & Synthesis Center. All opinions, findings, conclusions or recommendations are solely those of the authors.

References

[Dong *et al.*, 2009] X.L. Dong, L. Berti-Equille, and Divesh Srivastava. Truth discovery and copying detection in a dynamic world. *VLDB*, 2009.

[Galland *et al.*, 2010] Alban Galland, Serge Abiteboul, A. Marian, and Pierre Senellart. Corroborating information from disagreeing views. In *WSDM*, 2010.

[Kleinberg, 1999] J M Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.

[Pasternack and Roth, 2010] Jeff Pasternack and Dan Roth. Knowing What to Believe (when you already know something). In *COLING*, 2010.

[Pasternack and Roth, 2011] Jeff Pasternack and Dan Roth. Generalized Fact-Finding. In *WWW*, 2011.

[Wu and Weld, 2007] Fei Wu and Daniel S. Weld. Autonomously semantifying wikipedia. *CIKM*, 2007.

[Yin *et al.*, 2008] Xiaoxin Yin, Philip S. Yu, and Jiawei Han. Truth Discovery with Multiple Conflicting Information Providers on the Web. *IEEE Transactions on Knowledge and Data Engineering*, 20(6):796–808, 2008.