Domain-Aware Multi-Truth Discovery from Conflicting Sources

Xueling LIN, Lei CHEN
Department of Computer Science and Engineering
Hong Kong University of Science and Technology, Hong Kong
xlinai@cse.ust.hk, leichen@cse.ust.hk

ABSTRACT

In the Big Data era, truth discovery has served as a promising technique to solve conflicts in the facts provided by numerous data sources. The most significant challenge for this task is to estimate source reliability and select the answers supported by high quality sources. However, existing works assume that one data source has the same reliability on any kinds of entity, ignoring the possibility that a source may vary in reliability on different domains. To capture the influence of various levels of expertise in different domains, we integrate domain expertise knowledge to achieve a more precise estimation of source reliability. We propose to infer the domain expertise of a data source based on its data richness in different domains. We also study the mutual influence between domains, which will affect the inference of domain expertise. Through leveraging the unique features of the multi-truth problem that sources may provide partially correct values of a data item, we assign more reasonable confidence scores to value sets. We propose an integrated Bayesian approach to incorporate the domain expertise of data sources and confidence scores of value sets, aiming to find multiple possible truths without any supervision. Experimental results on two real-world datasets demonstrate the feasibility, efficiency and effectiveness of our approach.

PVLDB Reference Format:
Xueling LIN, Lei CHEN. Domain-Aware Multi-Truth Discovery from Conflicting Sources. *PVLDB*, 11(5): 635 - 647, 2018.
DOI: https://doi.org/10.1145/3177732.3177739

1. INTRODUCTION

In the information explosion era, not all the data collected from the Web is correct. There exist conflicts in the answers provided by different data sources on the same set of questions or facts. For example, one online bookseller may provide a complete author list of a book, while another bookseller only provides the first author, or makes a mistake by treating press information as the author. Therefore, a key challenge in data integration is to derive the most complete and accurate aggregated records from diverse and sometimes conflicting sources.

Proceedings of the VLDB Endowment, Vol. 11, No. 5

DOI: https://doi.org/10.1145/3177732.3177739

One straightforward approach to determine the truth is to conduct majority voting on the collected dataset, which selects the majority answers as the output. Nevertheless, majority voting fails to take the reliability levels of different sources into consideration, which may lead to poor performance when the number of low-quality sources is large.

Thus, a better approach is multi-source aggregation, which evaluates the trustworthiness of each source. In order to solve this problem, many works have been proposed to derive the correct answers from a collection of data, with source reliability estimation serving as an important component [1] [3] [6] [7] [10] [11] [13] [14] [16] [17] [18] [19]. All these works follow an essential principle that reliable sources tend to provide more trustworthy information. Based on this principle, these works assign higher weights to reliable sources, so that the information from these sources can make a greater contribution in the truth discovery process.

The same principle is also utilized in multi-truth discovery, where multiple true values might exist for a single item. For example, a book or a conference paper might be collaborated by several authors. Currently there are a few works that pay attention to the multi-truth finding problem. *LTM* [19] proposes a probabilistic graphical model to discover multiple truths for each object, taking both false positive and false negative claims of the sources into consideration. Similarly, in [12] and [15], the authors measure the quality of a source as its precision and recall, then derive multiple truths of a fact based on the quality of sources that provide it.

However, in both single-truth and multi-truth scenarios, it is unfair to assign only one reliability score to each source. Different fractions of data from the same source can have different qualities. In fact, none of the sources is promised to be expert in every field. Source reliability usually varies among different domains. For example, a bookseller on the Web may provide more abundant and precise data for books in the science category, but less and lower quality data for books in the arts category; a movie website may be particularly accurate with respect to romantic comedies, but less reliable in action and adventure movies. Thus, it is better to consider domains separately in the truth finding model.

We summarize the major challenges of solving the domain-aware multi-truth finding problem as follows:

1. *Unknown quality of sources in different domains.* Since constructing the training dataset from large-scale data is rather nontrivial, truth finding is usually carried out with unsupervised approaches. In such cases, it is impossible to learn the quality of each source in different domains from the beginning, because none of the sources are guaranteed to provide 100% accurate information. Since inferring the truth relies on the reliability of sources significantly, truth finding results can be easily distorted by malicious sources. Therefore, it is

essential to develop promising methods to infer the sources' trustworthiness in various domains in an unsupervised way.

2. *Derive the proper domain subdivisions.* It is difficult for us to derive the proper domain subdivisions automatically in a straightforward way. Since there are different attributes related to an item, we need to choose the most suitable one for domain classification. For instance, predicting whether a source is more accurate with respect to books in science, or books published after 2010, is a nontrivial task.
3. *Discover the influence and correlations between different domains.* Domains in each source may not be independent. For example, a source with high reliability in action movies may also provide relatively high quality data for adventure movies.
4. *Unknown confidence of data values.* Compared with the single-truth finding problem, the multi-truth finding problem makes unique assumptions about the confidence of data values. In the multi-truth finding problem, it is common that most of the sources provide partially correct values for an item. Our task is to aggregate those values and infer the truth in the real-world. In such a scenario, a source claiming one value of an item does not imply that it opposes the other values claimed by other sources of the same item.

In this paper, we address the problem of discovering multi-truth on data provided by multiple sources in various domains. We derive the domain expertise of each source based on the information richness of the sources in various domains. We also investigate the correlations among different domains provided by each source. We then apply Bayesian analysis to infer the trustworthiness of each source in different domains as well as the truthfulness of values provided for each data object simultaneously.

To summarize, our main contributions are listed as follows:

1. We recognize the difference in source reliability among domains on the truth discovery task, and propose to incorporate the estimation of fine-grained domain-aware reliability into truth discovery.
2. We study the correlations between different domains of the data. We propose the concept of influence between domains to represent the possible relationship between various domain information provided by a source.
3. We propose a principled probabilistic Bayesian based approach to aggregating true answers and discovering source reliability without any supervision. Our method provides a principled avenue for incorporating domain expertise as priori knowledge of the sources into the truth discovery process. In particular, to solve the multi-truth finding problem, we define a method for calculating the mutual exclusion between different values. It follows the implication of the multi-truth finding problem: instead of directly rejecting the unclaimed values, a source is regarded as a partial provider of its unclaimed values. Our method naturally supports multiple truths for an entity and achieves more effective performance.
4. The experiments on two real-world datasets show that the proposed approach can significantly reduce the error rate compared with existing methods in multi-source aggregation.

In the following sections, we first describe our data model and formalize the problem in Section 2. In Section 3, with motivating examples, we introduce the approach to learn the domain expertise of each source based on its information richness of certain domains, and study the correlations and inference between the domains. We then illustrate the integrated Bayesian approach to infer the trustworthiness of the sources and the truthfulness of values for objects in Section 4. Section 5 presents our experimental results. We discuss the related work in Section 6 and conclude in Section 7.

2. PROBLEM FORMULATION

In this section, we provide the details of our data model and formally define the domain-aware multi-truth finding problem.

In general, a source provides information on an item about several attribute types. For each attribute, there are different domains. For example, an online bookseller provides *category* and *published year* of the books. For *category*, there are different domains, such as *science, arts, literature* and *biographies*; For *published year*, there are also different domains, such as *"1901 to 1920"*, and *"after 2000"*. In our paper, in order to simplify the discussion, we assume that different attribute types are independent and can be dealt with individually. We also assume that sources are independent.

Our truth finding problem considers a set of sources $\mathcal{S} = \{s_1, s_2, ..., s_n\}$, which provide values on a set of objects $\mathcal{O} = \{o_1, o_2, ..., o_m\}$ in a domain set $\mathcal{D}_a = \{d_1, d_2, ..., d_D\}$ of attribute a. Each object $o \in \mathcal{O}$ corresponds to a domain $d_i \in \mathcal{D}_a$. For example, the book *"Lake Champlain: Partnerships and Research in the New Millennium"* (ISBN10: 0306484692) is regarded as an object and it is in the *science* domain of attribute *book category*. We use o^d to indicate that object o resides in domain d.

Let $O(s)$ be the set of objects that source s provides values for. For each object $o \in O$, a source $s \in S$ can provide a value v. We note that there might be multiple true v for an object, which means that among the different values provided by an object, one or more of the values describe the real world and are *True*; the rest that conflict with the reality are regarded as *False*. Moreover, we denote the set of values claimed for the object o by source s as $V_s(o)$.

Now, we introduce some important definitions in this paper.

DEFINITION 1. *The veracity score of a value v, denoted by $\sigma(v)$, is the probability of v being correct.*

DEFINITION 2. *The trustworthiness of a data source s in domain d , denoted by $\tau_d(s)$, is the probability that s provides true values in d.*

We formally define our problem as follows.

Given a source set $\mathcal{S} = \{s_1, s_2, ..., s_n\}$, a domain set $\mathcal{D}_a$, the objects set $\mathcal{O} = \{o_1^{d_1}, o_2^{d_2}, ..., o_m^{d_D}\}$ where $d_1, d_2, ..., d_D \in \mathcal{D}_a$, the value set $V_s(o)$ for each s and o, our goal is to learn the veracity score $\sigma(v)$ of each value v provided for each object o, as well as $\tau_d(s)$ of each source s in each domain d.

Table 1 shows the variables and parameters used in the following discussion.

3. DOMAIN EXPERTISE INFERENCE

In this section, we first introduce our approach to infer domain expertise from the information richness in different domains with real-world examples. Then we propose a method for learning the correlation between different domains.

3.1 Motivating Examples

After obtaining information from sources and domains, the major challenge is to infer the domain-aware trustworthiness of each source in an unsupervised manner. In order to determine the sources'

Table 1: Notations used in this paper

Notation	*Description*
$\mathcal{S}$	Set of all data sources
$\mathcal{D}_a$	Set of all domains in attribute a
$\mathcal{O}$	Set of objects
o^d	An object o in domain d
$O^d(s)$	Set of objects in domain d provided by source s
$V_s(o)$	Set of values claimed for object o by source s
$S_o(v)$	Set of sources claiming value v for object o
$P_d(s)$	Domain percentage of source$s$ in domain d
$r_d(s)$	Domain richness score of source$s$ in domain d
$inf_s(d_k \rightarrow d_i)$	Inference from domain d_k to domain d_i
$e_d(s)$	Domain expertise score of source s in domain d
$\psi(o)$	Observation of the values provided for object o
$\sigma(v)$	Veracity of a value v
$\tau_d(s)$	Trustworthiness of a data source s in domain d
$c_s(v)$	Confidence score of value v provided by source s

expertise, one heuristic is to consider the distribution of the information richness of sources in different domains. The reason is that for some sources, the information richness usually varies a lot among different domains according to the domain expertise levels. For instance, a bookseller with large amounts of books in science, but only a few books related to arts, is more likely to be an expert in the science domain. Thus, we consider inferring the initial domain expertise score from the data richness of a particular domain.

We first define a unique factor called *global domain percentage*, denoted by $P_d(s)$, for each source s in each domain d. $P_d(s)$ is the percentage of data quantity provided by source s with respect to the total data quantity in domain s. It is computed in Equation 1, where $|O^d(s)|$ stands for the size of the set of objects provided by s in domain d.

$$P_d(s) = \frac{|O^d(s)|}{\sum_{s \in S} |O^d(s)|} \quad (1)$$

$P_d(s)$ measures the amount of data provided by source s in a certain domain d. It is only related to the total amount of data in the specified domain d, but has no correlations with any other domain.

We next use three real-world examples to illustrate the feasibility of the heuristic of inferring the initial domain expertise score from data richness.

EXAMPLE 3.1. *We consider a dataset that we collected from an online bookstore aggregator, AbeBooks.com. This dataset includes 54,591 bookstores (each corresponding to a data provider), together providing 210,206 books in 18 different categories, such as science, arts, literature and business. We have collected all the books provided by each bookseller and the category information of each book. We identify a book by its ISBN. We select 4 from all the collected booksellers for further discussion in our example. Table 2 shows an example that provides the data richness of each data source*[1]*. Table 3 shows the percentage distribution of the data sources in different categories. For example, the total quantity of science books is 10,000, where Strand provides 500 of them. Therefore, the global domain percentage in science of Strand is 500/10,000 = 0.05. Note that Strand's global domain percentage in children's books is 400/500 = 0.8, which is much higher than its global domain percentage in science, even though it provides more science books than children's books, since the total quantity of science books is much larger than that of children's books.*

This example shows that in some datasets it is common to have an uneven quantity distribution in different data categories. We illustrate how such an uneven distributed information richness reflects the data reliability in certain domains and contributes to truth finding results.

[1]For simplicity, we have rounded the number to the nearest tenth and hundredth

Table 2: The number of facts in different domains provided by 4 booksellers

Booksellers	*Number of Books in Different Categories*			
	science	*travel*	*arts*	*children*
Ergodebooks	7000	200	800	40
Stortbooks	2000	1500	100	40
Hennessey	500	100	4000	20
Strand	500	200	100	400
Total	10000	2000	5000	500

Table 3: The domain percentage distribution of 4 booksellers

Booksellers	*Percentages in Different Categories*				*std*
	science	*travel*	*arts*	*children*	
Ergodebooks	0.7	0.1	0.16	0.08	0.2557
Stortbooks	0.2	0.75	0.02	0.08	0.2888
Hennessey	0.05	0.05	0.8	0.04	0.3262
Strand	0.05	0.1	0.02	0.8	0.3231

EXAMPLE 3.2. *Continue with the same dataset. We observe that different booksellers provide diversified author lists for different books. We show the crawling results in Table 4. For evaluation purposes, we have manually checked the images of book covers of 407 randomly selected books as the ground truth. The facts provided by sources that match with the truth are marked in bold.*

Take the first book (ISBN10: 0306484692), a science book, as an example. From the image of the book cover, we find out that Ergodebooks and Irish BookSellers provide the most accurate information, while the information from BookExpress is incomplete, and that from BookVistas is incorrect. However, Ergodebooks does not always perform well. It provides an incomplete author list for the last book (ISBN10: 0007123655), which is categorized in children's books. Note that as shown in Table 3, Ergodebooks achieves a higher domain percentage in science than in children's books.

This example illustrates that for some particular cases, those sources with richer data in one domain provide higher quality results compared to other sources. Actually, such rules also apply to most of the cases in this book dataset, as shown in the next example.

EXAMPLE 3.3. *Continue with the same dataset. We use the global domain percentage (as calculated in Example 3.2) as the weights and assign weight to each of 18 domains of every data source. We then conduct a weighted voting on the dataset and select the authors with the highest voting score as output. We also conduct another version of weighted voting, in which we only select the top 10% of sources with the highest global domain percentage to be involved in the weighted voting process.*

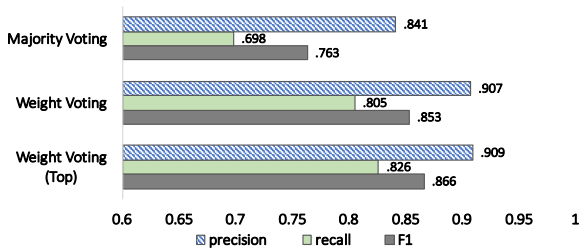


Figure 1: Comparison of majority voting and weighted voting on the book-author data set.

Table 4: Real-world example of conflicting Information about book authors

Book ISBN10	*Category*	*Booksellers*	*Authors*
0306484692	science	**Ergodebooks**	**Tom Manley; Pat Manley; Timothy B. Mihuc**
		Irish BookSellers	**Manley, Tom; Manley, Pat; Mihuc, Timothy B.**
		BookVistas	MIHUC
		BookExpress	Tom Manley
000215644X	travel	**Hemingway Ventures Ltd.**	**Malcolm Muggeridge; Alec R Vidler**
		AwesomBooks	**Malcolm Muggeridge; Alec R. Vidler**
		Ergodebooks	Malcolm Muggeridge
		BookExpress	Malcolm Muggeridge
0002557541	business and economics	**Ergodebooks**	**Lee Selleck; Francis Thompson**
		Irish BookSellers	**Lee Selleck; Francis Thompson**
		Better World Books	**Lee Selleck; Francis Thompson**
		BookExpress	Lee Selleck
0007123655	children	**AwesomBooks**	**Enid Blyton; Chorion CGI**
		Ergodebooks	Blyton, Enid
		Irish BookSellers	Blyton, Enid
		Better World Books	Blyton, Enid
		Hemingway Ventures Ltd.	1

Table 5: Example: The adjusted domain expertise scores of 4 booksellers in the attribute *category* when $\alpha = 1$.

Booksellers	*The adjusted domain expertise score*			
	science	*travel*	*arts*	*children*
Ergodebooks	**0.954**	0.4359	0.5426	0.392
Stortbooks	0.6	**0.9682**	0.199	0.392
Hennessey	0.3122	0.3122	**0.9798**	0.28
Strand	0.3122	0.4359	0.199	**0.9798**

We still use the manually labeled ground truth for evaluation. In Figure 1, we compare the two weighted voting results to the majority voting one. The experimental results show that the result of weighted voting based on the domain information richness is considerably higher than the result of majority voting. We measure the correctness with three metrics. Precision measures among the claimed values, how many are indeed true. Recall measures among the labeled true values, how many are claimed by the results. F-measure computes their harmonic mean (i.e. $F1 = \frac{2*prec*rec}{prec+rec}$).

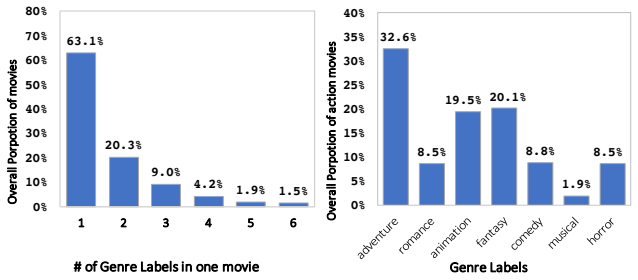
This example demonstrates that sources with higher information richness in certain domains have a positive effect on the truth finding problem on the same domains. It makes sense in reality, because when one source provides more data in one domain, it usually implies that this source tends to be an expert in that domain. Therefore, it is crucial to detect and involve such data richness differences and distinguish source qualities for different domains, so as to improve the data fusion quality.

3.2 Domain Expertise of Data Source

Given a source s with data in different domains, we first compute the domain richness score $r_d(s)$ for s regarding to domain d. $r_d(s)$ is a factor that describes the data richness of s in d. In Equation 2 we compute this score for a certain data source s, based on $P_d(s)$ in source s in domain d. α is a predefined adjust factor based on the distribution of global domain percentage of different sources.

$$r_d(s) = \sqrt{1 - (\alpha \cdot P_d(s) - 1)^2} \quad (2)$$

EXAMPLE 3.4. *Continue with the same dataset in Example 3.1. Table 5 shows the calculated results of the domain richness scores of 4 sources in 4 domains. We use Equation 2 to emphasize and distinguish the differences in the information richness of each source.*

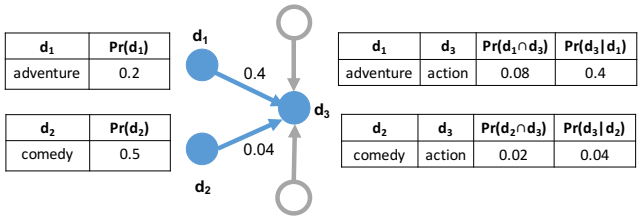


(a) (b)

Figure 2: (a) The number of genre labels that have been assigned to the same movie in the *imdb* dataset. (b) The genres assigned to *action* movies at the same time in the *imdb* dataset.

Nevertheless, although the data items provided by a source are categorized into different domains, such domains may have correlations among one another. For example, the book *Harry Potter and the Deathly Hallows (ISBN: 0545010225)* is categorized as *children's books* and *literature* at the same time by *Fallen Leaf Books, IOBA* at *AbeBooks.com*. Similarly, the movie *Harry Potter and the Goblet of Fire* is assigned to three genres *adventure*, *family* and *fantasy* by *www.imdb.com*. Take the movie dataset of *imdb* as an example. We have collected data of 212,685 movies from *imdb*. As shown in Figure 2(a), more than 36% of the movies have two or more genre labels. Specifically, as illustrated in Figure 2(b), for the movies that are classified into genre *action*, over 32% of them are also classified into genre *adventure*. However, only 1.9% of *action* movies are also put into the genre *musical*. From this example, we can learn that in *imdb*, a movie in the *action* domain may have a higher probability of being classified in the *adventure* domain, which implies that *action* and *adventure* may have a stronger correlation to each other. As demonstrated by this example, for some attributes of the items, some of the domain pairs have a stronger correlation with each other than other pairs. We propose to model the influence between domains using their probability distribution.

We consider the domain set $D = \{d_1, d_2, ..., d_i\}$ of an attribute of an item in the dataset. To discover the correlated domains of domain d_i in source s, we construct a star graph G_s with d_i as the internal node and $\{d_1, d_2, .., d_{k|k\neq i}\}$ as the leaves. The edge from leaf node $d_{k|k\neq i}$ to the internal node d_i represents the influence of domain $d_{k|k\neq i}$ on domain d_i. We define the influence $inf_s(d_{k|k\neq i} \rightarrow d_i)$ of domain $d_{k|k\neq i}$ d_i on d_i as the conditional



d_1	$Pr(d_1)$
adventure	0.2

d_2	$Pr(d_2)$
comedy	0.5

d_1	d_3	$Pr(d_1 \cap d_3)$	$Pr(d_3 \mid d_1)$
adventure	action	0.08	0.4

d_2	d_3	$Pr(d_2 \cap d_3)$	$Pr(d_3 \mid d_2)$
comedy	action	0.02	0.04

Figure 3: An example of probabilistic graph G_s representing the implication between different domains in source s.

probability of d_i given d_k, as shown in Equation 3. For each edge between the internal node d_i and a leaf node $d_{k|k\neq i}$, we assign the $inf_s(d_{k|k\neq i} \rightarrow d_i)$ as its weight.

$$inf_s(d_{k|k\neq i} \rightarrow d_i) = \frac{Pr(d_i \cap d_{k|k\neq i})}{Pr(d_{k|k\neq i})} = \frac{|O^{d_i}(s) \cap O^{d_{k|k\neq i}}(s)|}{|O^{d_{k|k\neq i}}(s)|} \quad (3)$$

Figure 3 illustrates an example of the influence between domains in the movie dataset. Consider nodes d_1, d_2 and d_3 highlighted in G_s. Among the data provided by s, 20% of the movies are classified in *adventure*, while 8% of them are labeled as *adventure* and *action*. Thus, we calculate the conditional probability that a movie is labeled as *action* given that it is already labeled as *adventure*, which is $0.08/0.2 = 0.4$. We regard this conditional probability as the influence of *adventure* on *action*. Similarly, the influence of *comedy* on *action* is calculated as $0.02/0.5 = 0.04$.

Note that $inf_s(d_k \rightarrow d_i) = 0$ when there is no intersection between d_i and d_k in the objects provided by source s. Such cases are applicable when there is no domain overlap in the attribute. For example, the domains of release year attribute, if being classified as *before 1980, 1981 to 2000, 2001 to now*, are independent from each other.

We define influence instead of similarity between domains because such a relationship is asymmetric. Note that we only consider the influence between the leaf nodes and the internal node of the star graph, but neglect the implication between any pair of leaf nodes, since we only pay attention to the influence between pairs of domains, instead of triples. We also consider the implications between domains of each source separately, since different sources have different domain labeling patterns and distributions.

We define the *adjusted domain expertise score* of a source s in domain d_i as:

$$e_{d_i}(s) = r_{d_i}(s) + \rho \cdot \sum_{d_{k|k\neq i}} r_{d_i}(s) \cdot inf_s(d_{k|k\neq i} \rightarrow d_i) \quad (4)$$

ρ is a parameter between 0 and 1 which controls the influence of any related domains. When an overlap between different domains is large, ρ should be set higher. When overlap seldom occurs in different domains, ρ should be set lower.

4. THE TRUTH INFERENCE MODEL

In this section, we propose our solution to the problem of resolving the conflict and deriving the truth. The basic idea is to build a joint probabilistic model, which contains two integral components: (1) the modeling of the domain expertise regarding to data richness, and (2) the modeling of truth aggregation from answers of each source. The first part has been discussed in Section 3. We present the details of the second part in Section 4.1. We propose the integrated Bayesian joint probabilistic model in Section 4.2, and demonstrate the algorithm in Section 4.3.

4.1 Truth Confidence Calculation

In order to model the truth aggregation from claims of each source, we first calculate the confidence of each value provided by each source. To determine the truth among multiple value candidates, the basic idea is that reliable sources will provide trustworthy information in certain domains with higher confidence, thus the truth should be closer to the claims from sources that are more reliable in these domains. A lot of truth discovery methods use weighted voting or average to aggregate the truth [1] [3] [6] [7] [10] [11] [13] [14] [16] [17] [18] [19], which overcomes the unfairness brought by the traditional voting and averaging scheme that assumes every source is equally reliable.

However, traditional single-truth finding methods assume that when a source supports one or more answers, it opposes the other potential answers. This assumption may not hold in the multi-truth problem, where sources may provide partially true values.

EXAMPLE 4.1. *Continue with the same book dataset mentioned in Example 3.1. The true value of the author list of book "Lake Champlain: Partnerships and Research in the New Millennium" (ISBN10: 0306484692) should be "Tom Manley; Pat Manley; Timothy B. Mihuc". However, the bookseller "Strand" claims that the author list of this book is "Tom Manley", which is a partially true value of the truth in the real-world. In such a scenario, we should not regard the value provided by "Strand" as a malicious value. Instead, this value still partially supports the truth. Providing the value "Tom Manley" does not imply that this bookseller opposes the value "Pat Manley" and "Timothy B. Mihuc".*

This example illustrates that in the multi-truth problem, for a certain value set $V(o)$ claimed for object o, we should also consider the sources that provide some (not all) elements in $V(o)$ as partial contributors. Moreover, we should also assign confidence on those values that a source does not support. However, it is unfair to assign evenly distributed confidence to the values that a source supports and the values that it opposes. Given a set of values $V(o)$ claimed for object o, if a source s claims that a subset $V_s(o) \subseteq V(o)$ as the vales of o, the confidence score of value $v \in V_s(o)$ is calculated as:

$$c_s(v) = \begin{cases} (1 - \frac{|V(o) \setminus V_s(o)|}{|V(o)^2|}) \frac{1}{|V_s(o)|}, v \in V_s(o) \\ \frac{1}{|V(o)^2|}, v \notin V_s(o) \end{cases} \quad (5)$$

For a value set $V_s(o)$, the sum of the confidence score of all the value $v \in V_s(o)$ is 1, when each $c_s(v)$ for each v is in the range of $(0, 1]$. This constraint is to ensure that the confidence score of each value will not be overwhelmed.

Here is an example. Suppose the universal value set of object o is $V(o) = \{a, b, c, d\}$ and there is a value set $V_s(o) = \{a, d\}$ for o claimed by s. In this case, value b and c are assigned $\frac{1}{4^2} = \frac{1}{16}$ as their confidence score from s, while a and d are assigned $(1 - \frac{1}{16} \cdot 2) \cdot \frac{1}{2} = \frac{7}{16}$. Note that in the traditional truth discovery method, if a source only provides a and d as candidates for the truth of this object, it implies that the scores assigned to b and c are 0.

In order to address the confidence scores in the multi-truth problem, *MBM* [15] also proposes the idea to involve the unclaimed values into confidence score assignment. However, *MBM* follows the principle that with a smaller sized value set claimed by the source, a lower confidence score will be assigned to its claimed values. It may be unfair to the claimed values when the size of $V(o)$ is small. For example, when there are only 2 values in $V(o)$, the confidence score of both values will be $\frac{1}{2}$, which may lead to lower specificity

in the interactive calculation of the source trustworthiness. To strive for a balance between the claimed and unclaimed values, we try to avoid this problem by assigning a relatively lower confidence score to the values that have not been claimed by the sources and pare down the influence brought by the portion of unclaimed values.

4.2 Integrated Bayesian Model

To derive the truth, we first need to compute the probability that a value is true. Intuitively, the computation should consider the quantity of sources that support or oppose the value, as well as the trustworthiness of these sources (i.e. the probability that they provide true values) in the related domain.

We first introduce the basic idea of the proposed Bayesian model. Let $\psi(o)$ be the observation of claims from the sources for values of object o. We use $\sigma(v)$ to represent *a priori* veracity that v is true, i.e., the probability that v is true. Our target is to estimate, for each $\psi(o)$, the probability that a set of output values v is true given the observed data, $Pr(v|\psi(o))$. In Equation 6, we use Bayes rule to express $Pr(v|\psi(o))$ based on $Pr(\psi(o)|v)$ and the inverse probabilities $Pr(\psi(o)|\bar{v})$, which are the probabilities of having the observed output data when v is true or false respectively.

$$\begin{aligned} Pr(v|\psi(o)) &= \frac{Pr(\psi(o)|v)Pr(v)}{Pr(\psi(o))} \\ &= \frac{Pr(\psi(o)|v)\sigma(v)}{Pr(\psi(o)|v)\sigma(v) + Pr(\psi(o)|\bar{v})(1-\sigma(v))} \\ &= \frac{1}{1 + \frac{1-\sigma(v)}{\sigma(v)} \cdot \frac{Pr(\psi(o)|\bar{v})}{Pr(\psi(o)|v)}} \end{aligned} \quad (6)$$

We then consider representing the two conditional probabilities, $Pr(\psi(o)|v)$ and $Pr(\psi(o)|\bar{v})$, based on the trustworthiness of the sources providing or opposing value v for o. Let $S_{o^d}(v)$ denote the set of sources that provide value v for object o in domain d. Similarly, $S_{o^d}(\bar{v})$ denotes the set of sources that do not provide v for o in domain d. Let $\tau_d(s)$ be the trustworthiness of a source s, i.e., the probability that the claims made by s are true in domain d. Note that $\tau_d(s)$ is the same as the precision of s in d. Hence, we can use $\prod_{s \in S_{o^d}(v)} \tau_d(s)$ to represent the probability that the sources that support v are correct, while $\prod_{s \in S_{o^d}(\bar{v})}(1-\tau_d(s))$ represents the probability that the sources opposing v are wrong.

$$\tau_d(s) = \frac{\sum_{o' \in O^d(s)} \sum_{v \in V_s(o')} \sigma(v)}{\sum_{o' \in O^d(s)} |V_s(o')|} \quad (7)$$

$$Pr(\psi(o)|v) = \prod_{s \in S_{o^d}(v)} \tau_d(s) \prod_{s \in S_{o^d}(\bar{v})} (1-\tau_d(s)) \quad (8)$$

$$Pr(\psi(o)|\bar{v}) = \prod_{s \in S_{o^d}(\bar{v})} \tau_d(s) \prod_{s \in S_{o^d}(v)} (1-\tau_d(s)) \quad (9)$$

In a single-truth scenario, using a single metric of precision to model $\tau_d(s)$ will achieve good results, since it outputs the fact that is mostly likely to be true. However, when there are multiple truths, measuring source trustworthiness by precision cannot utilize the value of negative claims to recognize an erroneous data.

EXAMPLE 4.2. *As shown in Table 4, the author list of the third book (ISBN10: 0002557541) should be "Lee Selleck; Francis Thompson". The precision of all the 4 sources providing claims for this book is 1.0, since none of them involve false values. However, obviously, the bookseller "BookExpress" has missed one true value. Therefore, it is not applicable to evaluate the quality of sources using only precision in the multi-truth finding problem.*

To overcome the disadvantage of only considering the precision of sources, we consider involving metrics of recall and specificity to model source quality. Note that recall of source s is the probability of true values being claimed as true, while $1 - recall$ is the false negative rate. Sources with high recall tend not to miss the true values. Specificity of s is the probability of false values being claimed as false, while $1 - specificity$ is the false positive rate. Sources with high specificity tend not to involve false values.

As first introduced in [19], recall and specificity are important in the multi-truth finding problem because we are looking for sources with high recall and high precision. Therefore, we model the recall and specificity of sources as two independent quality measures to cover the complete spectrum of source trustworthiness.

The major difficulty is to estimate the recall and specificity of the data source in unsupervised learning when the truth is actually unknown. *TruthFinder* [16] derives the correctness probability of a value from the Beta distribution of the recall and false positive rate of its providers. Applying Beta distribution enforces assumptions about the generative process of the data, but when this model does not fit the actual data, the results will contradict the reality. *PrecRec* [12] computes the recall and the specificity from the training data set, which needs additional effort from supervised learning. *MBM* [15] uses predefined positive precision and negative precision to represent the initial correctness of the sources and improve the two rates in the later iterations. We decide to use predefined recall and specificity of all sources in our model, and improve the accuracy of estimation during the truth inference.

Obviously, our previous definition of trustworthiness of source s, $\tau_d(s)$, is not sufficient to describe the recall and specificity at the same time. Thus, we involve $\tau_d^{rec}(s)$ and $\tau_d^{sp}(s)$, which stand for the trustworthiness of s in recall and specificity in domain d, respectively. Let $\bar{V}_s(o)$ denote the set of values claimed for o by other sources except s. We redefine these two measures based on the veracity score $\sigma(v)$ of value v as follows:

$$\tau_d^{rec}(s) = \frac{\sum_{o \in O^d(s)} \sum_{v \in V_s(o)} \sigma(v)}{\sum_{o \in O^d(s)} |V_s(o)|} \quad (10)$$

$$\tau_d^{sp}(s) = \frac{\sum_{o \in O^d(s)} \sum_{v' \in \bar{V}_s(o)} (1-\sigma(v'))}{\sum_{o \in O^d(s)} |\bar{V}_s(o)|} \quad (11)$$

Specifically, $\tau_d^{rec}(s)$ stands for the probability that the values claimed by source s for object o are true, among all values claimed by source s for o. $\tau_d^{sp}(s)$ represents the probability that the values not claimed by source s for object o are false, among all the values that have not been claimed by source s for o. We adopt the average probability that the values provided by a source are true or false as the trustworthiness of this source.

We next propose our approach to extend the Bayesian model by involving domain expertise and multi-truth assumptions. Domain expertise $e_d(s)$ in domain d, introduced in Section 3.2, represents the heuristically estimated probability that source s provides true values in domain i. Confidence score $c_s(v)$, discussed in Section 4.1, represents the probability that value v is correct. We plug these two factors as parameters into the model. In order to avoid the elimination of factors during calculation and iteration, we model all the factors as powers of the model.

To compute the confidence score of value v, we redefine the likelihood of $\psi(o)$ under different assumptions on the correctness of v as follows:

$$Pr(\psi(o)|v) = \prod_{s \in S_{^od}(v)} \tau_d^{rec}(s)^{e_d(s)c_s(v)} \prod_{s \in S_{^od}(\bar{v})} (1-\tau_d^{sp}(s))^{e_d(s)c_s(v)} \tag{12}$$

$$Pr(\psi(o)|\bar{v}) = \prod_{s \in S_{^od}(\bar{v})} (\tau_d^{sp}(s))^{e_d(s)c_s(v)} \prod_{s \in S_{^od}(v)} (1-\tau_d^{rec}(s))^{e_d(s)c_s(v)} \tag{13}$$

Through substituting Equations 12 and 13 into Equation 6, we can derive the probability of v being true under the observation $\psi(o)$.

4.3 The Algorithm

We propose an iterative algorithm for the integrated Bayesian model. The detailed steps are presented in Algorithm 1.

Algorithm 1 DART: Integrated Bayesian Analysis in Domain-Aware Truth Finding

Input: The object sets, O; The data source sets S; The value sets $V_s(o)$ for each s in object o; The domain set D and the mapping between each object o in O and each domain d in D.
Output: $\{v|v \in V(o), \sigma(v) \geq \theta\}$ for all $o \in O$;

$\alpha, \theta, \rho \leftarrow$ default value;
for each $s \in S$ **do**
  for each $d_i \in D$ **do**
    $P_{d_i}(s) \leftarrow$ Equation 1
    $r_{d_i}(s) \leftarrow$ Equation 2
  end for
  for each $d_i \in D$ **do**
    $inf_s(d_{k|k \neq i} \rightarrow d_i) \leftarrow$ Equation 3
    $e_{d_i}(s) \leftarrow$ Equation 4
    $\tau_{d_i}^{rec}(s), \tau_{d_i}^{sp}(s) \leftarrow$ default value;
  end for
end for
for each $o \in O$ and each $s, v \in V_s(o)$ **do**
  $\sigma(v) \leftarrow$ default value;
  $c_s(v) \leftarrow$ Equation 5
end for
while uncoverage **do**
  for each $o \in O$ and each $s, v \in V_s(o)$ **do**
    $Pr(\psi(o)|v), Pr(\psi(o)|\bar{v}) \leftarrow$ Equation 12 and 13
    $\sigma(v) \leftarrow$ Equation 6
  end for
  for each $s \in S$ **do**
    for each $d_i \in D$ **do**
      $\tau_{d_i}^{rec}(s), \tau_{d_i}^{sp}(s) \leftarrow$, Equation 10 and 11
    end for
  end for
end while

We initially assign the predefined values to the parameters α, θ, ρ. The trustworthiness of the sources regarding their recall and specificity, as well as the value veracity, are also initialized with predefined values. Then we calculate the domain expertise score of each source in various domains, according to its global domain percentage, and the data richness score along with the adjusted domain expertise score in each domain. Note that the adjusted domain expertise score will remain unchanged during the whole calculation.

We then start the recursive call of the truth inference until converge. In each round, we calculate the probability of values being true as well as the trustworthiness of the sources simultaneously. We first use Equation 12 and 13 to infer the intermediate factors. Then we apply Equation 6 to update the veracity scores of values based on the trustworthiness of the sources that support or oppose them. Afterwards, we update the trustworthiness based on the new veracity scores of values they provided, based on Equation 10 and 11. Then we start a new round. The algorithm terminates when the changes of veracity score of each value remains in an interval. For each object, we pick values with a veracity score greater than θ as the output truth for this object.

We then analyze the complexity of Algorithm 1. Suppose that there are N objects and M sources, and on average there are k domains for each object and j values about each object provided by one source. In the initialization part, for each source, we calculate its domain expertise score in the k domains, which takes k time. Then we calculate the adjusted domain expertise score by involving the inference between the k domains and initialize its trustworthiness in recall and specificity, which takes $k^2 + 2k$ time. Thus, for M sources, it takes $O(k^2M + 3kM)$ time. Moreover, assigning confidence scores to all values of all sources takes $O(jMN)$ time. Thus, it takes $O(k^2M + 3kM + jMN)$ in the initialization.

In the iteration part, suppose that there are I iterations. In each iteration, for each object provided by each source in a certain domain, both calculating the confidence score of a value and inferring the veracity of the value take constant time. Therefore, $O(jMN)$ is needed to calculate the veracity of each value provided for each object. Moreover, the update of the trustworthiness of sources in recall and specificity for each source takes $O(2kM)$ time.

To summarize, the time complexity of Algorithm 1 is $O(k^2M + 3kM + jMN + jIMN + 2kIM)$, denoted by $O(MN)$, which is linear with respect to the number of objects and sources.

5. EXPERIMENTS

We first describe two real-world datasets in Section 5.1. In Section 5.2, we test the performance of the proposed model on multiple data sources, compared with the state-of-the-art approaches in truth discovery as well as baseline methods. We analyze the parameters' sensitivity in Section 5.3. Finally, we conduct experiments on synthetic dataset to test robustness of our model in extreme cases.

All the experiments presented are conducted on a server with 32GB RAM, 1.92GHz CPU, with CentOS Linux Release 7.3.1611 installed. All the algorithms including previous methods were implemented in Python 2.7.5.

5.1 Data Description

Since the information richness feature is required in our framework, none of the existing datasets satisfy the data requirements of our experiments. Thus, we prepare two real-world datasets by ourselves. We have also prepared perturbed real datasets to test the robustness of the algorithms on low-quality data as well as low-coverage data. The details of the datasets are shown below.

BOOK: We collected the *Book* dataset from *AbeBooks.com* in April 2017. It contains 54,591 different sources registered as booksellers and provides 2,338,559 listing information (i.e., bookstores selling books) for 210,206 books. Each source provides 0.000005% (1 book) to 28.7% (6,0317 books) of the whole collection. On average, each book has 4.3 different sets of authors, indicated by 12.5 booksellers. Specifically, we crawled all the book data from each bookseller to ensure the data richness property. A similar dataset crawled from *AbeBooks.com* has been released in [16] and used by other existing works [1], [12], [15], [17]. However, this dataset only includes books about *computer science*. We expanded the dataset by involving more book data in 18 categories, including

crime fiction, children's books, science fiction, horror stories, literature, arts, romance fiction, biographies, business, cookbooks, craft books, history, reference, religion, science, self-help, social science and *travel books*. We have conducted pre-cleaning on the authors' names to filter noise such as numbers and garbled codes. We select the books with conflicting information on the author lists from different data sources for validation. We randomly select 407 books and manually check the authors' names printed on the book covers to get the ground truths. In every round of the experiment part, we randomly pick 120 books as a test set, and repeat 10 times.

MOVIE: We collected the *Movie* dataset in July 2017. This dataset contains movie data from 15 different websites, including *imdb, allmovie, amazon, instantwatcher, moviefone, metacritic, movieinsider, 1moviesonline, goodfilms, dewanontons, letterboxd, filmcrave, ifcfilms, top250tv* and *agoodmovietowatch*. The dataset provides 1,134,432 listing information (i.e. source providing a movie) for 468,607 movies. The genres of these movies include *action, adventure, animation, biography, children, comedy, crime, documentary, drama, faith, family, fantasy, history, horror, music, romance, science-fiction, sports, thriller, war* and *western* (21 in total). The release year is from 1900 to 2017. Again, we crawled all the movie data from each data source to ensure the data richness property. We use this dataset to infer the true answers for the *directors* attribute. In the validation part, we pick up the movies mentioned by at least two data sources. Thus, there are in total 16,955 movies for validation experiments on the director attribute. On average, each movie has 2.32 different sets of directors provided by 3.25 websites. Similar movie datasets have also been used in [15] [19] for experimental validation of the multi-truth problem. However, those datasets do not guarantee the data completeness of each source. Thus, we prepare the dataset by ourselves. We have carried out the same pre-cleaning task on the movie dataset as we have done for the book dataset. We randomly select 210 movies and label the directors from the movie posters to get the ground truths. In every round of the experiment, we randomly pick 80 movies as test set, and repeat 10 times.

5.2 Model Performance Validation

5.2.1 Baselines and Metrics

We compare our models with several state-of-the-art techniques together with voting strategies. We briefly summarize them as follows, and refer readers to the original publications for details.

Majority Voting regards a value as true if the proportion of sources that claim the value is the largest.

TruthFinder [16] considers the positive claims only, and for each object, it calculates the probability that at least one positive claim is correct using the precision of the sources.

AccuSim [1] applies Bayesian analysis to iteratively detect dependence between sources and discover the truth from conflicting information. It also considers the accuracy of sources and the similarity between values during the truth finding process.

LTM [19] constructs a graphical model and uses Gibbs sampling to determine the source quality and truthfulness of each value provided for an object. It considers two types of errors under the scenarios of multiple truths: false positive and false negative.

MBM [15] leverages the unique mapping features of sources and values to reformulate the multi-truth finding problem. It uses the mutual exclusion of the values to reflect the inter-value implications. It also breaks the source reliability into two parameters, one for the false positive error and the other for the false negative error.

Weighted Voting uses the adjusted domain expertise score of each source as its weight to calculate the total voting score of each value. The value set with the highest score is chosen as true.

Domain-Separate TruthFinder and **Domain-Separate AccuSim** separate each source across its domains and treat them as different sources, then employ *TruthFinder* [16] and *AccuSim* [1] to infer the truth. Both approaches serve as natural baselines.

DART is our **D**omain-**A**wa**R**e **T**ruth Discovery model. DART incorporates the domain expertise score and the confidence score definition for truth determination.

DARTinf additionally involves the influence between pairs of domains when inferring the domain expertise score.

DARTinf-Top only selects those data sources with the highest domain expertise score to be involved based on DARTinf. In the Books dataset, we select the top 20% of sources with the highest score as the contributors, while in the Movies dataset, we select the top 50% as the contributors. We examine the impact of selecting a different portion of top-contributors in Section 5.3.

We excluded the comparison with several methods that are not applicable in our multi-truth finding problem. The fine-grained truth discovery algorithms proposed in [8] [2] [20] are designed for crowdsourcing tasks, where the major effort is emphasized in inferring the topical knowledge from the task text description information. In [12], computing the recall of a data source relies on knowledge of the set of true triples, and the numbers of true triples provided by sources are learned from a training dataset, which is semi-supervised. Moreover, [18] and [5] target on heterogeneous data fusion, while our method focuses on categorical data.

To ensure a fair comparison, parameters for the algorithms above are set according to the optimal settings suggested by their authors. For our method, α is set to be 1.5 in book dataset, and remains as 1 in movie dataset. The reason is that there are a large number of data sources (i.e., 54,591) in the book dataset, and the percentages of books provided by different sources are unevenly distributed. There are a large quantity of sources (i.e., 23%) with very small global domain percentage (i.e., smaller than 0.01%). Thus, we need to use an adjusted factor to emphasize the influence of these parts of data. Otherwise their contribution will be eliminated.

Moreover, in our experiments, we set $\rho = 0.2$ for the book dataset, and $\rho = 0.3$ for the movie dataset. The reason is that in the book dataset, 12.49% books has two or more categories labeled by one source, while 39.11% movies has two or more genres labeled by one source in the movie dataset. Since the overlaps between domains occur more frequently in movies dataset than in books dataset, ρ for movies should be set higher. We show the experimental results of changing ρ when inferring the domain expertise of sources in various datasets in Section 5.3.

We initialize the $\tau_d^{rec}(s)$ and $\tau_d^{sp}(s)$ for each source in every domain as 0.8 and 0.9 in book dataset, and 0.9 and 0.9 in movie dataset respectively. The parameter sensitivity is also studied in Section 5.3. Without supervised training, we set the threshold confidence score θ to 0.5. We have also set the *a priori* veracity score $\sigma(v)$ of each value v to 0.5.

5.2.2 Comparison of Truth-Finding Methods

Table 6 shows the performance of different algorithms on the two datasets in terms of precision, recall and F-measure. Our algorithm achieves the best recall and F-measure among all the compared methods on both datasets. Our methods also achieve rather high precision, when DARTinf-Top achieves the best precision in Movie-directors. Note that DARTinf performs better than DART in both datasets, which demonstrates the importance of involving inference between domains. DARTinf-Top performs even better by neglecting the sources with less expertise in certain domains.

Table 6: Comparison of different algorithms on the three datasets. Our algorithms are conducted on the category/genre attribute. Precision measures among the returned true triples, how many are indeed true; recall measures among the provided true triples, how many are returned. F-measure computes their harmonic mean (i.e. $F1 = \frac{2*prec*rec}{prec+rec}$). The best, second best and the third best performance values are in bold.

Methods	*Book-author dataset*			*Movie-director dataset*		
	Precision	*Recall*	*F1*	*Precision*	*Recall*	*F1*
Majority Voting	0.9024	0.7400	0.8132	0.9127	0.8190	0.8633
TruthFinder	**0.9048**	0.8302	0.8659	0.9171	0.9018	0.9093
AccuSim	0.8545	0.6996	0.7693	**0.9336**	0.8709	0.9012
LTM	0.8850	0.8805	0.8827	0.9185	0.8904	0.9042
MBM	0.8400	**0.9322**	0.8892	0.7813	0.9344	0.8510
Weighted Voting	**0.9175**	0.7902	0.8491	0.9039	0.8078	0.8531
Domain-Separate TruthFinder	**0.9063**	0.8339	0.8686	0.9224	0.9008	0.9115
Domain-Separate AccuSim	0.8567	0.7053	0.7736	**0.9383**	0.8880	0.9124
DART	0.8750	**0.9319**	**0.9025**	0.9107	**0.9717**	**0.9402**
DARTinf	0.8762	**0.9306**	**0.9026**	0.9307	**0.9625**	**0.9463**
DARTinf-Top	0.8777	**0.9322**	**0.9041**	**0.9414**	**0.9625**	**0.9518**

Majority Voting achieves lower recall on both datasets. The reason is that most sources tend to provide single truth for the requested object. On the other hand, Majority Voting achieved relatively high precision on both datasets, since both datasets have multiple claims on each object and therefore the values with majority votes are very likely to be true.

In addition, AccuSim does not perform well on both datasets, where its recall is rather low. The reason is that we choose conflicting data for the experiment, where most of the books and movies have two to three truths. Therefore, the multi-truth problem could not be well addressed by algorithms aimed at solving single truth. In addition, both TruthFinder and LTM achieve relatively high precision, especially in the movie-director dataset where there is less conflicting information. The recall of these methods are relatively lower when compared with MBM and DART, since both MBM and DART take the mutual exclusive relation of values into consideration when assigning the confidence scores. Although the precision of MBM is lower, its recall is higher than other state-of-the-art techniques. The resaon is that MBM gives a high confidence score to the unclaimed values of the sources and can easily detect more potential truths. However, the drawback is that false positive counts will be raised since some false values are also included. We avoid this problem by adjusting the confidence score of each value. DART will not assign a heavy score to the unclaimed values. Thus, the false positive rate is lower and DART achieves higher precision.

Moreover, both of the baseline methods, Domain Separate TruthFinder and Domain Separate AccuSim, perform slightly better than the original methods TruthFinder and AccuSim. The reason is that these two baseline methods take domain difference into consideration. It indicates that the same source will perform differently on different domains. Thus, the performance will be improved if we consider different domains of the same source separately. However, this naive approach does not achieve high gain, since it will reduce the number of answers dramatically on each domain, especially when the quantity of domains is large. Since the data are insufficient, it will lead to an incorrect estimation of source expertise, and hence the overall performance of truth inference will drop. Faitcrowd [8] also discusses the limitations caused by data insufficiency in this method. DART outperforms these two baseline methods by taking the data richness in different domains into consideration. It tends to trust the sources with richer data in certain domains and thus avoids the negative impact that comes from the malicious data providers with low domain expertise. DARTinf further neglects these drawbacks by considering the possible correlations between domains, and hence increases the amount of data that contributes to the inference of source trustworthiness. Involving the correlations between domains makes the expertise scores of sources more reasonable and closer to the real-world.

Table 7: The performance comparison of domain-aware algorithms on the attribute *published year/release year* of books and movies. In both datasets, the domains are classified as *before1920, 1921-1940, 1941-1960, 1961-1980, 1981-2000, 2001-now*. WV stands for Weighted Voting. DS-TF and DS-Accu stands for Domain-Separate TruthFinder and Domain-Separate AccuSim.

Methods	*Book-author dataset*			*Movie-director dataset*		
	Precision	*Recall*	*F1*	*Precision*	*Recall*	*F1*
WV	0.9098	0.7716	0.8350	0.9135	0.7919	0.8483
DS-TF	0.9063	0.8339	0.8686	0.9211	0.9118	0.9164
DS-Accu	0.8567	0.7053	0.7736	**0.9367**	0.8861	0.9107
DART	0.8810	**0.8983**	**0.8896**	0.9082	**0.9717**	0.9389
DARTinf-Top	**0.9084**	**0.8983**	**0.9033**	0.9210	**0.9717**	**0.9457**

To summarize, the experimental results show that our model has significantly reduced the error rate compared with other methods.

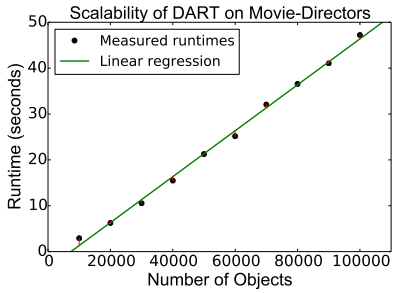
5.2.3 Performance Evaluation on Different Attributes

In order to investigate the impact of choosing different attributes to classify the domains, we have also conducted another experiment to study the performance of DART and DARTinf-Top on the *published year/release year* attribute on book-authors and movie-directors datasets. As demonstrated in Table 7, when it comes to the domain division in attribute *published year* and *release year*, the results are also promising.

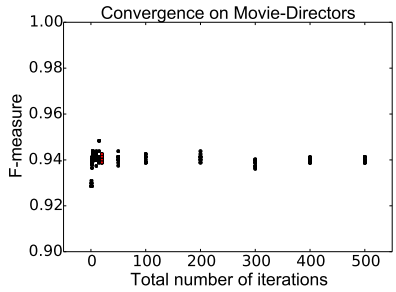
Most of the methods achieve slightly lower precision on book-authors dataset.The major reason is that values of *published year* of books in our datasets are not evenly distributed. More than 87% books are published after 1981. Only very few booksellers have high domain expertise for books published before 1980, and thus the results could be easily misled by the malicious results of these sources. On the contrary, in the movie dataset, domains enjoy more uniform distribution in *released year*. For instance, over 80% of sources provide movies in the domain "1921 to 1940". In this case, truth finding can take advantage of relying more on the real expertise on the specified domain. This experimental result implies that domain-aware algorithms may achieve better results on those attributes with domain values more uniformly distributed.

5.2.4 Efficiency

We also examine the execution time of each algorithm, as listed in Figure 4(c). We create 5 small datasets by randomly sampling



(a) Measurement of the runtime for 10 iterations of DART for varying numbers of objects. The included linear regression enjoys an exceptional goodness-of-fit of $R^2 = 0.6693$



(b) Performance evaluation of DART under different number of iterations. We initialize the $\tau_d^{rec}(s)$ and $\tau_d^{sp}(s)$ for each source in every domain as 0.9 and 0.9 respectively.

	Runtime (seconds) vs. #Objects				
#Objects	***2000***	***4000***	***6000***	***8000***	***10000***
Majority Voting	0.183	0.361	0.480	0.705	1.113
Weight Voting	0.345	0.561	0.978	1.812	2.455
TruthFinder	1.249	4.421	10.012	18.052	28.539
Accusim	2.376	9.107	22.634	38.000	52.823
LTM	2.505	12.035	32.092	55.823	82.952
MBM	**0.593**	**1.008**	**1.509**	**2.019**	**2.541**
DART	**0.636**	1.155	**1.704**	**2.292**	2.985
DARTinf	**0.644**	**1.151**	**1.706**	**2.291**	**2.955**
DARTinf-Top	0.652	**1.132**	1.725	2.335	**2.912**

(c) The execution time of all algorithms (in seconds) for the movie datasets. We have executed each algorithm 10 times and use the average running time as the final record.

Figure 4: Efficiency Measurement

2K, 4K, 6K, 8K and 10K movies from the entire 111,987 movies and selecting all facts associated with the sampled movies. The results show that all algorithms, except LTM, have comparable execution time, while LTM is more sensitive to the data scale. DART takes slightly more time than MBM, which is the fastest one. We believe that it is acceptable because DART needs to search for source expertise in various domains during the truth inference, which may take a while. Specifically, there is only a very slight difference between the execution time of DART, DARTinf and DARTinf-Top, indicating that calculating intra-domain influence and picking up expertized sources does not take further time.

To further verify that DART runs linearly with respect to the number of objects, we perform linear regression on the running time as a function of dataset size, as shown in Figure 4(a). It yields an exceptional goodness-of-fit R^2 score of 0.6693, which demonstrates the scalability of DRAT.

Our truth inference approach is an iterative algorithm, thus, we also need to know how many iterations it requires to reach a satisfying F-measure. As illustrated in Figure 4(b), we examine the algorithm performance after 1, 2, 3, 5, 10, 15, 20, 50, 100, 200, 300, 400, 500 iterations on 100 randomly generated objects with labels. We repeat it 10 times to account for randomization due to sampling. The result indicates that at 20 iterations, the algorithm achieves an optimal F-measure, which is around 94%, with extremely low variance. Additional iterations will not further improve its performance, thus we can conclude that DART converges quickly with a small number of iterations.

5.3 Parameter Sensitivity

We also explore the impact of the different parameter settings of our algorithm. As shown in Figure 5(c) and 5(d), the setting of the initial default values of τ^{sp} will not seriously affect the performance. However, Figure 5(a) and 5(b) indicates that it is better to set τ^{rec} for each source in range(0.5, 0.9), otherwise the performance will be significantly affected.

We also examine the impact brought by the variance of the parameter ρ, which controls the influence of related domains in Equation 3. Larger ρ implies that the effect from similar domains is more influential. We conduct the control experiment in both books data and movie data. The results show that the F-measure of the algorithm will not benefit much from the domain-dependency analysis if ρ is set too small. The reason is that a small ρ neglects the influence from correlated domains even though there are big overlaps between domains. On the other hand, a large ρ may mix up the correlated domains, leading to blurred boundaries between domains and hence weaken the benefits of domain expertise classification. However, the overall impact of different values of ρ is still limited. Figure 5(e) and 5(f) show that the precision and recall of the algorithm varies by no more than 5% if ρ does not change fiercely.

In addition, we examine the performance of DARTinf-Top when involving different numbers of sources with highest domain expertise scores to determine the truths. As shown in Figure 5(g), for the book-authors dataset, F1 achieves the highest when we select the top 20% sources with the highest scores as contributors, while in the movie-director dataset, F1 is comparably high when we choose the top 50% to 60% sources. The most suitable portion of sources that should participate in truth finding depends on the quantity of sources. Since on average there are around 3,000 different sources in one domain of the book dataset, filtering out more unprofessional sources will contribute to the quality of the final output. Thus top 10% to 20% sources perform better in this situation. However, in the movie dataset there are much fewer sources, thus we can involve more top sources, e.g., the top 50% to 60%, to improve the F-measure.

5.4 Experiments on Synthetic Data

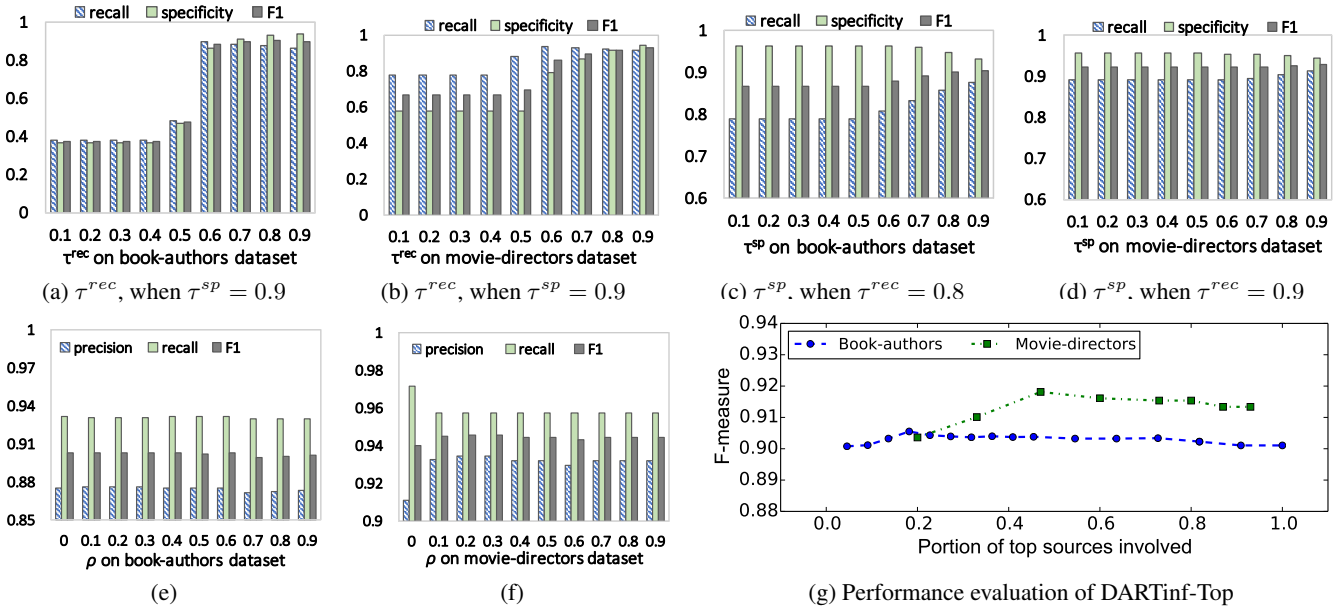
We have also conducted experiments on perturbed real datasets to examine the performance of our model in extreme cases.

5.4.1 Data with Low Overall Quality

In order to better study the applicability of DART in real-world datasets, we conduct two experiments on datasets with low overall quality sources. The first one is conducted on **book-authors dataset and movie-directors without data cleaning.** Without the data cleaning procedure, 4.4% of the book data and 2.9% of the movie data contain noise, such as numbers and garbled codes. As shown in Figure 6(a) and 6(b), the performance of all the algorithms degrades, especially in precision. However, DART, DARTinf and DARTinf-Top still keep the best performance. Note that although DARTinf-Top does not gain much improvement as it achieves in the clean dataset, it still beats all the priori techniques.

The second one is conducted on **book-authors dataset with 0.1% to 10% noise added.** We insert randomly generated 0.1% to 10% noise into the facts provided by each source and compare the results. As illustrated in Figure 6(c), DARTinf-Top still beats all others in all cases. To investigate the performance, we plot the changes in the precision and recall of DART and DARTinf-Top in Figure 6(d).

The results of both experiments indicate that all algorithms may lose a little advantage with dirty data. Note that noise will not significantly affect the performance of algorithms that assume single-



(a) τ^{rec}, when $\tau^{sp} = 0.9$ (b) τ^{rec}, when $\tau^{sp} = 0.9$ (c) τ^{sp}, when $\tau^{rec} = 0.8$ (d) τ^{sp}, when $\tau^{rec} = 0.9$

(e) (f) (g) Performance evaluation of DARTinf-Top

Figure 5: Performance evaluation regarding parameter sensitivity. The experiments in conducted in book-authors dataset and movie-directors dataset, both rely on the domain classification on the category/genre attributes.

truth, such as Majority Voting and AccuSim. The reason is that the value with the maximum number of supporters is still very likely to be true. Hence, the precision of these methods will not be affected. However, the precision of the algorithms that assume multi-truth, such as TruthFinder, LTM, MBM and our model, is more easily affected by noise. With more noise, the precision of these algorithms drops, since they tend to involve more possible values as final output. Specifically, DARTinf-Top outperforms DART in precision, since it only considers the values provided by top-sources, instead of all the values. Hence, the probability that DARTinf-Top is affected by malicious data is lower compared with that of DART. Nevertheless, the recall of DART and DARTinf-Top keeps rather high despite the noise, because noise will not affect the selection of the possible true values. Therefore, DARTinf-Top still outperforms other techniques in F- measure.

5.4.2 *Noise in Sources with High Domain Expertise*

We also examine DARTinf-Top when there are malicious data in the sources with high domain expertise scores.

We use **book-authors dataset, with 1% to 30% noise added to facts provided by top 10% sources with the highest domain expertise scores.** For each object o^d, we randomly add noise to the facts provided by top 10% sources with the highest $e_d(s)$. As demonstrated in Figure 6(e), the precision of DARTinf-Top slightly drops when more noises are added. The result indicates that the performance of DARTinf-Top will degrade when there is more noise in the sources with higher domain expertise scores. Note that more noise will affect the precision of our model, since it tends to involve more values as truth and has higher chance to mistakenly include malicious data. Nevertheless, there is no significant change in the recall of DARTinf-Top, which proves that noise will not affect the selection of possible true values in our model. Therefore, the performance of DARTinf-Top in F-measure is still promising.

5.4.3 *Data with Low-Coverage Sources*

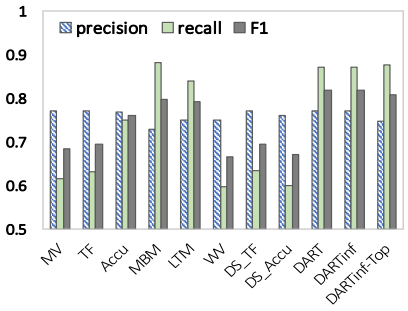
In our model, the inference of domain expertise scores is based on domain coverage. High domain expertise score implies high domain coverage. Our model relies more on the sources with dominating domain coverage to infer truth. However, in some cases, certain domains may not have sufficient information. Therefore, we also need to examine the performance of our methods with such a lower coverage setting.

We conduct experiments on **book-authors dataset, with objects provided by sources in different range of domain expertise scores.** We first rank the sources according to their domain expertise scores in descending order. We then divide the sources into 10 groups. In each group, we randomly select 100 sources that provide values for the objects with ground truths. We conduct DART and DARTinf-Top over these sources and objects. Figure 6(f) illustrates the experimental results. Specifically, the number N on x-axis represents group N. Group N is the top $(N-1) \times 0.1$ to $N \times 0.1$ sources after the ranking. For example, group 2 means the top 10% to 20% sources with the highest domain expertise scores.

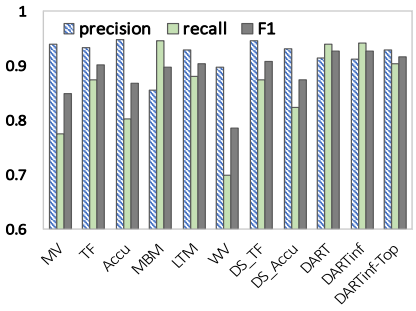
The results show that the precision of our model will not be significantly affected even with the sources of low domain expertise scores. However, the recall of our model will decrease if we only involve sources with low domain coverage. The reason is that sources with low domain coverage usually only provide single value for objects (e.g., first author of a book), which contributes to overall precision but reduce the recall. The recall of DARTinf-Top is more sensitive to domain coverage, since it relies more on sources with high coverage. We believe that the result in lower coverage setting is still acceptable, since the high precision contributes to the F-measure and makes it promising.

6. RELATED WORK

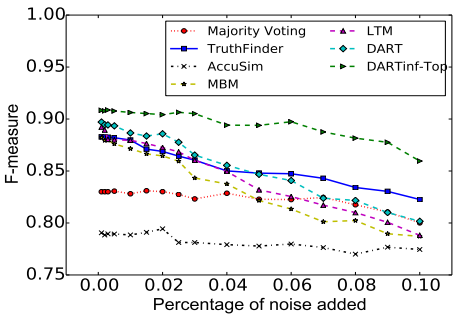
There has been extensive work in the area of data fusion, targeting on resolving conflicts and finding the truth. *TruthFinder* [16] was the first work to formally formulate the truth finding problem and propose a Bayesian based algorithm that iteratively infers source quality and truth. Most of the truth discovery methods, such as [9] [1] [3] [5] [18] [4], assume that there is only one truth for each fact provided by the data source. Based on this assumption, the most trustworthy information is selected as truth. There are also other works, such as *LTM* [19], *PrecRec* [12], *MBM* [15], proposed to solve the multi-truth problem. *2-Estimates* [3] adopts



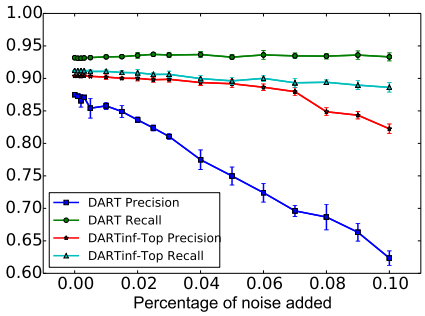
(a) Experiments on the unclean book-authors dataset



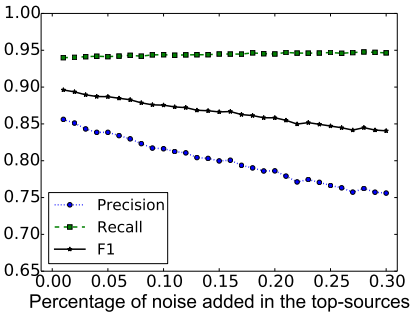
(b) Experiments on the unclean movie-directors dataset



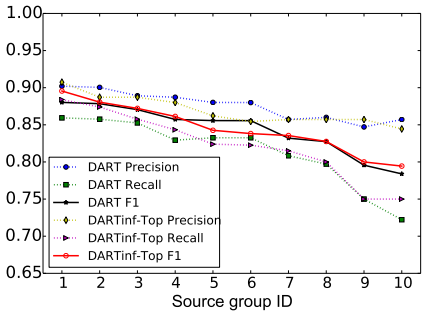
(c) Performance comparison of different algorithms when the data quality is low.



(d) Performance of DART and DARTinf-Top with 0.1% to 10% noise added in the data.



(e) Performance of DARTinf-Top, with 1% to 30% noise added in facts provided by top 10% sources



(f) Performance of DART and DARTinf-Top on sources with various domain expertise scores

Figure 6: Performance comparison of different algorithms with synthetic datasets. All experiments are repeated 10 times. Specifically, in (f), the number N on x-axis represents group N. Group N is the top $(N-1) \times 0.1$ to $N \times 0.1$ sources after being ranked according to domain expertise scores in descending order. For example, group 2 means top 10% to 20% sources with highest domain expertise scores.

complementary vote to involve multiple possible truths, while *3-Estimates* [3] augments *2-Estimates* by considering the difficulty of getting the truth about each object. In addition, *LTM* [19] aims at discovering multiple truths by applying a probabilistic graphical model. It breaks source quality into two factors, a false positive and a false negative, in order to better model the multi-truth problem. *PrecRec* [12] and *MBM* [15] also consider calculating both the precision and recall of sources to satisfy the multi-truth assumption.

There are also current efforts in the truth inference problem in the crowdsourcing area [21] [8] [2]. Some of them also try to utilize the fine-grained reliability of sources. *FaitCrowd* [8] employs a probabilistic graphical model to divide tasks into topical-level clusters and estimate a source's topical reliability accordingly. However, the number of topics is predefined and the major limitation lies in the lack of semantics of the topic clusters. A similar method is also used in *iCrowd* [2]. Similarity metrics and topic models are applied to obtain the similarity and topic distribution for each micro task. Tasks with large text similarity have a higher chance of being classified into the same domain. Nevertheless, it may lead to the wrong domain classification, since similar sentences may focus on different domains. *DOCS* [20] is another work that also clusters the tasks into different domains. It consults an existing knowledge base to obtain domain information based on the task text description. However, the detailed information of unusual data may not be included in the knowledge base.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we study the problem of discovering multiple truths for a data item from conflicting sources in various domains. We investigate the correlations between domain expertise and domain data richness of the data sources. We also define and apply the influence between different domains from the same data source to determine the data expertise of the sources. Specifically, we leverage the unique features of the multiple-truth problem, which emphasizes that sources may provide partially correct values of a data item, to determine the confidence score of each value set provided by various data sources. We propose an integrated Bayesian approach, which comprehensively incorporates the domain expertise of the data source and confidence score of the value, to infer multiple possible truths of a data item. Experiments on two real-world datasets demonstrate the effectiveness of our approach.

There are still several interesting challenges in this problem. Our approach outperforms other state-of-the-art algorithms in multi-truth-finding problems, since we have involved domain expertise, which is determined by information richness, to infer source quality. However, the quantity of data may change from time to time. Thus, we consider modifications of our model to account for the data updates in future work. Another challenge is to better estimate the influence between domains. Currently we only consider the influence between pairs of domains. In future, we will involve more complicated triple correlations, such as triangular relations, to obtain more accurate domain expertise of data sources.

Acknowledgments

This work is supported in part by the Hong Kong RGC Project 16202215, Science and Technology Planning Project of Guangdong Province, China, No. 2015B010110006, NSFC Grant No. 61729201, 61232018, Microsoft Research Asia Collaborative Grant and NSFC Guang Dong Grant No. U1301253.

8. REFERENCES

[1] X. L. Dong, L. Berti-Equille, and D. Srivastava. Integrating conflicting data: the role of source dependence. *PVLDB*, 2(1):550–561, 2009.

[2] J. Fan, G. Li, B. C. Ooi, K.-l. Tan, and J. Feng. icrowd: An adaptive crowdsourcing framework. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, pages 1015–1030. ACM, 2015.

[3] A. Galland, S. Abiteboul, A. Marian, and P. Senellart. Corroborating information from disagreeing views. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 131–140. ACM, 2010.

[4] Q. Li, Y. Li, J. Gao, L. Su, B. Zhao, M. Demirbas, W. Fan, and J. Han. A confidence-aware approach for truth discovery on long-tail data. *PVLDB*, 8(4):425–436, 2014.

[5] Q. Li, Y. Li, J. Gao, B. Zhao, W. Fan, and J. Han. Resolving conflicts in heterogeneous data by truth discovery and source reliability estimation. In *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*, pages 1187–1198. ACM, 2014.

[6] X. Li, X. L. Dong, K. Lyons, W. Meng, and D. Srivastava. Truth finding on the deep web: Is the problem solved? *PVLDB*, 6(2):97–108, 2012.

[7] X. Liu, X. L. Dong, B. C. Ooi, and D. Srivastava. Online data fusion. *PVLDB*, 4(11):932–943, 2011.

[8] F. Ma, Y. Li, Q. Li, M. Qiu, J. Gao, S. Zhi, L. Su, B. Zhao, H. Ji, and J. Han. Faitcrowd: Fine grained truth discovery for crowdsourced data aggregation. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 745–754. ACM, 2015.

[9] J. Pasternack and D. Roth. Knowing what to believe (when you already know something). In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 877–885. Association for Computational Linguistics, 2010.

[10] J. Pasternack and D. Roth. Making better informed trust decisions with generalized fact-finding. In *IJCAI*, volume 11, pages 2324–2329, 2011.

[11] J. Pasternack and D. Roth. Latent credibility analysis. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1009–1020. ACM, 2013.

[12] R. Pochampally, A. Das Sarma, X. L. Dong, A. Meliou, and D. Srivastava. Fusing data with correlations. In *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*, pages 433–444. ACM, 2014.

[13] G.-J. Qi, C. C. Aggarwal, J. Han, and T. Huang. Mining collective intelligence in diverse groups. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1041–1052. ACM, 2013.

[14] V. Vydiswaran, C. Zhai, and D. Roth. Content-driven trust propagation framework. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 974–982. ACM, 2011.

[15] X. Wang, Q. Z. Sheng, X. S. Fang, L. Yao, X. Xu, and X. Li. An integrated bayesian approach for effective multi-truth discovery. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 493–502. ACM, 2015.

[16] X. Yin, J. Han, and S. Y. Philip. Truth discovery with multiple conflicting information providers on the web. *IEEE Transactions on Knowledge and Data Engineering*, 20(6):796–808, 2008.

[17] X. Yin and W. Tan. Semi-supervised truth discovery. In *Proceedings of the 20th international conference on World wide web*, pages 217–226. ACM, 2011.

[18] B. Zhao and J. Han. A probabilistic model for estimating real-valued truth from conflicting sources. *Proc. of QDB*, 2012.

[19] B. Zhao, B. I. Rubinstein, J. Gemmell, and J. Han. A bayesian approach to discovering truth from conflicting sources for data integration. *PVLDB*, 5(6):550–561, 2012.

[20] Y. Zheng, G. Li, and R. Cheng. Docs: a domain-aware crowdsourcing system using knowledge bases. *PVLDB*, 10(4):361–372, 2016.

[21] Y. Zheng, G. Li, Y. Li, C. Shan, and R. Cheng. Truth inference in crowdsourcing: is the problem solved? *PVLDB*, 10(5):541–552, 2017.