Published in final edited form as:

KDD. 2015 August ; 2015: 675–684. doi:10.1145/2783258.2783277.

On the Discovery of Evolving Truth

Yaliang Li[1], **Qi Li**[1], **Jing Gao**[1], **Lu Su**[1], **Bo Zhao**[2], **Wei Fan**[3], and **Jiawei Han**[4]

Yaliang Li: yaliangl@buffalo.edu; Qi Li: qli22@buffalo.edu; Jing Gao: jing@buffalo.edu; Lu Su: lusu@buffalo.edu; Bo Zhao: bo.zhao.uiuc@gmail.com; Wei Fan: fanwei03@baidu.com; Jiawei Han: hanj@illinois.edu

[1]SUNY Buffalo, Buffalo, NY USA

[2]LinkedIn, Mountain View, CA USA

[3]Baidu Big Data Lab, Sunnyvale, CA USA

[4]University of Illinois, Urbana, IL USA

Abstract

In the era of big data, information regarding the same objects can be collected from increasingly more sources. Unfortunately, there usually exist conflicts among the information coming from different sources. To tackle this challenge, truth discovery, i.e., to integrate multi-source noisy information by estimating the reliability of each source, has emerged as a hot topic. In many real world applications, however, the information may come sequentially, and as a consequence, the truth of objects as well as the reliability of sources may be dynamically evolving. Existing truth discovery methods, unfortunately, cannot handle such scenarios. To address this problem, we investigate the temporal relations among both object truths and source reliability, and propose an incremental truth discovery framework that can dynamically update object truths and source weights upon the arrival of new data. Theoretical analysis is provided to show that the proposed method is guaranteed to converge at a fast rate. The experiments on three real world applications and a set of synthetic data demonstrate the advantages of the proposed method over state-of-the-art truth discovery methods.

Keywords

Truth Discovery; Source Reliability; Dynamic Data

1. INTRODUCTION

Nowadays, with information explosion, it becomes much more convenient to collect information from multiple places. For example, to know the weather condition of a specific location, we can get the information from multiple weather services; to get the up-to-date

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*Data mining*

information about some stocks, multiple websites are recording the real-time information; to query the flight status, multiple agencies may provide such information. However, these collected information could conflict with each other.

To better utilize the collected multi-source information, an important task is to resolve the conflicts among them, and output the trustworthy information. In the light of this challenge, truth discovery [3,5,7–9,14,17,21,23,25] is emerging as a promising paradigm that can help people identify trustworthy information from multiple noisy information sources, and it has been applied in various application domains [4, 11, 18, 19]. In contrast to voting or averaging approaches that treat all sources equally, truth discovery methods estimate the source reliability and infer the trustworthy information simultaneously. As the source reliability degrees are usually unknown *a priori*, in truth discovery, source reliability estimation and trustworthy information inference are tightly combined by the following principle: If a source provides trustworthy information more often, it will be assigned a high reliability; Meanwhile, if one piece of information is claimed by high quality sources, it will be regarded as trustworthy information.

Most of the existing truth discovery algorithms are proposed to work on static data. They can not handle the scenarios where the collected information comes sequentially, which happen in many real world applications. Consider the aforementioned applications: The weather condition, the stock information, and the flight status are collected in real-time. These applications reveal the necessity to develop truth discovery methods for such scenarios.

In the scenarios that information is collected continuously, new challenges are brought by the nature of such applications. First, as data comes sequentially from multiple sources in a dynamic environment, we cannot afford to re-run the batch algorithm at each timestamp. Instead, we need approaches that scan data once and conduct real-time truth discovery to facilitate processing and storage on large-scale data. Unfortunately, existing truth discovery approaches are not developed to handle dynamic streaming data. They work in batch processing manner, and cannot incrementally update truth and source reliability.

Second, unique characteristics of dynamic data are observed across various real world applications: 1) The true information of objects evolves over time, and for a specific object, the temporal smoothness exists among its information at different timestamps. 2) The observed source reliability changes over time, which is not consistent with the assumption held by existing approaches as they assume the source has unchanged reliability. In Section 2, we will illustrate more details about these observations and discuss the difficulties they bring to truth discovery.

To tackle the aforementioned two challenges, in this paper, a new truth discovery method is developed for dynamic scenarios. We first propose an effective solution that can update truth and source reliability in an incremental fashion. Thus the proposed method can work in real-time, and the data only needs to be visited once. To capture the unique characteristics of dynamic data, two factors, namely, smoothing factor and decay factor, are incorporated into the proposed approach to model the evolution of both truth and source reliability. Further, we give theoretical analysis of the proposed method, and show its fast rate of convergence.

To demonstrate the effectiveness and efficiency of the proposed method, we conduct a series of experiments on three real world applications and a set of synthetic datasets. By comparing with state-of-the-art truth discovery methods, the improvement brought by the proposed method is justified. We also analyze the effect of the smoothing factor and decay factor, explain how these factors can capture the characteristics of dynamic data, and test the sensitivity of the proposed method with respect to these factors.

In summary, our contributions in this paper are:

- Motivated by many real world applications, we study the truth discovery task under dynamic scenarios, in which the information is collected continuously, and evolution exists in both object truths and source reliability.
- We develop an incremental truth discovery method that can be applied in real-time scenarios. Two more factors are incorporated into the proposed method to capture the characteristics of dynamic data.
- Theoretical analysis is presented to prove the convergence of the proposed method. The rate of convergence is also given.
- We test the proposed method on three real world applications and several synthetic datasets, and the improvements on both performance accuracy and efficiency are demonstrated.

In the following sections, we first present the observations and challenges under dynamic scenarios in Section 2. In Section 3, after formally defining the task, the proposed solution is derived, and further the smoothing factor and decay factor are incorporated. In Section 4, we give theoretical analysis on the proposed method. Section 5 shows the experiments we conduct to validate the effectiveness and efficiency of the proposed method. We discuss related work in Section 6, and conclude the paper in Section 7.

2. OBSERVATIONS

As mentioned before, in this section, we explore and summarize common evolutionary patterns observed across various applications. Later in this paper we will present our solutions that are motivated by these patterns. In the following, we use three truth discovery tasks, i.e., weather forecast, stock records, and flight status integration, to illustrate the effect of dynamic changes on truth discovery. These three datasets have been used before in truth discovery literature [7, 8, 26] as static data, but they all involve dynamically changing data. Specifically, we are interested in getting true answers for weather forecast of cities, real-time recording of stocks, and status of flights by merging data that are continuously collected from multiple websites. More details and experimental results on these datasets can be found in Section 5.

There are two major observations regarding the impact of dynamic data on truth discovery: 1) Truth is evolving but temporal smoothness is observed, and 2) source reliability changes, which differs from the assumption held by truth discovery approaches applied to static settings.

Truth Evolution

In Figures 1, we demonstrate the evolution of truths (x-axis denotes time, and y-axis denotes the true value that we are interested in: temperature, market capitalization, or flight arrival time). Figure 1a shows the highest temperature of New York City during a period of two months, Figure 1b illustrates stock information over one month (weekdays only), and Figure 1c is about the arrival time of a particular flight on each day of one month (arrival time is translated into minutes from 12am, for example, 07: 30am is translated into 450 mins). From these figures, we can observe that the value is constantly changing but the change within a small time window is smooth. For example, in temperature data, if today's highest temperature is 42F, it is more likely that the highest temperature tomorrow will not deviate much from 42F. Similar patterns can be observed on stock data. For flight data, as the scheduled arrival time for a flight is almost the same, for most of the days, the actual arrival time is around the arrival time of the previous ones (temporal smoothness). Only a few exceptions are observed in which arrival time is quite different from the scheduled arrival time (the peaks in Figure 1c). These exceptions will be discussed and analyzed in experiments (Section 5).

Source Reliability Evolution

Truth discovery approaches can infer trustworthy information from conflicting multi-source data as it takes source reliability into consideration. To estimate such source reliability, existing approaches make the assumption that each source's reliability is consistent over all the claims it makes. This assumption is made in [22] and further adopted by [3, 5, 7, 8, 14, 23, 25], and it works well in static settings where all the claims are processed simultaneously. However, this assumption does not hold any more in dynamic environment. Figure 2 demonstrates how sources' reliability changes over time on the three datasets we adopt. The reliability of a source is quantified by comparing the sources' claims with the true values and measuring the closeness between them. An interesting observation is that source reliability fluctuates around a certain value, which may correspond to the underlying true source reliability. However, at different timestamps, the observed source reliability reflects the effect of both the underlying source reliability and some other factors that are "local" to each timestamp. For example, a source that usually provides accurate flight information may fail at certain timestamps when some unusual events happen. We call such factors as environment factors, which are different across time.

The above two observations demonstrate the characteristics of dynamic data that need to be considered in modeling truth discovery on such data. Moreover, as the information comes continuously, it requires that the computation process should be in real-time. In the following section, we first propose an effective approach that can update truth and source reliability in real-time, and then address the effect of temporal smoothness and environment factors in dynamic data.

3. METHODOLOGY

We start with introducing some concepts in truth discovery, and then formally define the task. In the remaining part of this section, we first build an efficient algorithm for truth discovery with dynamic data, which provides an incremental scheme to guarantee high efficiency. Based on it, smoothing factor and decay factor are incorporated into the proposed method to capture the observations discussed above.

3.1 Problem Formulation

To describe the notations clearly, we group them as follows:

Input—Consider a set of objects $\mathscr{O}$ that we are interested in, and for each of them $o \in \mathscr{O}$, related information can be collected from *S* sources at each timestamp *t* ∈ {1, 2, 3,…}. Let $v_{o,t}^{s}$ represent the information from the *s*-th source about the object *o* at the *t*-th timestamp. For convenience, let's denote all the information from source *s* at time *t* as $\mathscr{X}_t^s$, that is, $\mathscr{X}_t^s = \left\{v_{o,t}^s\right\}_{o\in\mathscr{O}}$. Further, the size this set is of denoted as $c_t^s = |\mathscr{X}_t^s|$.

Output—After collecting information from different sources, our goal is to aggregate these information and output trustworthy ones. Let $\hat{v}_{o.t}^{*}$ be the aggregated result for object *o* at time *t*, and $\mathscr{X}_t^{*} = \{\hat{v}_{o.t}^{*}\}_{o\in\mathscr{O}}$ be the whole set aggregated results at time *t*.

Besides the aggregated results, truth discovery methods can also estimate sources' reliability degrees. Let w_s denote the weight (reliability degree) of the *s*-th source, and $\mathscr{W}$ represent the whole set of source weights. As source weights are estimated based on their information errors (difference) compared with the aggregated results, here we introduce some notations about source errors. Let $e_{o,t}^{s}$ indicate the error of the *s*-th source made on object *o* at time *t*, and e_t^s contain the errors on all the objects for source *s* at time *t*. Further, we have some notations for accumulated errors: $e_{1:t}^{s}$ denotes all the errors of source *s* from time 1 to time *t*, and $e_{1:t}$ contains such information for all the sources.

Table 1 summarizes the notations used in this paper.

Task Definition—The studied task is formally defined as follows. For a set of objects we are interested in, at timestamp *T*, related information is collected from *S* sources. Our goal is to find the most trustworthy information $\hat{v}_{o,T}^{*}$ for each object *o* by resolving the conflicts among information from different sources $\left\{v_{o,T}^{s}\right\}_{s=1}^{S}$. Meanwhile, to guarantee the efficiency, the proposed method should not re-visit the information at previous timestamps *t* ∈ {1, 2, 3,…,*T* – 1}. Besides the efficiency requirement, compared with tradition truth discovery tasks, the main difference of the proposed one is that the temporal evolution patterns within both objects and sources are investigated.

3.2 Proposed Method

When applying the existing truth discovery methods on dynamic data, the key limitation is their efficiency. Most of them iteratively update estimated source reliability and the identified trustworthy information. Thus multiple visits of the whole dataset are required. In dynamic scenario, it becomes inefficient or even infeasible as the data comes continuously. In the light of this challenge, we first develop an efficient truth discovery method for dynamic data by exploring the equivalence between optimization-based solution and maximum a posteriori estimation.

Optimization-Based Solution—At time *T*, we have all the information from the timestamp 1 to *T*. Based on the principles of truth discovery, we can consider the following optimization problem to infer both source reliability and trustworthy information:

$$min_{\mathscr{W},\{\mathscr{X}_t^*\}_{t=1}^T} L_T = \sum_{t=1}^{T} l_t, \quad (1)$$

where l_t is the loss function at time *t*, which is defined as follows:

$$l_t = \theta \sum_{s=1}^{S} w_s \sum_{o=1}^{c_t^s} \left(v_{o,t}^s - \hat{v}_{o,t}^* \right)^2 - \sum_{s=1}^{S} c_t^s log(w_s). \quad (2)$$

The motivation behind this loss function is following: 1) The first term $\sum_{s=1}^{S} w_s \sum_{o=1}^{c_t^s} (v_{o,t}^s - \hat{v}_{o,t}^*)^2$ measures the weighted distance between the provided information $v_{o,t}^s$ and the aggregated result $\hat{v}_{o,t}^*$. By minimizing this part, the aggregated result $\hat{v}_{o,t}^*$ will be closer to the information from the sources with high weight w_s. Meanwhile, if a source provides information that is far from the aggregated results, in order to minimize the total loss, it will be assigned a low weight. 2) The second term $\sum_{s=1}^{S} c_t^s log(w_s)$ works as a constraint to prevent w_s approaching 0, which leads to the trivial optimum for the first term. 3) Parameter θ adjusts the trade-off between these two terms above.

The benefits of adopting this optimization-based formulation are: 1) It encodes the idea of truth discovery. 2) It allows us to incorporate constraints and prior knowledge about source weights. 3) In the following, we will show that this formulation can be linked with MAP estimation which gives an efficient incremental solution.

In this optimization problem (Eq. (1)), two sets of variables are involved, source weights $\mathscr{W}$ and aggregated results $\{\mathscr{X}_t^*\}_{t=1}^T$. To solve this problem, we adopt coordinate descent [1], in which one set of variables are fixed in order to solve for the other set of variables.

When source weights $\mathscr{W}$ are fixed, to infer aggregated results $\hat{v}_{o,t}^*$, we take the derivative of Eq. (2) with respect to $\hat{v}_{o,t}^*$, and get the following formula:

$$\hat{v}^*_{o,t} = \frac{\sum_{s=1}^{S} w_s \cdot v^s_{o,t}}{\sum_{s=1}^{S} w_s}. \quad (3)$$

According to this weighted combination strategy to compute the aggregated results, the information provided by high quality sources will play more important roles, which keeps consistent with the basic principle of truth discovery.

However, to estimate the source weights $\mathscr{W}$ at time *T*, according to Eq. (1), we need to re-visit all the information from the first timestamp to the current one, which leads to additional cost and makes the algorithm inefficient.

Maximum A Posteriori Estimation—In order to improve the efficiency, we re-examine the above object function from another perspective, and propose to learn source weights based on a probabilistic model.

Recall that the error of the *s*-th source on object *o* at time *t* is defined as $e^s_{o,t} = v^s_{o,t} - \hat{v}^*_{o,t}$. As the weight of source indicates the quality of its provided information, the errors of a specific source given its source weight w_s can be assumed to follow a normal distribution:

$e^s_{o,t}|w_s \sim N(0, \frac{1}{\theta w_s})$, where θ is the trade-off parameter in the loss function. If the source weight w_s is high, the errors will be small, which is equivalent to the idea that the aggregated results should be close to the information from high quality sources. Next, we formally prove that the above optimization problem can be translated into an equivalent likelihood estimation task.

Theorem 3.1: Given the fixed aggregated results $\{\mathscr{X}^*_t\}_{t=1}^{T}$ and $e^s_{o,t}|w_s \sim N\left(0, \frac{1}{\theta w_s}\right)$, minimizing loss function L_T in Eq. (1) is equivalent to maximizing likelihood $\prod_{s=1}^{S} p(e^s_{1:T}|w_s)$.

Proof: We first give the formulation of likelihood:

$$\begin{aligned}\prod_{s=1}^{S} p(e^s_{1:T}|w_s) &= \prod_{t=1}^{T}\prod_{s=1}^{S} p(e^s_t|w_s) = \prod_{t=1}^{T}\prod_{s=1}^{S}\prod_{o=1}^{c^s_t} p(e^s_{o,t}|w_s) \\ &\propto \prod_{t=1}^{T}\prod_{s=1}^{S}\prod_{o=1}^{c^s_t}\left(\sqrt{\theta w_s}\right) e^{-\frac{\theta w_s (e^s_{o,t})^2}{2}} \\ &= \prod_{t=1}^{T}\prod_{s=1}^{S}\left(\sqrt{\theta w_s}\right)^{c^s_t} e^{-\frac{\theta w_s \sum_{o=1}^{c^s_t}(e^s_{o,t})^2}{2}}.\end{aligned} \quad (4)$$

To maximize Eq. (4), we can minimize its negative log likelihood, which is given as follows:

$$\begin{aligned} -log\left(\prod_{t=1}^{T}\prod_{s=1}^{S}\prod_{o=1}^{c_t^s}p(e_{o,t}^s|w_s)\right) \propto \\ \sum_{t=1}^{T}\left(\frac{1}{2}\sum_{s=1}^{S}\theta w_s\sum_{o=1}^{c_t^s}\left(e_{o,t}^s\right)^2-\frac{1}{2}\sum_{s=1}^{S}c_t^s log(w_s)-\frac{1}{2}\sum_{s=1}^{S}c_t^s log\theta\right) \end{aligned} \quad (5)$$

Since the third term in Eq. (5) is a constant, Eq. (5) and Eq. (1) are equivalent.

According to Theorem 3.1, the optimization problem in Eq. (1) can be transferred into likelihood estimation. Thus the posterior distribution of w_s after timestamp T can be linked with the distribution after timestamp $T - 1$ as follows:

$$\begin{aligned} p(w_s|e_{1:t}^s) &= p(w_s|e_T^s, e_{1:(T-1)}^s) \\ &\propto p(w_s, e_t^s, e_{1:(T-1)}^s) \\ &\propto p(e_t^s|w_s, e_{1:(T-1)}^s)p(w_s|e_{1:(T-1)}^s) \\ &= p(e_T^s|w_s)p(w_s|e_{1:(T-1)}^s) \end{aligned} . \quad (6)$$

Eq. (6) gives an incremental way to estimate the source weight at time T: The weight can be updated based on the weight at time $T - 1$, and it is not necessary to re-visit the previous data. This dramatically improves the efficiency of the proposed method.

In order to incorporate prior knowledge, we use Maximum a posteriori (MAP) estimation to estimate source weight w_s. We set the prior distribution for w_s as Gamma distribution $p(w_s) \propto \text{Gamma}(\alpha, \beta)$, which is equivalent to have an initial loss $l_0=\sum_{s=1}^{S}w_s\beta-\sum_{s=1}^{S}(\alpha-1)log(w_s)$.

By combining the prior knowledge and Eq. (6), we get the posterior distribution of w_s after timestamp T:

$$\begin{aligned} p(w_s|e_{1:t}^s) &\propto p(e_T^s|w_s)p(w_s|e_{1:(T-1)}^s) \propto p(w_s)\prod_{t=1}^{T}p(e_t^s|w_s) \\ &\propto w_s^{\alpha-1}e^{-\beta w_s}\prod_{t=1}^{T}(\sqrt{\theta w_s})^{c_t^s}e^{-\frac{\theta w_s\sum_{o=1}^{c_t^s}(e_{o,t}^s)^2}{2}} \\ &\propto w_s^{\alpha-1}e^{-\beta w_s}w_s^{\frac{\sum_{t=1}^{T}c_t^s}{2}}e^{-\frac{\theta w_s\sum_{t=1}^{t}\sum_{o=1}^{c_t^s}(e_{o,t}^s)^2}{2}} \\ &= w_s^{\alpha-1+\frac{\sum_{t=1}^{T}c_t^s}{2}}e^{-(\beta+\frac{\theta}{2}\sum_{t=1}^{T}\sum_{o=1}^{c_t^s}(e_{o,t}^s)^2)w_s} . \end{aligned} \quad (7)$$

This indicates that: $p(w_s|e_{1:t}^s) \sim \text{Gamma}(\alpha+\frac{\sum_{t=1}^{T}c_t^s}{2}, \beta+\frac{\theta}{2}\sum_{t=1}^{T}\sum_{o=1}^{c_t^s}(e_{o,t}^s)^2)$. Thus the MAP estimation for source weight w_s after timestamp T is:

$$w_s = \frac{2\alpha - 2 + \sum_{t=1}^{T} c_t^s}{2\beta + \theta \sum_{t=1}^{T} \sum_{o=1}^{c_t^s} \left(e_{o,t}^s\right)^2} \quad (8)$$

From Eq. (8), we can see that the estimated source weight is inversely proportional to the average difference between its provided information and the aggregated results. According to the basic principle of truth discovery, if a source provides information that is close to the aggregated results, its corresponding weight should be high. The above equation for source weight estimation follows this principle. Meanwhile, parameters from prior distribution α and β also exert their effect on the source weight estimation.

Let's denote the accumulated counts for the *s*-th source by times-tamp *t* as a_t^s, where $a_0^s = 2\alpha - 2$ and $a_t^s = a_{t-1}^s + c_t^s$. Similarly, let b_t^s represent the accumulated error for the *s*-th source by timestamp *t*, where $b_0^s = 2\beta$, and $b_t^s = b_{t-1}^s + \theta \sum_{o=1}^{c_t^s} \left(e_{o,t}^s\right)^2$. Thus, Eq. (8) becomes:

$$w_s = \frac{a_{T-1}^s + c_T^s}{b_{T-1}^s + \theta \sum_{o=1}^{c_T^s} \left(e_{o,T}^s\right)^2}. \quad (9)$$

Following this equation, to update the source weights after times-tamp *T*, we only need to count the information and calculate the errors that happen within the timestamp *T*. This guarantees that we do not need to re-visit the data at previous timestamps.

Algorithm Flow—So far, we have derived the incremental way to update source weights. The whole computation procedure for each timestamp is illustrated in Algorithm 1. Similar to the existing truth discovery methods, the proposed method follows the general principle to estimate source reliability and infer trustworthy information. The advantage of the proposed method is that it works in an incremental way and only scans the data once. Thus it has great efficiency and is suitable for dynamic scenario. The theoretical analysis in Section 4 will prove that the estimated source weights converge to the true source reliability. In the following, we will show that by slightly modifying this solution, the proposed method can capture the observations in Section 2.

Algorithm 1: Incremental Algorithm Flow

```
Input: Information from S sources at timestamp $T: \{\mathscr{X}_t^s\}_{s=1}^S$, and the accumulated counts and errors at previous timestamp.
Output: Aggregated information $\mathscr{X}_t^s$.
    Update the aggregated results $\hat{v}_{o,T}^*$ according to Eq. (3) based on the current estimation of source weights.
    Compute the count $c_T^s$ and error $e_{o,t}^s$ for each source.
    Update the weight of s-th source according to Eq. (9), and meanwhile, update the accumulated counts and accumulated errors for next timestamp.
    return $\mathscr{X}_t^*$.
```

Smoothing Factor—In Section 2, we show that the information of objects is evolving in a smooth way. To capture this observation, we add one smoothness constraint on the aggregated results. Thus, the loss function at time *t* becomes:

$$\begin{aligned} l_t = & \theta \sum_{s=1}^{S} w_s \sum_{o=1}^{c_t^s} (v_{o,t}^s - \hat{v}_{o,t}^*)^2 - \sum_{s=1}^{S} c_t^s log(w_s) \\ & + \theta\lambda \sum_{o \in \mathscr{O}} (\hat{v}_{o,(t-1)}^* - \hat{v}_{o,t}^*), \end{aligned} \tag{10}$$

where λ is a parameter to control the effect of this smoothness constraint. This added term will not affect the source weight estimation step as the aggregated results are fixed in this step. For aggregation step, we can take the derivative of Eq. (10) with respect to $\hat{v}_{o,t}^*$:

$$\hat{v}_{o,t}^* = \frac{\sum_{s=1}^{S} w_s v_{o,t}^s + \lambda \hat{v}_{o,(t-1)}^*}{\sum_{s=1}^{S} w_s + \lambda}. \tag{11}$$

Let's consider the aggregated results at previous timestamp *t* − 1 as a pseudo source for the corresponding objects at timestamp *t*. If we treat λ as the weight of this pseudo source, the aggregation is still in the form of weighted combination of all the sources. Based on this, we incorporate the smoothness assumption in a simple yet elegant way: at each time *t*, we treat the previous aggregated result $\hat{v}_{o,(t-1)}^*$ as the information from a pseudo source, and the proposed algorithm remains the same.

One thing needs to set is the tuned parameter λ, which can be interpreted as "the weight of the pseudo source". Big λ suggests that we should rely on this constraint more, while small λ can relax this constraint. We will study the effect of this parameter on both real world applications and synthetic data in Section 5.

Decay Factor—Another observation in Section 2 is that the observed source reliability fluctuates around its true source weight, and it can be explained by the environment factor. In order to tackle this challenge in the proposed method, we introduce a decay factor γ for source weight estimation as follows:

$$w_s = \frac{2\alpha - 2 + \sum_{t=1}^{T} \gamma^{T-1} \cdot c_t^s}{2\beta + \theta \sum_{t=1}^{T} \sum_{o=1}^{c_t^s} \gamma^{T-t} \cdot (e_{o,t}^s)^2} \tag{12}$$

If we treat the true source weight as a "global" weight, this formula enables us to estimate a "local" one, which captures both the true source weight and the environment factor. The introduced decay factor γ is used to balance the true source weight and environment factor: If γ is close to 1, the estimated local weight will be close to the true source weight; While if it is smaller than 1, the local weight captures the environment factor more. In experiments we will show that by incorporating this decay factor, the proposed method can model the balance between the true source weight and the environment factor.

4. THEORETICAL ANALYSIS

In this section, we prove the estimated source weights given by the proposed method (Eq. (8)) will converge to the true source weights, and further, the rate of convergence is as fast as $o\left(\frac{1}{\sqrt{T}}\right)$. To prove this, we need the following lemma first.

Lemma 4.1

Suppose for each T, there exists a $\mathscr{W}_T^*$ that minimizes $L_T = \sum_{t=1}^{T} l_t$. Then the posterior distribution $p(\mathscr{W}|e_{1:t})$ satisfies the asymptotic normality:

$$(\nabla^2 L_T(\mathscr{W}_T^*))^{1/2}(\mathscr{W} - \mathscr{W}_T^*) \xrightarrow{d} N(0,1),\ as\, t \to \infty, \quad (13)$$

where $\mathscr{W} \sim p(\mathscr{W}|e_{1:T})$.

Proof

We use the Theorem 2.1 in [2] to prove the asymptotic normality. The theorem states that if a series of functions satisfies certain sufficient conditions, then the asymptotic normality stated in Lemma 4.1 holds. Therefore, we only need to prove that our functions satisfy those conditions. Recall that the posterior distributions $p\,(\mathscr{W}|e_{1:t})$ are a series of probability density distributions with respect to *t*. Suppose that $\mathscr{W}_T^*$ is the local maximum of $log\, p(\mathscr{W}|e_{1:T})$. We need to prove the conditions as listed below:

P1. $\nabla log\, p(\mathscr{W}|e_{1:T})|\mathscr{W}_T^* = 0$.

P2. $\sum_T \equiv \left\{-\nabla^2 log\, p(\mathscr{W}|e_{1:T})|\mathscr{W}_T^*\right\}^{-1}$ is positive definite.

C1. “Steepness”: $\sigma_t^2 \to 0$, as $t \to \infty$, where σ_t^2 is the largest eigenvalue of Σ_t.

C2. “Smoothness”: for any $\varepsilon > 0$, there exists an integer *N* and $\delta > 0$ such that, for any *t* > *N*, $\mathscr{W}' \in H(\mathscr{W}_t^*;\delta) = \left\{|\mathscr{W}' - \mathscr{W}_t^*| < \delta\right\}$, $\nabla^2 log\, p(\mathscr{W}|e_{1:t})|_{\mathscr{W}'}$ satisfies $I - A(\varepsilon) \le \nabla^2 log\, p(\mathscr{W}|e_{1:t})_{|\mathscr{W}'} \left\{\nabla^2 log\, p(\mathscr{W}|e_{1:t})|\mathscr{W}_t^*\right\}^{-1} \le I + A(\varepsilon)$, where *I* is identity matrix and *A*(ε) is positive semidefinite symmetric matrix whose largest eigenvalue goes to 0 as $\varepsilon \to 0$.

C3. “Concentration”: for any $\delta > 0$, the probability $\int_{H(\mathscr{W}_t^*;\delta)} p(\mathscr{W}|e_{1:t}) d\mathscr{W} \to 1$ as $t \to \infty$.

Proof of (P1) and (P2)—We first prove (P1) and (P2). Here we assume that at each time *t*, the number of claims made by a source c_t^* is a positive constant. The truths $v_{o,t}^*$ for *t* = 1, …,*T* are also known, so we can treat $\theta \sum_{0=1}^{c_t^s} (v_{o,t}^s - v_{o,t}^*)^2$ as a constant (denoted as q_t^s). As shown is Eq. (7) and the definition of the loss functions (Eq. (2)):

$$log\,p(\mathscr{W}|e_{1:T})=log\,p(\mathscr{W})+\sum_{t=1}^{T}log\,p(e_{1:t}|\mathscr{W}) \propto -\sum_{t=0}^{T}l_t.$$

Therefore, $logp\,(\mathscr{W}|e_{1:T}) = -C*L_T$, where C is a positive constant. Since there exists a $\mathscr{W}^*_T$ that minimizes L_T, we have:

$$\nabla L_T(\mathscr{W}^*_T)_s=[\sum_{t=0}^{T}q^s_t-\frac{\sum_{t=0}^{T}c^s_t}{w^*_{s,T}}]_s=0.$$

Further,

$$\nabla^2 L_T(\mathscr{W}^*_T)=diag(\frac{\sum_{t=0}^{T}c^s_t}{w^{*2}_{s,T}})_s,$$

which is positive definite. Note that $\nabla^2 L_T(\mathscr{W}^*_T)^{-1}=diag(\frac{w^{*2}_{s,T}}{\sum_{t=0}^{T}c^s_t})_s$ corresponds to Σ_T in (P2), and therefore (P1) and (P2) are satisfied.

Proof of (C1)—As $T \to \infty$, $\sum_{t=0}^{T}c^s_t$, the number of claims made by a source, will go to infinity, which means all the eigenvalues of $\nabla^2 L_T(\mathscr{W}^*_T)^{-1}$ will go to 0. Thus the "steepness" condition (C1) is satisfied.

Proof of (C2)—Since all elements in $\nabla^2 L_t(\mathscr{W})$ is continuous with respect to $w_{s,t}$, the "smoothness" assumption (C2) for $\nabla^2 L_t(\mathscr{W})$ is straightforward. To be more specific, if $\mathscr{W}' \in H(\mathscr{W}^*_t;\delta)=\{|\mathscr{W}'-\mathscr{W}^*_t|<\delta\}$, we have $\frac{1}{1+\delta'}<\frac{w^{*2}_{s,t}}{w'^2_s}<\frac{1}{1-\delta'}$ where $\delta'=\frac{\delta}{w^*_{s,t}}$. Since $\nabla^2 L_t(\mathscr{W}')\nabla^2 L_t(\mathscr{W}^*_t)^{-1}=diag(\frac{w^{*2}_{s,t}}{w'^2_s})$, it immediately implies (C2).

Proof of (C3)—Based on (P1) and (P2), $\mathscr{W}^*_T$ exists and for each source, it can be calculate as $\frac{\sum_{t=0}^{T}c^s_t}{\sum_{t=0}^{T}q^s_t}$. In fact, $w^*_{s,T}$ is the mean of $\mathrm{Gamma}(\sum_{t=0}^{T}c^s_t,\sum_{t=0}^{T}q^s_t)$, the posterior distribution. As $T \to \infty$, $\sum_{t=0}^{T}q^s_t$ will go to infinity. Since $\frac{\sum_{t=0}^{T}c^s_t}{\sum_{t=0}^{T}q^s_t}$ is a constant, the variance of the posterior distribution $\frac{\sum_{t=0}^{T} c_t^s}{\sum_{t=0}^{T} q_t^s} * \frac{1}{\sum_{t=0}^{T} q_t^s} \to 0$, which means $E_{p(\mathscr{W}|e_{1:T})}\left[(\mathscr{W}-\mathscr{W}_T^*)^2\right] \to 0$. Therefore, concentration” assumption is proved.

As shown above, all the sufficient conditions for Theorem 2.1 in [2] are satisfied, so Lemma 4.1 holds.

Lemma 4.1 states the approximated distribution of the posterior $p(\mathscr{W}|e_{1:t})$. As $T \to \infty$, the asymptotic distribution implies that the estimation based on the posterior distribution can converge to the minimizer of the accumulated loss $\mathscr{W}_T^*$. Next, we will show that the weight given by Eq. (8) can converge to the true source weight $\mathscr{W}^*$ at rate of $o(\frac{1}{\sqrt{T}})$.

Theorem 4.2

If for any t, l_t = l, then as T $\to \infty$, $\hat{\mathscr{W}}$, given by Eq. (8), converges to $\mathscr{W}^*=\arg\min_{\mathscr{W}} E\,[l]\, at$

$$\sqrt{T}(\hat{\mathscr{W}}-\mathscr{W}^*) \xrightarrow{d} N(0,\sum), \quad (14)$$

where $\sum=(\nabla^2 E(l))^{-1} Var(\nabla l)(\nabla^2 E(l)^{-1})'$.

Proof

In order to prove Theorem 4.2, we will first prove that $\sqrt{T}(\hat{\mathscr{W}}-E_{p(\mathscr{W}|e_{1:T})}) \to 0$, then prove that $|E_{p(\mathscr{W}|e_{1:T})}(\mathscr{W})-\mathscr{W}_T^*|=o(\frac{1}{\sqrt{T}})$, and finally prove that $\sqrt{T}(\mathscr{W}_T^*-\mathscr{W}^*) \xrightarrow{d} N(0,\sum)$.

The weight given by Eq. (8) is the posterior mode of $p(w_s|e_{1:t}^s)$, a Gamma distribution. Therefore,

$$\begin{aligned}|\hat{\mathscr{W}}-E_{p(\mathscr{W}|e_{1:T})}(\mathscr{W})| &= \sum_{s=1}^{S} |\hat{w}_s - E_{p(w_s|e_{1:T}^S)}(w_s)| \\ &= \sum_{s=1}^{S} \frac{2}{2\beta+\theta\sum_{t=1}^{T}\sum_{o=1}^{c_t^s}(e_{o,t}^s)^2} \\ &= \Theta(\tfrac{1}{T}) = o(\tfrac{1}{\sqrt{T}}),\end{aligned} \quad (15)$$

where $E_{p(w_s|e_{1:T}^s)}(w_s)$ is the posterior mean.

From Lemma 4.1, we can take the expectation on the asymptotic distribution and get the following:

$$E\left\{(\nabla^2 L_T(\mathscr{W}_T^*))^{1/2}(\mathscr{W}-\mathscr{W}_T^*)\right\} \to 0,$$

which implies that

$$|E_{p(\mathscr{W}|e_{1:T})} - \mathscr{W}_T^*| = o(1)|(\nabla^2 L_T(\mathscr{W}_T^*))^{-1/2}|.$$

As shown in the proof for Lemma 4.1, $\nabla^2 L_T(\mathscr{W}_T^*) = \Theta(T)$, therefore $|E_{p(\mathscr{W}|e_{1:T})} - \mathscr{W}_T^*| = o(\frac{1}{\sqrt{T}})$.

Since $\mathscr{W}_T^* = arg\min_{\mathscr{W}} L_T$, it is also the minimizer of $\frac{1}{T}\sum_{t=0}^{T} l_t$, which converges to $E(l)$ by the law of large number. By the theorem of stochastic gradient decent [16], $\sqrt{T}(\mathscr{W}_T^* - \mathscr{W}^*) \xrightarrow{d} N(0, \sum)$, where $\sum (\nabla^2 E(l))^{-1} Var(\nabla l)(\nabla^2 E(l)^{-1})'$

So far, we prove the followings: $\sqrt{T}(\hat{\mathscr{W}} - E_{p(\mathscr{W}|e_{1:T})}) \rightarrow 0$, $\sqrt{T}(E_{p(\mathscr{W}|e_{1:T})} - \mathscr{W}_T^*) \rightarrow 0$, and $\sqrt{T}(\mathscr{W}_T^* - \mathscr{W}^*) \xrightarrow{d} N(0, \sum)$. By Slutsky's theorem, we get $\sqrt{T}(\hat{\mathscr{W}} - \mathscr{W}^*) \xrightarrow{d} N(0, \sum)$.

The theoretical analysis above demonstrates that the proposed method (Eq. (8)) has not only the intuitive explanation, but also the theoretical guarantee: the estimated source weights converge to the true weights in a fast speed. We will confirm this claim experimentally in next section.

5. EXPERIMENTS

In this section, we experimentally validate the proposed method for truth discovery under dynamic scenario from the following aspects: 1) On three real world datasets, the proposed method achieves better performance comparing with state-of-the-art truth discovery algorithms, while the efficiency is significantly improved. We also show that the introduced smoothing and decay factors can capture the temporal relations among the evolving truths and source weights. 2) On synthetic data, the theoretical analysis is confirmed by the experiments. We further systematically study the effect of smoothing and decay factors, and the efficiency is tested on large-scale dataset.

5.1 Experiment Setup

Compared Methods—For the proposed method, we test the incremental version (denoted as **Dyna**mic **T**ruth **D**iscovery, **DynaTD** for short), incremental version with smoothing factor (**DynaTD+smoothing**), incremental version with decay factor (**DynaTD+decay**), and incremental version with both smoothing and decay factors (denoted as **DynaTD+all**). By gradually adding more components into the proposed method, we can investigate the benefit of considering these factors and understand how it tackles the challenges.

For baseline methods, the following state-of-the-art truth discovery methods are implemented: **TruthFinder** and **AccuSim** adopt pre-defined rules to update both source reliability and aggregated results. Investment borrows the idea that a source "invests" its reliability on the information it provides. **3-Estimate** further extends the scope by considering the difficulty of aggregating information for specific objects. **GTM** is a probabilistic graphical model based truth discovery method that is designed for continuous data, and **CRH** is a truth discovery framework that can incorporate various distance functions and work with heterogeneous data. **CATD** is a recent truth discovery approach that considers the confidence interval of the source reliability estimation.

Beyond the above truth discovery algorithms, we also implement two naive methods **Mean** and **Median**, which do not consider the source reliability and simply take the mean or median of all information as aggregated results.

Note that the baseline methods cannot take streaming data as input. Therefore, their settings are not exactly the same with the proposed methods. For all the baselines, since they cannot capture the temporal relations among evolving truths and source weights, they are deployed on the entire dataset in a batch way, and treat the same object at different timestamps as different objects. In contrast, the proposed methods work in an incremental fashion, and can deal with the evolving truths and source weights. To model the temporal relations among the true information, either the groundtruth or the aggregated results for past timestamps can be adopted.

Performance Metrics—To evaluate the performance, we calculate the following metrics on aggregate results by comparing them with the groundtruth: Mean of Absolute Error (MAE) and Root of Mean Squared Error (RMSE). MAE uses L^1-norm distance that penalizes more on small errors, while RMSE adopts L^2-norm distance that gives more penalty on big errors. In the results analysis, we will discuss their difference in more detail. For both metrics, lower value indicates better performance.

To assess the efficiency, we also report each method’s running time. For each baseline, we implement it and set its parameters according to the original papers. All the methods are run on a machine with 16G RAM, Intel Core i7 processor.

5.2 Experiments on Real World Datasets

Datasets—In order to evaluate the proposed method in real world applications, we adopt the aforementioned Weather Forecast dataset, Stock Record dataset, and Flight Status dataset as testbeds. Here we provide more details about these datasets.

- Weather Forecast: We collect high temperature forecast information for 88 big US cities from HAM weather (HAM)[1], Wunderground (Wund)[2], and World Weather On-line (WWO)[3]. The collection lasts for more than two months (Oct. 7, 2013 to Dec 17, 2013). In addition to the forecast information, real high temperature observations of each day are also collected for evaluation purpose.

[1] http://www.hamweather.com
[2] http://www.wunderground.com
[3] http://www.worldweatheronline.com

- Stock Record: The stock data [8] contains information for 1000 stocks that are collected from 55 sources during each weekday of July 2011. As we assume the information is continuous data type, *Market* property is adopted. The groundtruth information is provided by the authors.
- Flight Status: The flight data [8] extracts departure and arrival information for 1200 flights from 38 sources during every day in December 2011. All the time information is translated into minutes (for example, 07: 30am is translated into 450 mins). The groundtruth information is also available from the authors.

Performance Comparison—Table 2 summarizes the results for all the methods on the three real datasets. In terms of aggregation accuracy, the proposed method achieves best performance on every dataset, and the improvement is promising: on Weather dataset, compared with the best baseline **CRH**, the proposed method's MAE decreases by 3% while RMSE decreases by 6.23%; on Stock dataset, compared with the best baseline method **CRH**, MAE and RMSE of the proposed method decrease by 82.4% and 70.1% respectively; while on Flight dataset, compared with the best baselines Investment and **GTM**, the proposed method's MAE reduces by 7.4% and RMSE reduces by 11.4%. Among the baseline methods, **Mean** and **Median** simply aggregate the multi-source information without considering source reliability, and thus they have the worst performance. **TruthFinder**, **AccuSim**, **Investment** and **3-Estimates** take categorical data as input, so they are not able to handle continuous data type well. In this sense, **GTM**, **CRH** and **CATD** are more appropriate for these tasks. Our proposed method is based on the advantages of these existing methods, and by considering the time-evolving truths and source weights in dynamic scenarios, it achieves the best performance.

In terms of efficiency, the running time of the proposed methods is close to **Mean** and **Median**, which can be viewed as the lower bound of running time for truth discovery algorithms. Compared with the most efficient truth discovery baseline method, the proposed method significantly reduces the running time: on Weather dataset, it runs two times faster than **CRH**; on Stock dataset, it runs four times faster than **GTM**; while on Flight dataset, it runs eight times faster than Investment. Most truth discovery methods require some iterations to converges, so it is time-consuming. Among them, **TruthFinder** and **AccuSim** need to calculate the implication function, and **3-Estimates** needs to estimate the difficulty of each object, so their running time increases dramatically.

To summarize, under the dynamic scenario, the proposed methods outperform all the baselines in terms of both accuracy and efficiency. Among the proposed methods, we can observe that by adding smoothing factor and decay factor, the performance is further improved. Next, we analyze the effect of these two factors.

The Effect of Smoothing Factor—Smoothing factor is used to deal with the evolving truths. Pseudo sources are created to model this, and the smoothing factor adjusts how much influence these pseudo sources will exert. Figure 3 shows the effect of different smoothing factors on the three real world datasets. For Weather dataset, a relative small smoothing factor leads to the best performance. For Stock dataset, the change range of stock information on the collected data is small, so a big smoothing factor is suitable. The most interesting one is Flight dataset: the MAE and RMSE change in different ways as smoothing factor increases. Except some special cases, a flight will depart or arrive around the scheduled time, so a large smoothing factor will enforce the aggregated results to be close to the history, which can avoid large errors, which leads to the decreasing in RMSE as L^2-norm gives more penalty on large errors. However, enforcing the results to be close to history information will bring some small errors, since the real departure and arrival time would not stay exactly the same. As a consequence, MAE increases as it penalizes more on small errors. From these results we can see that the introduced smoothing factor has the ability to capture the unique characteristics of real world applications.

The Effect of Decay Factor—Under dynamic scenario, the errors might be introduced by environment factors instead of the source itself, so the observed source reliability fluctuates around its true weight. To tackle this challenge, decay factor is introduced to estimate local weights. We first examine how environment factors are captured by comparing the estimated weights without decay factor and with decay factor. Figure 4 reports the comparison on three real world datasets, in which blue dot lines show the estimated weights without considering decay factor while red line represents the estimated weights by considering decay factor. From these estimated weights, we can draw the conclusion that: without decay factor, the estimated source weight converges to its true weight (we will further confirm this in the experiment on simulated data); while by considering the decay factor, the estimated weights capture both the true weight trend and the local environment factors.

Next, we study the effect of decay factor. Figure 5 illustrates how the performance (in terms of both MAE and RMSE) changes with respect to decay factor. For Weather application, as the environment factor in it is relative stable and small, the performance reaches the best when the decay factor is close to 1, that is, the local weights should be close to the global weights. However, for Stock application, the information changes in a fast speed, which means the effect of environment factors is big, so the improvement can be achieved when the decay factor is close to 0 and local weights are estimated. For Flight application, the environment is stable for most cases, but it may change rapidly, especially when flights delay due to unanticipated reasons, so a medium decay factor is more suitable. From these results, we observe that the introduced decay factor can handle the challenge of the fluctuated source weights.

5.3 Experiments on Synthetic Datasets

The experiments on the real world datasets demonstrate that the proposed method can improve the accuracy while largely reduce the running time, and the effect of introduced factors are also justified. In this section, we design a series of experiments on synthetic datasets to confirm the theoretical analysis, study the relationship between the introduced factors and environment factors, and test the efficiency on large-scale datasets.

Weight Convergence Study—To confirm the theoretical convergence analysis of the proposed method, we simulate the behavior of ten sources with different reliability levels for

20 timestamps. We assign each source a true weight, and at each timestamp, we generate 2000 observations for each source according to its true weight, where the errors follows $N(0, \frac{1}{w_s^*})$. Then the simulated data is fed into the proposed method to estimate source weights. Figure 6 illustrates the comparison, in which black lines show the true source weights and red dot lines represent the estimated weights by the proposed method. We observe that the estimated weights (red dot line in figures) quickly converge to the corresponding true weights (black line in figures). This result experimentally confirms Theorem 4.2 in Section 4.

Smoothing Factor—As the smoothing factor is introduced to capture the information evolution, here we study how the change rate of information can affect the best choice for smoothing factor. For clarity and simplicity purpose, we simulate a linear evolution of the true information, and control the change rates by different slopes, where a small slope indicates a slow change rate and vice versa. The data is generated using the same method as described in the previous experiment. Figure 7 shows the best smoothing factor for different change rates, and the choice is made based on MAE and RMSE respectively. They both show that when information changes slowly, we can rely on history information more and thus big smoothing factor works better. In contrast, the smoothness constraint should be relaxed when the information changes quickly.

Decay Factor—We also study the relation between decay factor and environment factors. The environment factors are simulated through adding Gaussian noise to the observations, and they can be tuned by changing the variance of the Gaussian distribution. A small variance indicates a stable environment and vice versa. As the environment would have the same impact on all sources, we add the same level of noise to all the sources at a given timestamp. Other settings are kept the same as the one in above convergence experiment. Figure 8 shows the best decay factor with respect to different levels of environment factor. When the environment factor is small, the local weight is almost the same as the true weight (global one), so the decay factor should be close to 1. On the other hand, when the environment factor is quite large, the local weight, which captures the environmental changes, should play a more important role, so the best decay factor will tend to be small.

Efficiency Study—We further test the efficiency of the proposed method DynaTD on large scale dataset. We simulate various scales of datasets by controlling the number of claims made by each source at a given timestamp. As **Mean** and **Median** do not estimate source reliability, they have the optimal efficiency comparing with truth discovery methods. Here we use their running time as references and compare the running time of the proposed method with them. From Figure 9, the conclusion can be drawn that the proposed method (black line) has the nearly optimal efficiency compared with **Mean** (blue line) and **Median** (red line), even on very large dataset. This improvement is brought by the fact that the proposed method incrementally updates the source weights and only scans the whole data once.

6. RELATED WORK

Truth discovery is a hot topic for resolving conflicts among multi-source noisy information. Its advantage is the estimation of source reliability: instead of treating all sources equally, these methods infer the reliability of each source and incorporate such estimated source reliability into the aggregation. The early-stage truth discovery methods [5, 8, 14, 21] iteratively update the estimated source reliability and the aggregated results according to some pre-defined rules.

Recently, more truth discovery methods are proposed to fit various scenarios. As most existing methods take facts (categorical data) as input, to enlarge the scope of applications, GTM [24] is specially designed for continuous data, and CRH [7] is a framework that can plug in different types of distance function to capture the unique characteristic of different data types and conduct the estimation jointly. Another direction of truth discovery is the source correlation analysis [3, 17], in which sources are not independent and they may copy from each other. To further improve the advantage of truth discovery, people are considering to estimate more information rather than a single reliability degree for each source, and they enrich the meaning of source reliability from different aspects [6, 13, 15, 25]. Nowadays, truth discovery has been applied into several domains, including social and crowd sensing [18, 19], knowledge fusion [4], online health communities [11], etc.

There is a few work that shares some similarities with our work. In [10], the authors consider the information for truth discovery can be collected continuously, but they study this scenario from the source perspective. That is, they dynamically choose information sources from a pool to retrieve information, while in our setting, the information sources are fixed. The proposed method in [12] takes into account the evolving true information of objects, and estimates the truths of current timestamp based on sources' historical claims. However, in our setting, no historical data are kept due to space limit. In [20], the authors propose a method to handle time-varying truths, but it works in batch operation on categorical data. A single-pass truth discovery method [26] is proposed to fit streaming data, but it fails to consider the unique characteristics of dynamic data and the proposed method is designed for categorical data. To the best of our knowledge, we are the first to propose an efficient algorithm to capture the temporal relations among both information and source reliability for truth discovery.

7. CONCLUSIONS

In this paper, we propose to discover truths from dynamic data, where the collected information comes sequentially, and both truths and the source reliability evolve over time. This is a challenging task since we have to come up with an efficient way to capture the temporal relations among the identified trustworthy information and source reliability. To address the efficiency issue, we propose an incremental method by studying the equivalence between optimization-based solution and MAP estimation. Theoretical analysis shows that the proposed method can guarantee the convergence of the estimated source weights and the rate of convergence is as fast as $o(\frac{1}{\sqrt{T}})$. To capture the temporal relations among both the identified trustworthy information and source reliability, we further incorporate two more factors, smoothing factor λ and decay factor γ, into the proposed method. Experiments on both real world and synthetic datasets demonstrate that the proposed method has great efficiency while the integration performance is further improved by capturing those temporal relations. The effect of smoothing factor λ and decay factor γ are also studied under various settings.

Acknowledgments

The work was sponsored in part by the U.S. Army Research Lab. under Cooperative Agreement No. W911NF-09-2-0053 (NSCTA), National Science Foundation IIS-1319973, IIS-1017362, IIS-1320617, and IIS-1354329, HDTRA1-10-1-0120, and grant 1U54GM114838 awarded by NIGMS through funds provided by the trans-NIH Big Data to Knowledge (BD2K) initiative (www.bd2k.nih.gov), and MIAS, a DHS-IDS Center for Multimodal Information Access and Synthesis at UIUC. The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing any funding agencies.

References

1. Bertsekas, DP. Non-linear Programming. 2. Athena Scientific; 1999.
2. Chen CF. On asymptotic normality of limiting density functions with bayesian implications. Journal of the Royal Statistical Society. Series B (Methodological). 1985:540–546.
3. Dong XL, Berti-Equille L, Srivastava D. Integrating conflicting data: The role of source dependence. PVLDB. 2009; 2(1):550–561.
4. Dong XL, Gabrilovich E, Heitz G, Horn W, Lao N, Murphy K, Strohmann T, Sun S, Zhang W. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. Proc of KDD. 2014:601–610.
5. Galland A, Abiteboul S, Marian A, Senellart P. Corroborating information from disagreeing views. Proc of WSDM. 2010:131–140.
6. Li Q, Li Y, Gao J, Su L, Zhao B, Murat D, Fan W, Han J. A confidence-aware approach for truth discovery on long-tail data. PVLDB. 2015; 8(4):425–436.
7. Li Q, Li Y, Gao J, Zhao B, Fan W, Han J. Resolving conflicts in heterogeneous data by truth discovery and source reliability estimation. Proc of SIGMOD. 2014:1187–1198.
8. Li X, Dong XL, Lyons KB, Meng W, Srivastava D. Truth finding on the deep web: Is the problem solved? PVLDB. 2012; 6(2):97–108.
9. Li Y, Gao J, Meng C, Li Q, Su L, Zhao B, Fan W, Han J. A survey on truth discovery. arXiv preprint arXiv:1505.02463. 2015
10. Liu X, Dong XL, Ooi BC, Srivastava D. Online data fusion. PVLDB. 2011; 4(11):932–943.
11. Mukherjee S, Weikum G, Danescu-Niculescu Mizil C. People on drugs: credibility of user statements in health communities. Proc of KDD. 2014:65–74.
12. Pal A, Rastogi V, Machanavajjhala A, Bohannon P. Information integration over time in unreliable and uncertain environments. Proc of WWW. 2012:789–798.
13. Pasternack J, Roth D. Comprehensive trust metrics for information networks. Army Science Conference. 2010
14. Pasternack J, Roth D. Knowing what to believe (when you already know something). Proc of COLING. 2010:877–885.
15. Pasternack J, Roth D. Latent credibility analysis. Proc of WWW. 2013:1009–1020.
16. Pasupathy R, Kim S. The stochastic root-finding problem: Overview, solutions, and open questions. ACM Transactions on Modeling and Computer Simulation (TOMACS). 2011; 21(3):19.
17. Pochampally R, Sarma AD, Dong XL, Meliou A, Srivastava D. Fusing data with correlations. Proc of SIGMOD. 2014:433–444.
18. Su L, Li Q, Hu S, Wang S, Gao J, Liu H, Abdelzaher T, Han J, Liu X, Gao Y, Kaplan L. Generalized decision aggregation in distributed sensing systems. Proc of RTSS. 2014:1–10.

19. Wang D, Kaplan L, Le H, Abdelzaher T. On truth discovery in social sensing: A maximum likelihood estimation approach. Proc of IPSN. 2012:233–244.

20. Wang S, Wang D, Su L, Kaplan L, Abdelzaher T. Towards cyber-physical systems in social spaces: The data reliability challenge. Proc of RTSS. 2014:74–85.

21. Yin X, Han J, Yu PS. Truth discovery with multiple conflicting information providers on the web. Proc of KDD. 2007:1048–1052.

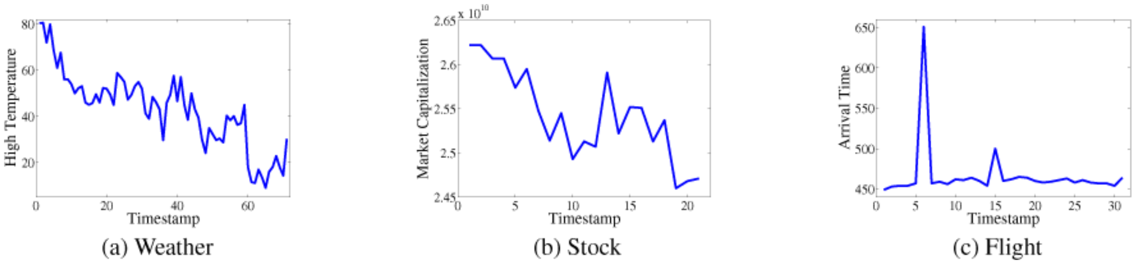
22. Yin X, Han J, Yu PS. Truth discovery with multiple conflicting information providers on the web. IEEE Transactions on Knowledge and Data Engineering. 2008; 20(6):796–808.

23. Yin X, Tan W. Semi-supervised truth discovery. Proc of WWW. 2011:217–226.

24. Zhao B, Han J. A probabilistic model for estimating real-valued truth from conflicting sources. Proc of QDB. 2012

25. Zhao B, Rubinstein BIP, Gemmell J, Han J. A bayesian approach to discovering truth from conflicting sources for data integration. PVLDB. 2012; 5(6):550–561.

26. Zhao Z, Cheng J, Ng W. Truth discovery in data streams: A single-pass probabilistic approach. Proc of CIKM. 2014:1589–1598.



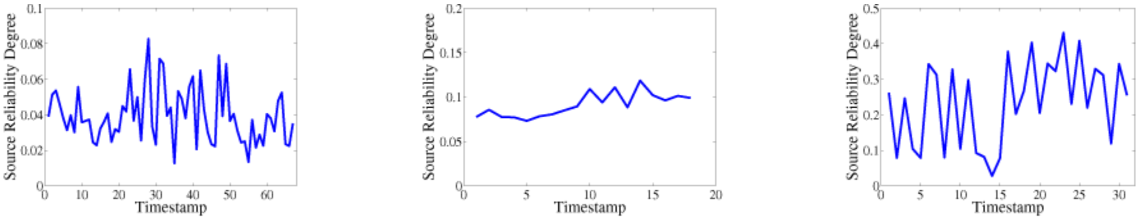
(a) Weather

(b) Stock

(c) Flight

Figure 1.

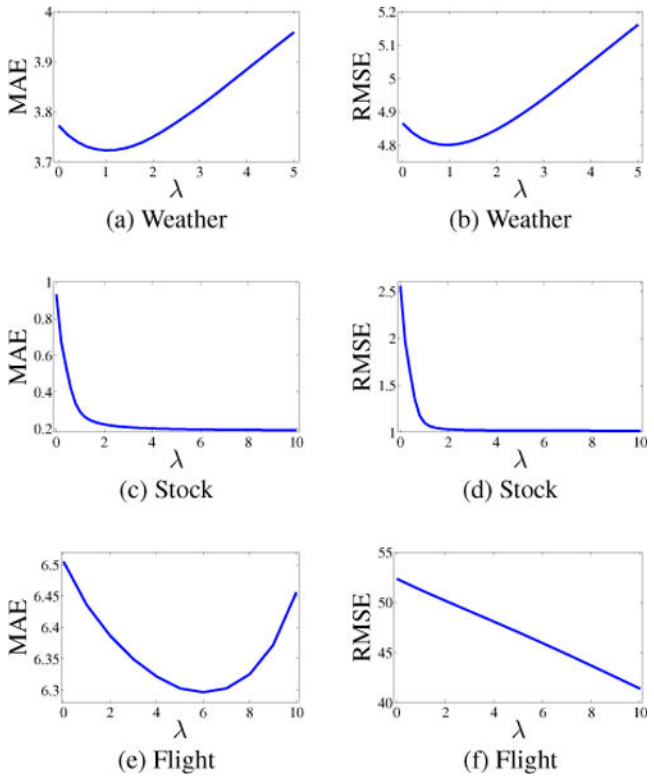
Truth Evolves over Time.



(a) Weather (b) Stock (c) Flight

Figure 2.

The Observed Source Reliability Changes over Time.



(a) Weather

(b) Weather

(c) Stock

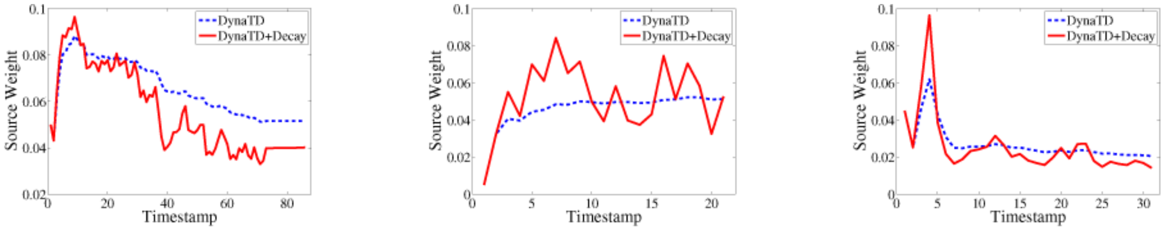
(d) Stock

(e) Flight

(f) Flight

Figure 3.

Performance w.r.t. Smoothing Parameter λ



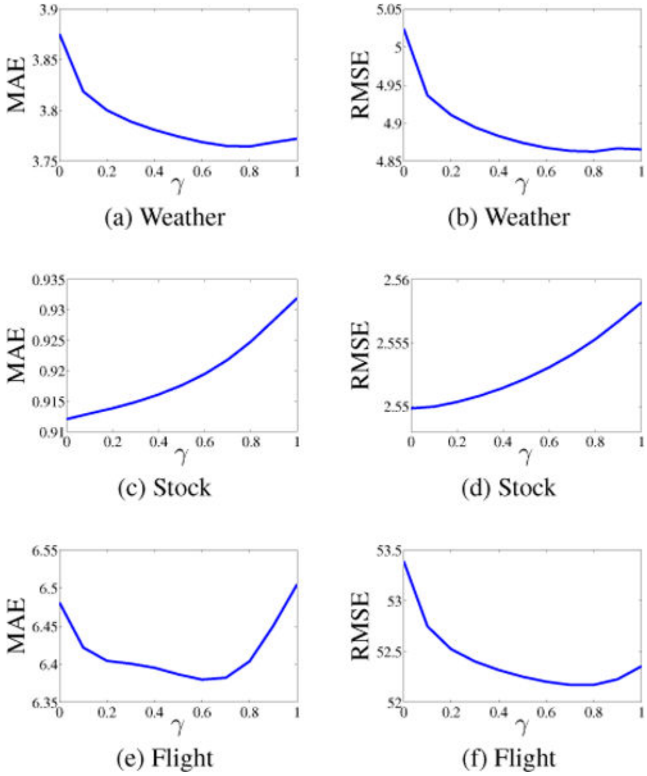
(a) Weather

(b) Stock

(c) Flight

Figure 4.

Source Weight Comparison



(a) Weather

(b) Weather

(c) Stock

(d) Stock

(e) Flight

(f) Flight

Figure 5.

Performance w.r.t. Decay Parameter γ

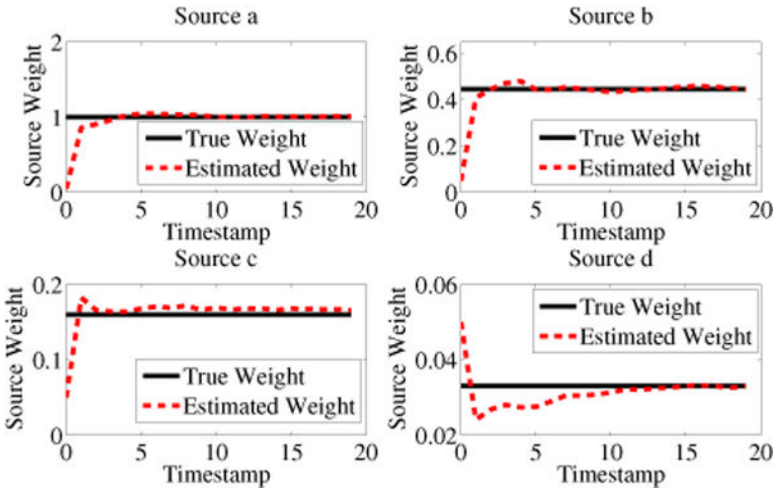


Figure 6.
Source Weight Convergence

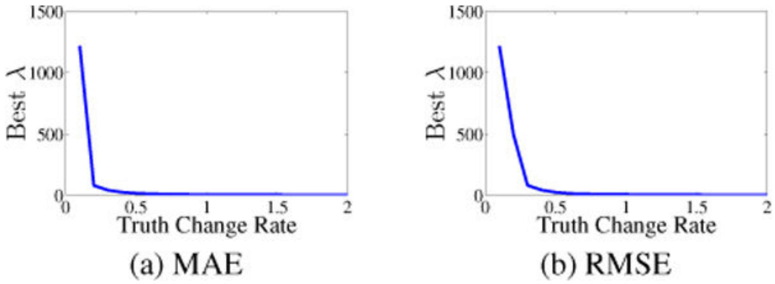


Figure 7.

The effect of Smooth Factor λ

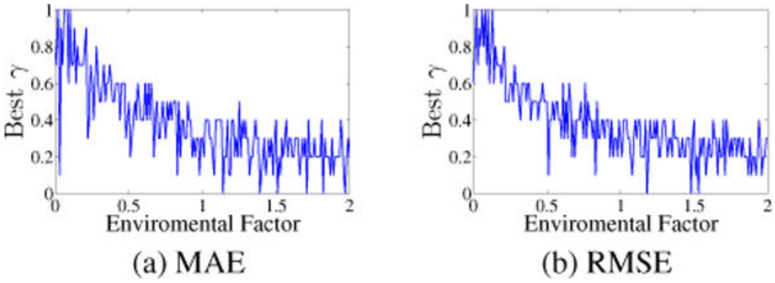


Figure 8.

The effect of Decay Factor γ

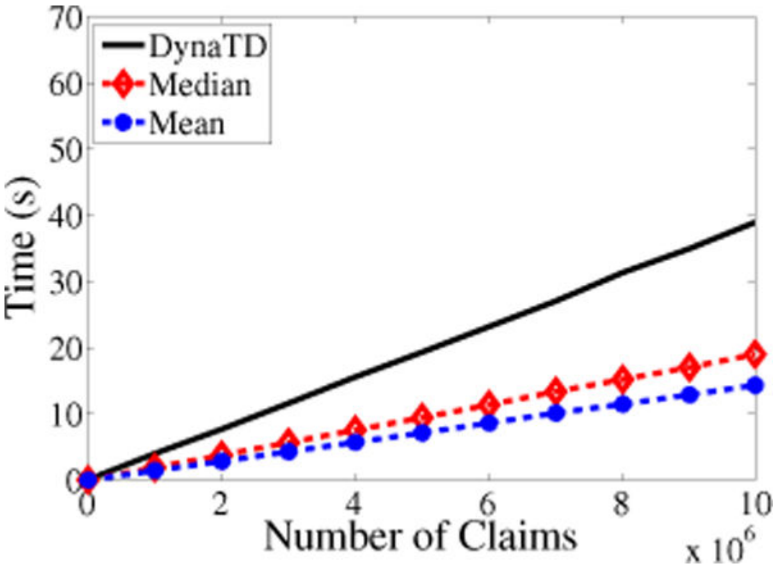


Figure 9.

Running Time w.r.t. Number of Claims

Table 1

Notations

Notation	Definition
$v_{o,t}^s$	information for object *o* from source *s* at time *t*
$\mathscr{X}_t^s$	set of all the information from source *s* at time *t*
c_t^s	number of claims provided by source *s* at time *t*
$\hat{v}_{o,t}^*$	the aggregated result for object *o* at time *t*
$\mathscr{X}_t^*$	set of aggregated results at time *t*
w_s	weight of source *s*
$\mathscr{W}$	set of all the source weights
$e_{o,t}^s$	error on object *o* made by source *s* at time *t*
e_t^s	set of errors made by source *s* at time *t*
$e_{1:t}^s$	errors of source *s* from time 1 to time *t*
$e_{1:t}$	errors of all the sources from time 1 to time *t*

Table 2

Performance Comparison

Method	Weather Dataset			Stock Dataset			Flight Dataset		
	MAE	RMSE	Time(s)	MAE	RMSE	Time(s)	MAE	RMSE	Time(s)
DynaTD	3.7722	4.8655	0.2659	0.9320	2.5582	1.5226	6.5050	52.3547	7.8491
DynaTD+smoothing	3.7230	4.8007	0.2801	0.1563	0.7885	1.5320	6.2966	45.9228	7.6452
DynaTD+decay	3.7646	4.8627	0.2450	0.9121	2.5498	1.5260	6.3798	52.2029	7.6444
DynaTD+All	**3.7093**	**4.7857**	0.2849	**0.1481**	**0.7845**	1.5210	**6.2309**	**45.8221**	7.6165
Mean	4.9903	6.4982	0.1167	0.9438	2.5357	**0.5701**	8.2575	51.5801	**3.1808**
Median	4.9004	6.5752	**0.1038**	0.9133	2.6897	0.5919	7.8097	58.2965	3.2109
GTM	4.7463	6.1749	1.1480	0.8863	2.5365	6.6506	7.6506	51.6956	30.6503
CRH	3.9493	5.1038	0.6371	0.8398	2.6234	9.1807	8.6980	58.1676	38.9449
CATD	4.6310	6.0178	3.1769	0.8952	2.5527	14.0154	8.6453	53.0601	81.1288
TruthFinder	4.7573	6.6054	32.6739	0.8933	2.6589	494.9164	8.9633	62.6080	598.6464
AccuSim	4.5369	6.2482	33.6196	0.8963	2.6595	505.6787	7.5661	60.8732	622.4706
Investment	4.8999	6.8722	0.9750	0.9136	2.6889	7.3751	6.7258	60.1398	25.5679
3-Estimates	4.8804	6.8350	31.9051	0.9096	2.6908	224.3704	7.2561	60.7468	1186.7492