Trust Analysis with Clustering

Manish Gupta Yizhou Sun Jiawei Han
University of Illinois at Urbana Champaign
{gupta58,sun22, hanj}@illinois.edu

ABSTRACT

Web provides rich information about a variety of objects. Trustability is a major concern on the web. Truth establishment is an important task so as to provide the right information to the user from the most trustworthy source. Trustworthiness of information provider and the confidence of the facts it provides are inter-dependent on each other and hence can be expressed iteratively in terms of each other. However, a single information provider may not be the most trustworthy for all kinds of information. Every information provider has its own area of competence where it can perform better than others. We derive a model that can evaluate trustability on objects and information providers based on clusters (groups). We propose a method which groups the set of objects for which similar set of providers provide "good" facts, and provides better accuracy in addition to high quality object clusters.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information filtering; H.2.8 [**Information Systems Applications**]: Database Applications—*Data mining*

General Terms

Algorithms, Experimentation, Measurement

Keywords

trust, fact finding, clustering

1. INTRODUCTION

Large amounts of structured information are available on the web. Consider the set of author lists for different books available on book selling websites. First there is a need to establish the trustworthiness of each of the websites across all the different books and then there is a need to cluster the books according to the similarity between the sets of "good" websites for each of the books. Note that this clustering is based on trustworthiness and may be quite different from natural clustering on a single dimension.

2. MOTIVATION AND RELATED WORK

In presence of conflicting time-varying information provided by a large number of possibly dependent sources, voting may not be the best method for veracity analysis. Yin et al. [4] presented the basic truth finder model which aimed at finding true facts from a large amount of conflicting information on many subjects that is provided by various web sites. Dong et al. [1] studied the problem of finding true values and determining the copying relationship between sources, when the update history of the sources is known. We perform a clustering and ranking of websites and objects iteratively. The closest related clustering work is RankClus [3]. We use a similar philosophy for the design of our algorithm.

In [4], provider trust depends on confidence of facts (author lists) published by that provider, while confidence of a fact depends on trustworthiness of the providers that publish that fact and confidence of other related facts. They compute the confidence of the facts and the overall trustworthiness rankings of the providers iteratively in terms of each other. Assumption is that a trustworthy provider (website) provides the right information for all the objects (books), which may not be true. It might be good to propagate trust information of a provider to recompute the confidence of facts of only those objects for which the provider is considerably trustworthy. The problem is how to define clusters of objects such that those objects for which a group of top providers have similar trust are grouped together.

WWW 2011, March 28–April 1, 2011, Hyderabad, India.
ACM 978-1-4503-0637-9/11/03.

3. THE ITERATIVE FACT FINDER MODEL

Basic truth finder [4] provides a model to compute global trustworthiness of providers and ranks facts associated with the objects based on their confidence. Trustworthiness of provider p is $t(p)$ and $s(f)$ is confidence of the fact f. Each fact is associated with an object. Optionally, implication from fact f_1 to fact f_2 ($imp(f_1 \rightarrow f_2)$) denotes influence of fact f_1 on fact f_2. Let $P(f)$ denote the set of providers that publish fact f and $F(p)$ denote the set of facts provided by provider p. Pasternack et al. [2] introduce a few more fact finders (Sums, Average.Log, Investment) which are based on the same framework as truth finder [4], but differ in the way the confidence and trust values are computed.

Figure 1 provides an example that shows why cluster based ranking of providers can be different from global ranking and how it can be useful for computing fact confidences. Providers p_1 and p_2 provide facts f_{ij} (i^{th} provider, j^{th} object) for five objects o_1 to o_5. p_1 provides good facts for objects o_1 and o_2 (and bad facts for o_3 and o_4) while p_2 provides good facts for the other three objects (and bad facts for o_1 and o_2). Since p_2 provides good facts for more objects, most fact finders would rank p_2 higher than p_1. But if we look at the trust profiles of the objects in the provider space, we notice that objects o_1 and o_2 have a similar profile while the objects o_3, o_4, o_5 have similar profiles. Thus, we can cluster objects into two clusters. We notice that for cluster c_1, p_1 would be ranked higher than p_2 and vice versa for cluster c_2. Thus, if we cluster objects in the provider trustworthiness space, we may derive interesting clusters and thereby improve trust and confidence computations. Note that an object can belong to only one cluster, but a provider may be an expert for multiple clusters of objects.

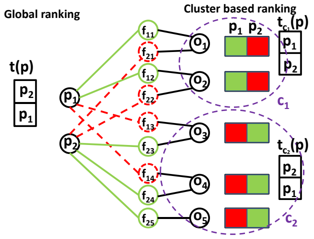

Figure 1: Trustworthiness information propagation should be restrictive

4. OUR APPROACH

We want to do **trust analysis** based on trustworthiness of providers and confidence of facts related to the objects, and obtain **clustering** of objects. We hypothesize that objects can be clustered based on provider trustworthiness profiles ($t_o(p)$) personalized to the particular object. Also, restrictive flow of trust information across objects, using clusters, can improve ranking accuracy of facts and providers.

Table 1: Accuracy (TF=Truth Finder [4], AL=Average Log [2], Inv=Investment [2])

	books-orig				population				bdate				ddate			
	Voting	Basic	BCFF	ACFF	Voting	Basic	BCFF	ACFF	Voting	Basic	BCFF	ACFF	Voting	Basic	BCFF	ACFF
TF	87.676	**93.495**	93.162	**93.495**	95.494	**97.077**	97.045	96.987	89.261	82.261	82.261	82.261	92.857	**96.065**	95.341	**96.065**
Sums	87.676	85.043	**86.293**	85.210	95.494	**97.534**	97.435	**97.534**	89.261	**90.913**	86.609	86.609	92.857	93.746	**95.522**	93.971
AL	87.676	86.970	86.937	**87.520**	95.494	96.710	**97.273**	96.958	89.261	90.913	86.609	**90.957**	92.857	93.261	95.116	**95.522**
Inv	87.676	88.192	**88.551**	88.151	95.494	97.436	97.530	**97.573**	89.261	**90.957**	86.609	**90.957**	92.857	94.123	95.116	**95.312**

If we know some natural clustering $C = \{c_k\}_{k=1}^{K}$, we can run basic fact finder model for each of the clusters of objects separately. But, we would not use the information about providers related to objects in other clusters. Also, this method needs some input clustering. Clusters are fixed and depend on a particular dimension only.

Alternatively, we can compute provider trust per object and then cluster the objects using their object-conditional trust vectors ($t_o(p)$) as shown in Algorithm 1. $t_o(p)$ is computed as the confidence of the fact provided by provider p for the object o. It denotes the trust of information provider p conditioned with respect to the object o.

Algorithm 1 Basic Cluster-based Fact Finder (BCFF) (using TF)

```
Input: 1. Facts $f$ provided by different providers related to objects $o \in O$. 2. Implications matrix imp.
Initialize $t_o(p)$ to a value $v$ $\forall w$, where $0 \leq v \leq 1$, $o \in O$ where $O$ is the set of all objects.
while $\{\exists o||t_o^t - t_o^{t-1}| \geq \delta\}$ do
    For every fact $f$, $\sigma(f) = log(\prod_{p \in P(f)}(1 - t_o(p)))$
    For every fact $f$,
    $\sigma^*(f) = \sigma(f) + \rho \sum_{o(f')=o(f)} \sigma(f')imp(f' \to f)$
    For every fact $f$, $s(f) = \frac{1}{1+e^{\gamma\sigma^*(f)}}$
    For every provider $p$, $t_o(p) = s(f)$ where $f = f_{po}$.
end while
Cluster $t_o$ vectors using KMeans.
return  $s(f)$ for every $f$, object clusters and topK most trustworthy providers for each cluster
```

This kind of clustering in the trust space is a novel form of clustering and may provide quite different clusters compared to clustering on natural dimensions. No training data or data related to any other dimensions of the objects are needed. But, note that there is no trust information sharing between objects in BCFF. Every iteration in Algorithm 1 simply recomputes trust of providers based on implications between various facts about the same object. Our algorithm should start with an initial clustering, perform trust analysis using this clustering (similar to BCFF) and then refine the clusters using the analysis obtained on the previous set of clusters. So, we define cluster-conditional trust, $t_{c_k}(p)$, as the trust of the provider p considering the facts published by the provider p related to objects in cluster c_k.

Algorithm 2 outlines our method which ensures the right information flow among related (in the sense of provider trust) objects only. It performs alternate clustering and trust analysis iterations. The clustering steps bring similar objects together, while the trust analysis steps compute better cluster-conditional trust rankings and better fact confidence values. While performing trust analysis, the flow of trustworthiness information for a provider is restricted to be within the current cluster. Modifications in fact confidence values (in analysis steps) leads to better object-conditional trust vectors which are then re-clustered to get modified clusters using KMeans clustering.

Algorithm 2 Advanced Cluster-based Fact Finder (ACFF)

```
Input: 1. Facts $f$ provided by different providers related to objects $o \in O$. 2. Implications matrix imp.
Obtain clusters $\{c_k\}_{k=1}^{K} \leftarrow$ BCFF
while No change in clusters $c_k$ do
    Iteratively compute $t_c(p)$ for all clusters and providers and $s(f)$ for all facts.
    Compute $t_o(p)$ for all objects using fact confidences $s(f)$.
    $\{c_k\}_{k=1}^{K} \leftarrow KMeans(t_o)$. Clustering is done on data of size ($\#clusters \times \#providers$).
end while
return  $s(f)$ and $t_k(p)$ for every $f$ and $p$ for every cluster $c_k$
```

Clusters can be considered distinct if cluster conditional trust vectors are far apart from each other. Else, clustering is not really effective and hence our algorithms would not provide much gains. So, we perform smoothing and compute best fact $= argmax_f((1-\alpha)s_C(f) + \alpha s_G(f))$ where s_C and s_G are cluster based and global fact confidence scores resp. α was set to avg. cosine similarity between cluster centroids.

5. EXPERIMENTS AND RESULTS

We experimented with multiple datasets: Abebooks.com Books Dataset (provided by Yin [4]), Wikipedia Biography Infobox Datasets and population dataset (subset of one used by Pasternack [2]). We cleaned all datasets to ensure that (provider, object) is a primary key. Original datasets somehow failed to maintain this constraint. In books (1265) datasets, facts are author lists, providers are book websites (894); ground truth (100 books) is obtained manually from book cover scans. In biography infobox dataset (258 people), facts are birth and death dates and providers are contributors (4392) on wikipedia; ground truth (24 bdates, 182 ddates) is obtained by consensus from multiple websites. For population data (30011 cities), fact is population of cities, providers are different wikipedia contributors (1361); ground truth (290 cities) is obtained from US Census data for 2000.

We use two metrics: accuracy and compactness. Accuracy measures accuracy of most confident fact for any object obtained by different algorithms. Accuracy of a fact is defined differently for different types of facts (Books: same as [4], population: $1 - \frac{diff}{max}$, dates: $1 - \frac{diff\ in\ days}{100}$). Compactness is used to evaluate clustering quality and is defined as intra-cluster similarity:avg. inter-cluster similarity.

Table 1 shows accuracy gains of our cluster based fact finders over basic fact finders. For books dataset, clustering obtained does not seem to match with any natural clustering. For population dataset, with five clusters, we observe: contributors are clustered into (IL, CA, NY), (PA, VT, MI), (IL, AL, AR), (IN, IA, GA), (MN, OH, NY). Note we used cities for experiments, but report clusters on states for clarity.

6. CONCLUSION

We proposed algorithms for trust analysis using cluster based methods. We showed using four datasets that our algorithms perform better than traditional fact finders and generate interesting clusters. In the future, we plan to refine our clustering methods, e.g., by clustering in other spaces.

Acknowledgements: Thanks to X. Yin and J. Pasternack for their datasets, and V. Vydiswaran and anonymous reviewers for comments. Research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-09-2-0053, DHS, and NSF IIS-09-05215. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government.

7. REFERENCES

[1] X. L. Dong, L. Berti-Equille, and D. Srivastava. Truth discovery and copying detection in a dynamic world. *PVLDB*, 2(1):562–573, 2009.

[2] J. Pasternack and D. Roth. Knowing what to believe (when you already know something). *Coling 2010*, 877–885, Beijing, China.

[3] Y. Sun, J. Han, P. Zhao, Z. Yin, H. Cheng, and T. Wu. Rankclus: integrating clustering with ranking for heterogeneous information network analysis. *EDBT*, 565–576, 2009.

[4] X. Yin, J. Han, and P. S. Yu. Truth discovery with multiple conflicting information providers on the web. *TKDE*, 20(6):796–808, 2008.