Dynamic Truth Discovery on Numerical Data

Shi Zhi* Fan Yang* Zheyi Zhu* Qi Li* Zhaoran Wang† Jiawei Han*
**Computer Science*, *University of Illinois at Urbana-Champaign*
{shizhi2, fyang15, zzhu27, qili5, hanj}@illinois.edu
†*Industrial Engineering & Management Sciences*, *Northwestern University*
zhaoran.wang@northwestern.edu

***Abstract*—Truth discovery aims at obtaining the most credible information from multiple sources that provide noisy and conflicting values. Due to the ubiquitous existence of data conflict in practice, truth discovery has been attracting a lot of research attention recently. Unfortunately, existing truth discovery models all miss an important issue of truth discovery — the truth evolution problem. In many real-life scenarios, the latent true value often keeps changing dynamically over time instead of staying static. We study the dynamic truth discovery problem in the space of numerical truth discovery. This problem cannot be addressed by existing models because of the new challenges of capturing time-evolving source dependency in a continuous space as well as handling missing data on the fly.**

We propose a model named *EvolvT* for dynamic truth discovery on numerical data. With the hidden Markov framework, EvolvT captures three key aspects of dynamic truth discovery with a unified model: truth transition regularity, source quality, and source dependency. The most distinguishable feature of the modeling part of EvolvT is that it employs Kalman filtering to model truth evolution. As such, EvolvT not only can principally infer source dependency in a continuous space, but also can handle missing data in a natural way. We establish an expectation-maximization (EM) algorithm for parameter inference of EvolvT and present an efficient online version for the parameter inference procedure. Our experiments on real-world applications demonstrate its advantages over the state-of-the-art truth discovery approaches.

***Index Terms*—Truth Discovery, Kalman Filtering, Streaming Data**

I. Introduction

Nowadays, people can access a vast amount of information from all kinds of sources every day. However, the information sources may provide mistaken information due to lack of expertise, malicious purposes, broadcasting failures, staleness, etc. Moreover, the individual information sources may have missing records or only provide partial information. To get the complete and precise information, it is necessary to leverage multiple information sources. Take an analysis on the quality of information for the stock market as an example. Market capitalization is one of the key information that investors are interested in. Based on the statistics on the collected market capitalization data from 55 sources during 2011[1], we find that for the 1000 stocks of interest, the sources provide market capitalization information for 95.6% of them on average, and 19.7 days on average out of 21 trading days. Meanwhile, there is only one source, called “pc-quote”, that provides information for all stocks during July 2011, but it ranks at the bottom in terms of precision. This drives us to develop an efficient algorithm to discover the reliable information with complete coverage along time.

One naive approach to aggregate these 55 sources and resolve their conflicts for the stock information is to take the median/mean for each entry as the final result. However, the median/mean may still suffer from the mistakes provided by the low quality sources. Its results can be further improved by *truth discovery* approaches. In truth discovery, the source reliability plays an important role, as the reliable sources are more likely to provide correct information. However, the source reliability is usually unknown in real-world applications, neither the trustworthy information. Thus, the general principles are introduced here to estimate both the trustworthy information and source reliability: if the piece of information is from a reliable source, then it is more trustworthy, and the source that provides trustworthy information is more reliable. The truth discovery methods have witnessed success in resolving conflicts in various scenarios [1], [2] and various domains such as information extraction [3], [4], event detection [5], [6] and online health community [7].

However, most of the existing truth discovery algorithms are proposed to work on static data, but the batch algorithms do not properly solve the dynamic truth discovery problem such as the aforementioned example on stock information. It is mainly due to three reasons. First is the efficiency issue. Since the data arrives sequentially, there are two ways to adapt the batch algorithm. One is to wait until all data are collected. It is usually unrealistic as the data stream may never end, but the timely results are needed. The other way is that when new data arrive, re-run the batch algorithms all over on the data from the first timestamp to current timestamp. However, it can be very costly on large-scale data.

Second, we observe that in real-world scenarios, the truths of the same entity at consecutive timestamps are correlated in many cases, which raises the challenge of temporal dependency [8], [9]. However, the batch truth discovery algorithms ignore such correlations. For example, we examine the auto-correlation of market capitalization of 100 stock symbols, and with a significant portion of time the auto-correlation is larger than 0.2. This auto-correlation provides us an evidence that using estimated true values from the history can benefit the estimation of current true values. Moreover, the historical data can help to solve the “missing data problem”, a common case that the sources, or even all sources, fail to record any

[1] http://lunadong.com/fusionDataSets.htm

observations to certain objects at a certain timestamp. If we can correctly estimate the correlation along time, we may alleviate the missing data problem using the latent truths from the previous timestamps as a smoother for the current estimation and making a prediction on it.

Thirdly, inspired by [10], [11], we also observe that the source quality would evolve over time and the source quality consistency assumption of existing methods does not hold any more. Also, as illustrated in [12], sources may copy from each other, or get information from similar sources. Similar situations appear when multiple stock information websites are actually operated by the same head company. This may be harmful to validate the truthfulness when happening among bad sources. Thus, understanding the source dependency can help us to better estimate the truth.

The evolving truth discovery problem has obtained more attention in recent years [10]–[16]. However, they either work on categorical data or have strong assumptions on the data properties and need substantial understanding of the data to choose proper parameters and models (More details can be found in Section II). In this work, a new truth discovery method for evolving numerical data based on hidden Markov model is developed for dynamic scenarios. We take into account evolving truth, source quality, source correlation in our model, yet do not have the hassle of setting parameters. The case study shows its effectiveness compared with previous methods. Our contributions are summarized as follows:

- We model the dynamic truth discovery problem on numerical data into a hidden Markov framework. Based on Kalman filter, our model captures truth transition from the past to the current timestamp in a principled way. Such a design not only makes our model able to balance historical smoothing and current observations for accurate truth inference, but also robust to missing observations.
- We establish an expectation-maximization (EM) algorithm for parameter inference of EvolvT, and propose an efficient O(T) online version by blocked Kalman filtering and smoother. The inference procedure allows the model to infer latent truth in a fast and online manner.
- We have conducted extensive experiments on real-world datasets. The experimental results demonstrate that our model not only achieves significantly better accuracies compared to existing methods for dynamic truth discovery, but also is robust to missing data.

II. Related Works

Truth discovery problem has been studied to resolve the conflict among sources. The essential idea is by incorporating the source quality, information from high-quality sources is more trustworthy, and should weighs more in truth estimation. It is first formally introduced by Yin et. al. [1], which models source quality as a single score and iteratively updates source quality and truth value in an unsupervised way. The idea is shared in some early works [17]–[19]. These algorithms focus on categorical truth.

Then more papers propose new truth discover algorithms in various scenarios. CRH [20] is an integrated framework for both numerical and categorical truths, by defining different loss function and combines them into an optimization object function, and [21] proposes a new framework for long-tail phenomenon. Probabilistic graphical model is also widely adopted in truth discovery domain, where the latent truth and source quality can be modeled as latent variables. Expectation-Maximization (EM) is naturally used to infer the truth and source quality [22]–[26].

There have been some recent works that solve the subproblems discussed in our model. As for source dependency analysis, [27] is the first to consider copy-cats among sources and integrate it into the inference of truth values. [28] also directly models the real-value truth by putting normal distribution assumption on the observation given the latent truth. Since both of them are focused on static data, it does not make use of the correlation of truths between timestamps.

In works that study the temporal change of truth, Pal et. al. [14] model the history of the objects using hidden semi-Markovian process. Li et. al. [10] provide an incremental framework that updates truths and source weights as new data come, but the temporal correlation is captured by manually prefixed parameters, which are estimated from the data in our model in contrast. Li et. al. [11] propose a framework to reduce the times of weight update when applying dynamic truth algorithms [10], [20]. Yao et. al [13] utilize time series models to help the truth estimation, but it requires substantial knowledge on the data to choose a proper time series model. These methods also assume that sources are independent, which is not true in many real cases. We propose a more general model which considers the source dependency.There are several models proposed for categorical data for dynamic truth discovery [12], [15], [16], but these models cannot be easily adapted to handle continuous data. Our method models the temporal correlation of numerical truth and incorporates source dependency, and we further propose an efficient and effective online estimation algorithm.

III. The model

In this section, we first formulate the evolving truth discovery problem using hidden Markov model, where the truths are the hidden variables. Then, we provide Kalman filter and smoother with the efficient blocked parameter updating under expectation maximization (EM) schema. We provide an effective data pre-processing method and an online algorithm with pre-train steps for practical use.

A. Problem Formulation

1) Notation: **Input.** Let $\mathcal{O} = \{o_1, o_2, \ldots, o_O\}$ be the objects that we are interested in. Let $\mathcal{S} = \{s_1, s_2, \ldots, s_S\}$ be the set of sources. Numerical observations of O objects can be collected from S sources at each timestamp $t \in \{1, 2, \ldots, T\}$ ($t \in 1 : T$ [2]). Let $v^i_{j,t}$ represent the observation provided

[2] $a : b$, where a, b are arbitrary integers and $b \geq a$, represents the set $\{a, a+1, \ldots, b\}$ in this work.

by the source s_i of the object o_j at the t-th timestamp. For convenience, we denote all the observations from source s_i at time t as $\mathcal{X}_t^i$, that is, $\mathcal{X}_t^i = \{v_{j,t}^i\}_{o_j \in \mathcal{O}}$. Further, the size of this set is denoted as $c_t^i = |\mathcal{X}_t^i|$.

Output. Let $\mu_{j,t}$ be the truth for object o_j at time t, and the output is the whole set of truths at time t, denoted by $\mathcal{T}_t = \{\mu_{1,t}, \mu_{2,t}, ..., \mu_{O,t}\}$.

Besides inferring truths, truth discovery methods can also estimate source reliability degrees. Let Σ denote the source covariance matrix. Its diagonal element σ_i^2 can be interpreted as the source quality of source s_i. Its off-diagonal elements $\sigma_{i,i'}$ can be used to measure the source dependency between source s_i and $s_{i'}$.

2) Task definition: We formally define the task in this paper as follows.

Inferring truth. Until timestamp T, we collect observations $\{v_{j,t}^{1:S}\}$ of O objects from S sources. Our goal is to infer the true values $\{\mu_{j,1:T}\}$ for each object o_j by aggregating observations $\{v_{1:O,1:T}^{1:S}\}$ of O objects from S sources.

Inferring source quality. Besides inferring truths, we also infer source covariance matrix during timestamp $t \in 1:T$ given observations of all objects from all sources.

Inferring other parameters. We can also infer sources dependency $\sigma_{i,i'}$ in our work. It is useful to capture the effects of copying among sources.

B. Batch Solution: Hidden Markov Model

We first build a hidden Markov model for evolving truth discovery when we can observe all data until time T. Figure 1 shows the dynamics of truths and observations. We use Markov process to model the dynamics of truths with the underlying temporal correlations. We assume first-order Markov property on latent truths, where the current latent truth depends on the latent truth from the last timestamp. The current observations of objects only depends on the latent truth of current timestamp, and observations are conditionally independent along the timeline. Assume that we observe the same set of objects along the timeline. $\boldsymbol{\mu}_t = (\mu_{1,t}, \dots, \mu_{O,t})^T$ denotes the vector of latent truths, where the superscript T represents the transpose of a vector or a matrix. Then, the dynamics of truths can be written in the following form

$$\boldsymbol{\mu}_{t+1} = A\boldsymbol{\mu}_t + \boldsymbol{\omega}_t \tag{1}$$

, where $t \in \{1, 2, \dots, T\}$, $\boldsymbol{\mu}_t$ is the latent vector we aim to estimate, and $A \in \mathbb{R}^{O \times O}$ is the transition matrix of latent truth for all timestamps. Eq. 1 means that the current truth is the linear combination of truths from last timestamp, plus an error term. If A is a diagonal matrix, truth of each object will only depend on the previous true value of the same object, whereas if A is non-diagonal, truth of an object will also depend on the true values of other objects at last timestamp.

Let the initial distribution of $\boldsymbol{\mu}_1$ follow a multivariate normal distribution

$$\boldsymbol{\mu}_1 \sim Normal(\boldsymbol{\pi}_1, V_1) \tag{2}$$

, where $\boldsymbol{\pi}_1 \in \mathbb{R}^{O \times 1}$ and $V_1 \in \mathbb{R}^{O \times O}$ are the mean vector and covariance matrix of the initial state, respectively.

Then, let the error vector $\boldsymbol{\omega}_t \in \mathbb{R}^O$ follow multivariate normal distribution

$$\boldsymbol{\omega}_t \sim Normal(\mathbf{0}, \Gamma) \tag{3}$$

, where $\Gamma \in \mathbb{R}^{O \times O}$ is the covariance matrix of the truths of all objects. It, together with the transition matrix A, reflects the dependency among objects.

We assume the observation for object o_j from source s_i at time t, i.e., $v_{j,t}^i$, fully depends on the truths at time t, i.e. $\boldsymbol{\mu}_t$, following the multivariate normal distribution

$$\mathbf{v_t} = C\boldsymbol{\mu}_t + \boldsymbol{\epsilon}_t, \quad C = I_O \otimes \mathbf{1_S}. \tag{4}$$

. $\mathbf{v_t} \in \mathbb{R}^{OS \times 1}$ is the stacked observation vector including all objects from their sources at timestamp t. It is firstly ordered by sources, then by objects. Specifically, $\mathbf{v_t} = (v_{1,t}^1, \dots, v_{1,t}^S, v_{2,t}^1 \dots, v_{O,t}^S)^T$. $\otimes$ is Kronecker product. $\mathbf{1_S} \in \mathbb{R}^{S \times 1}$is a vector in which all elements are ones. $I_O \in \mathbb{R}^{O \times O}$ is the identity matrix thus $C \in \mathbb{R}^{OS \times O}$. Implied by Eq. 4, we assume the mean of the observations are the centered at the truths. If $v_{j,t}^i$ is not provided by source s_i, we regard it as *missing data*. Note that the missing data is prevalent in the real truth discovery cases, where not all sources will provide observations for every object.

We assume the error vector $\boldsymbol{\epsilon}_t$ follows multivariate normal distribution independently as follows

$$\boldsymbol{\epsilon}_t \sim Normal(0, \Pi) \tag{5}$$

, where the diagonal blocks of Π are denoted by Σ, and the off-diagonal blocks all 0. The diagonal elements of Σ $(\sigma_1^2, \sigma_2^2 \dots, \sigma_S^2)$ are interpreted as source quality, because large variability of the observations could most likely be from unreliable sources. The off-diagonal elements of Σ represent the correlation between each pair of sources. The observations from the same source or each pair of sources will share the same diagonal or non-diagonal parameters from Σ. Thus the source quality σ_i^2 and source dependency $\sigma_{i,i'}$ are actually estimated from observations on all objects. If we assume sources are not correlated to each other, $\sigma_{i,i'}$ can also set to 0. Otherwise, they can be estimated from the data. We will discuss both the diagonal and non-diagonal cases in Section III.

The hidden Markov model is composed of Eq. (1)-(5). The parameters to estimate are transition matrix A, objects covariance matrix Γ, initial truth parameters $\boldsymbol{\pi}_1$, V_1 and source quality covariance matrix Σ. Given parameter values, Kalman filter and smoother are typical methods [29] to infer the latent truth at timestamp t by estimating $E(\mu_t|v_{1:t})$ (filtering) or $E(\mu_t|v_{1:T})$ (smoothing). The essential difference is that when estimating the expected value of current latent truth, filtering only uses previous observations, while smoothing uses observations from the past, the present and the future.

1) Model inference: To estimate the parameters in the model, both maximum likelihood and Bayesian methods are available. We refer [30] for background discussions on hidden Markov models. In this work, we adopt EM algorithm to

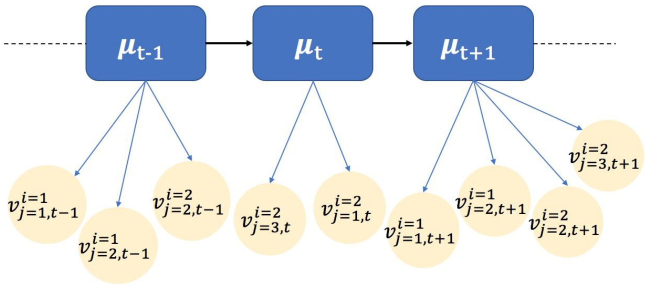


Fig. 1. Hidden Markov Model with observations from multiple sources

estimate the Kalman filter and smoother, the time-variant truths and the parameters iteratively. In most cases, not all sources will provide observations for all objects at any timestamp t. It is prevalent that one source does not provide any observation about some objects at timestamp t. Here, we treat unavailable data as missing data. We adopt the EM algorithm [31] to infer the truths, source quality and dependency information based on our model with missing data.

The joint log likelihood of the complete data $\boldsymbol{\mu}_{1:T}, \mathbf{v_{1:T}}$ can be written in the following form

$$\log P(\boldsymbol{\mu}_{1:T}, \mathbf{v_{1:T}}) = -\frac{1}{2}\log|V_1| - \frac{1}{2}(\boldsymbol{\mu}_1 - \boldsymbol{\pi}_1)V_1^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\pi}_1)$$
$$-\frac{T}{2}\log|\Gamma| - \frac{1}{2}\sum_{t=2}^{T}(\boldsymbol{\mu}_t - A\boldsymbol{\mu}_{t-1})^T\Gamma^{-1}(\boldsymbol{\mu}_t - A\boldsymbol{\mu}_{t-1})$$
$$-\frac{TO}{2}\log|\Sigma| - \frac{1}{2}\sum_{t=1}^{T}(\mathbf{v_t} - C\boldsymbol{\mu}_t)^T(I_O \otimes \Sigma^{-1})(\mathbf{v_t} - C\boldsymbol{\mu}_t) \tag{6}$$

.

2) E-step:: We use $\mu_{t|\tau}$ to denote the conditional expectation $E(\mu_t|v_{1:\tau})$, $V_{t|\tau}$ to denote the conditional covariance matrix $Var(\mu_t|v_{1:\tau})$ and $V_{t,t-1|\tau}$ to denote the conditional cross-covariance matrix $Cov(\mu_t, \mu_{t-1}|v_{1:\tau})$ for $t, \tau \in \{1, 2, \dots, T\}$. Starting from $t = 1$, we have the following Kalman filter forward recursions at r-th round

$$\begin{aligned} \mu_{t|t-1} &= A_{\langle r\rangle}\mu_{t-1|t-1} \\ V_{t|t-1} &= A_{\langle r\rangle}V_{t-1|t-1}A_{\langle r\rangle}^T + \Gamma_{\langle r\rangle} \\ \mu_{t|t} &= \mu_{t|t-1} + K_t(\mathbf{v}_t^* - C^*\mu_{t|t-1}) \\ V_{t|t} &= V_{t|t-1} - K_tC^*V_{t|t-1} \\ K_t &= V_{t|t-1}C^{*T}(C^*V_{t|t-1}C^{*T} + \Pi_{\langle r\rangle})^{-1} \end{aligned} \tag{7}$$

, where $\mathbf{v}_t^*$ is the vector by entering zeros in the $\mathbf{v}_t$ if the object is not observed and C^* is the matrix by zeroing out the corresponding row of the matrix C in Eq. (4). $A_{\langle r\rangle}$, $\Gamma_{\langle r\rangle}$ and $\Pi_{\langle r\rangle}$ are the parameters estimated from r-th round of M-step. The Kalman gain K_t is deducted by minimizing the trace of the covariance matrix $V_{t|t}$. The estimation of current timestamp $\mu_{t|t}$ is the combination of the prediction from previous timestamp and the current observations from all sources, and Kalman gain automatically balances these two parts.

Eq. (7) also reflects the advantages of the use of source quality. If we assume latent truths from different objects are conditional independent at time t, and sources are independent, the terms $C^*V_{t|t-1}(C^*)^T$ and Σ in K_t will be diagonal. When the j-th source is more reliable, i.e. the corresponding variance σ_j^2 in Σ is small, the entries related to s_j in K_t will be large. It would put more weight of the observations from s_j on the estimation of latent truth, and also put more weight in reducing the uncertainty, i.e. $V_{t|t}$.

The initial state prediction is $\mu_{1|0} = \boldsymbol{\pi}_{1,\langle r\rangle}$ and $V_{1|0} = V_{1,\langle r\rangle}$. Starting from $t = T$, we have Kalman smoother backward recursions at r-th round of 1 to T

$$\begin{aligned} \mu_{t-1|T} &= \mu_{t-1|t-1} + J_{t-1}(\mu_{t|T} - A_{\langle r\rangle}\mu_{t-1|t-1}) \\ V_{t-1|T} &= V_{t-1|t-1} + J_{t-1}(V_{t|T} - V_{t|t-1})J_{t-1}^T \\ J_{t-1} &= V_{t-1|t-1}A_{\langle r\rangle}^T(V_{t|t-1})^{-1} \\ V_{t-1,t-2|T} &= V_{t-1|t-1}J_{t-2}^T \\ &\quad + J_{t-1}(V_{t,t-1|T} - A_{\langle r\rangle}V_{t-1|t-1})J_{t-2}^T \end{aligned} \tag{8}$$

with initial value $V_{T,T-1|T} = A_{\langle r\rangle}V_{T-1|T-1} - K_TC^*A_{\langle r\rangle}V_{T-1|T-1}$.

3) Blocked Kalman filter and smoother in E-step: In the E-step, we use Kalman filter and Kalman smoother to estimate the covariance matrix and the cross-covariance matrix which will be used in M-step. If large numbers of objects are considered together, the dimension of $\boldsymbol{\mu}_t$ will be high. The matrix calculation in E-step will take huge memory and turn out to be computationally expensive. If objects are not correlated, or only correlated in their own group, the computational cost may be greatly reduced. Here, we provide a blocked Kalman filter and smoother when the following conditions are satisfied. (1) Given truths at time $t-1$, μ_{t-1}, truths in different blocks are independent at timestamp t. (2) Observations in different blocks are independent given truths at any timestamp t. Then, conditional latent truth of block b, i.e. $\mu_t^b|v_{1:t}$ and $\mu_t^b|v_{1:T}$, are independent among different blocks. That is, we can implement Kalman filter and Kalman smoother in E-step for decomposed blocked objects independently. Then we can update the blocks of $E(\mu_t|v_{1:t})$ and $E(\mu_t|v_{1:T})$ independently by conducting the same Kalman filtering and smoothing Eq. (7), (8) based on block related observations $v_{block,t}^{1:S}$ and parameters.

4) M-step:: In the M-step, we maximize the conditional expectation of log likelihood in Eq. (6) with respect to the latent truths. We add prior distribution of source quality matrix Σ in log likelihood. We can update parameters at r-th iteration as follows.

If we assume sources are independent, Σ will be a diagonal matrix with elements $\{\sigma_{1:S}^2\}$. We set the prior distribution of σ_i^2 as independent inverse gamma distribution,

$Inv - Gamma(\alpha_i, \beta_i)$, and σ_i^2 is in the form

$$\begin{aligned}\sigma^2_{i,\langle r+1\rangle} &= \frac{2\beta_i + \sum_{j=1}^{O}\sum_{t=1}^{T} E((v^i_{j,t} - \mu_{j,t})^2 | v_{1:T})}{2(\alpha_i + 1) + \sum_{t=1}^{T} c^i_t} \\ &= \frac{2\beta_i + \sum_{j=1}^{O}\sum_{t=1}^{T} ({v^i_{j,t}}^2 - 2v^i_{j,t}\mu_{j,t|T} + \mu^2_{j,t|T} + V_{j,t|T})}{2(\alpha_i + 1) + \sum_{t=1}^{T} c^i_t}\end{aligned} \tag{9}$$

, where $\mu_{j,t|T}$ and $V_{j,t|T}$ are the element of object o_j in $\mu_{t|T}$ and $V_{t|T}$ obtained in E-step, respectively.

If sources are dependent, we set the prior distribution of Σ inverse Wishart distribution, $\mathcal{W}(\Phi, \nu)$, and $\Sigma_{\langle r+1\rangle}$ is in the form

$$\Sigma_{\langle r+1\rangle} = \frac{\sum_{t=1}^{T}\sum_{j=1}^{O} D_{jt}\Sigma^i_{jt}D_{jt} + \Phi}{T \times O(T \times O + S + \nu + 1)} \tag{10}$$

, where $D_{jt} \in \mathbb{R}^{S\times S}$ is the permutation matrix that switches missing values in the observations of o_j, i.e. $\mathbf{v}_{j,t} = (v^1_{j,t}, v^2_{j,t}, \ldots, v^S_{j,t})^T$, to the end in order. That is, $D_{jt}\mathbf{v}_{j,t} = (\mathbf{v}^{(1)}_{j,t}, \mathbf{v}^{(2)}_{j,t})$ where $\mathbf{v}^{(1)}_{j,t}$ is the observed portion and $\mathbf{v}^{(2)}_{j,t}$ is the unobserved portion, where (1) and (2) are the lengths of the two vectors. The corresponding entries of o_j matrix C in Eq. (4) is permuted to $(C^{(1)}_{jt}, C^{(2)}_{jt})^T = D_{jt}$. The expression of covariance matrix of $(\mathbf{v}^{(1)}_{j,t}, \mathbf{v}^{(2)}_{j,t})$ is as follows

$$\Sigma^i_{jt} = \left\{ \begin{array}{cc} \Sigma^{11}_{jt} & \Sigma^{12}_{jt} \\ \Sigma^{21}_{jt} & \Sigma^{22}_{jt} \end{array} \right\} \tag{11}$$

, where

$$\begin{aligned}\Sigma^{11}_{jt} &= (v^{(1)}_{j,t} - C^{(1)}_{jt}\mu_{j,t|T})(v^{(1)}_{j,t} - C^{(1)}_{jt}\mu_{j,t|T})^T \\ &\quad + C^{(1)}_{jt}(V_{j,t|T})(C^{(1)}_{jt})^T \\ \Sigma^{12}_{jt} &= (\Sigma^{21}_{jt})^T = \Sigma^{11}_{jt}(\Sigma^{11}_{\langle r\rangle})^{-1}\Sigma^{12}_{\langle r\rangle} \\ \Sigma^{22}_{jt} &= \Sigma^{22}_{\langle r\rangle} - \Sigma^{21}_{\langle r\rangle}(\Sigma^{11}_{\langle r\rangle})^{-1}\Sigma^{12}_{\langle r\rangle} \\ &\quad + \Sigma^{21}_{\langle r\rangle}(\Sigma^{11}_{\langle r\rangle})^{-1}\Sigma^{11}_{jt}(\Sigma^{11}_{\langle r\rangle})^{-1}\Sigma^{12}_{\langle r\rangle}\end{aligned} \tag{12}$$

, and the r-th iteration of source covariance matrix Σ with respect to $(\mathbf{v}^{(1)}_{j,t}, \mathbf{v}^{(2)}_{j,t})$ is denoted as $\left\{ \begin{array}{cc} \Sigma^{11}_{\langle r\rangle} & \Sigma^{12}_{\langle r\rangle} \\ \Sigma^{21}_{\langle r\rangle} & \Sigma^{22}_{\langle r\rangle} \end{array} \right\}$

The initial state parameters π_1 and V_1 are estimated as follows

$$\pi_{1,\langle r+1\rangle} = \mu_{1|T}, \quad V_{1,\langle r+1\rangle} = V_{1|T} \tag{13}$$

.

The transition matrix A is estimated in the following way

$$A_{\langle r+1\rangle} = (\sum_{t=2}^{T} V_{t,t-1|T} + \mu_{t|T}\mu^T_{t-1|T})(\sum_{t=2}^{T} V_{t-1|T} + \mu_{t-1|T}\mu^T_{t-1|T})^{-1} \tag{14}$$

. If A is blocked matrix following the rules in Section III-B3, the blocks in A can be calculated in the same way of Eq. (14) using corresponding blocked objects information in $V_{t,t-1|T}, \mu_{t|T}, \mu_{t-1|T}$ and $V_{t-1|T}$.

The covariance matrix of the truths Γ is updated as follows

$$\begin{aligned}\Gamma_{\langle r+1\rangle} =& \frac{1}{T-1}(\sum_{t=2}^{T} V_{t|T} + \mu_{t|T}\mu^T_{t|T} \\ & - A_{\langle r+1\rangle}\sum_{t=2}^{T} V_{t,t-1|T} + \mu_{t|T}\mu^T_{t-1|T})\end{aligned} \tag{15}$$

. If Γ is blocked matrix following the rules in Section III-B3, the blocks in Γ will be calculated in the same way of Eq. (15) using corresponding block information in $V_{t,t-1|T}$, $\mu_{t|T}$, $\mu_{t-1|T}$, $V_{t|T}$ and A.

C. Comparison with previous methods

To explicitly illustrate the power of our model, we compare the deduction of our model with the evolving truth model [10]. In [10], source quality is modeled as source weight in the following way

$$w_i = \frac{2(\alpha_i - 1) + \sum_{t=1}^{T}\gamma^{T-t}c^i_t}{2\beta_i + \theta\sum_{j=1}^{O}\sum_{t=1}^{T}(\gamma^{T-t}(v^i_{j,t} - \mu_{j,t})^2)} \tag{16}$$

, where (α_i, β_i) is the parameter of Gamma prior distribution, $\mu_{j,t}$ is estimated using source weighted sum of observations, γ is the decay factor used to adjust the source weight, and θ is the regularization parameter. Compared with the update of our source quality σ^2_j in Eq. (9), we can see that our source quality parameter σ^2_j is similar to the inverse of source weight w_j in [10], i.e. $\sigma^2_j \approx 1/w_j$. Therefore, if we assume truths are independent of each other, sources are independent of each other, and observations given the truths are conditionally independent, our model will reduce to a similar solution of the model in [10]. The key advantage of our model, in addition to the power to model dependencies, is to use the population variance adjusted by Kalman smoother rather than the population variance adjusted by a fixed decay factor.

The updated $\mu_{j,t}$ in [10] is

$$\mu_{j,t} = \frac{\sum_{i=1}^{n_i} w_i v^i_{j,t} + \lambda\hat{v}^*_{j,(t-1)}}{\sum_{i=1}^{n_i} w_i + \lambda} \tag{17}$$

. Comparing with the update of $E(\mu_{j,t}|v_{1:t})$ in Eq. (7) and $E(\mu_{j,t}|v_{1:T})$ in Eq. (8) where K_t is Kalman Gain matrix related to source quality matrix Σ, we can see both Eq. (7) and (8) balance the new observation and previous state estimation. However, our balance is dynamic based on the estimated Kalman Gain. In [10], the balancing parameter λ is predefined and fixed.

Moreover, by relaxing the independence assumptions, our model is sufficiently general and flexible to provide other estimations such as source correlation and object correlation.

D. Data Preprocessing

The detection of outliers is important in making an accurate estimation of truth and source quality. Implied by Eq. (2), (3) and (5), we utilize the normal distribution to model the truths and observations. In most cases, the outliers of observations are unlikely to be the truth in most practical cases. The normalization of observations is also necessary especially we

Algorithm 1: Data Preprocessing

```
Data: The observation vector $v_{j,t}$ for object $o_j$ from $S$ sources
      at time $t$.
{Outlier Detection:}
$med$=nanmedian($v_{j,t}$) ;      // calculate the median
without missing value.
$diff$=abs($v_{j,t} - med$) ;   // calculate the absolute
deviations between $v^i_{j,t}$ and its median.
$med\_abs\_dev$=nanmedian($diff$) ;      // calculate the
median of absolute deviations.
if $med\_abs\_dev == 0$ then
    for $i$ in 0:(S-1) do
        if $diff[i] == 0.0$ or isnan($diff[i]$)==True then
            $out\_lier[v_{j,t}[i]] \leftarrow False$
        else
            $out\_lier[v_{j,t}[i]] \leftarrow True$
        end
    end
else
    for $i$ in 0:(S-1) do
        $modified\_zscore[i] = 0.6745 * diff[i]/med\_abs\_dev$
        if $modified\_zscore[i] > \delta$ then
            $out\_lier[v_{j,t}[i]] \leftarrow True$
        else
            $out\_lier[v_{j,t}[i]] \leftarrow False$
        end
    end
end
{Normalization:calculating z-scores:}
The normalized z-score of $v^i_{j,t}$ from source $s_i$ is
$(v^i_{j,t} - mean(v_{j,t}))/std(v_{j,t})$
```

take into account a large number of objects. The expressions in Eq. (9)-(12) implies that if one object has an extreme large scale compared to others, this object would dominate the source quality, making the source quality estimation biased.

We implement data preprocessing step before EM algorithm.To detect outliers of the observations of each object at time t, we use the median absolute deviation to find outliers [32]. After removing all outliers, we normalize the observations to its z-scores as the input of the EM algorithm. Specifically, for each object o_j at time t, normalized $v^i_{j,t}$ is ($v^i_{j,t}$-mean($v^{1:S}_{j,t}$))/std($v^{1:S}_{j,t}$). When there are no sufficient data at time t for object o_j, we aggregate observations from consecutive timestamps in a fixed length sliding window to normalize the data. The detailed data preprocessing algorithm is described in Algorithm 1. The observation whose modified z-score is larger than the threshold parameter δ will be classified as outliers and removed from the estimation. Following [32], we use 0.6745 as the constant multiplier in the modified z-score and set the threshold $\delta = 3.5$.

E. Online Solution: $EvolveT(T^*)$

The key challenge of integrating streaming data is that (1) we are not able to use the future data to estimate the present, and (2) using all the data at each timestamp is time-consuming. Here, we propose an online solution. When a new data point arrives, we update the estimation of current truth with the parameters at previous timestamp using Kalman filtering one step further without running Kalman smoother backwards in E-step, and update the parameters in M-step in an incremental way with O(1) complexity. The reason we can update the estimates incrementally is that Kalman forward recursion is defined in an incremental way, and parameters in Eq. (9), (10), (13), (14) and (15) contain only accumulated term $\sum_{t=1}^{T}(\cdot)$. Thus, this online version can incrementally update the truths and parameters sequentially with time complexity $O(T)$.

Though future data is not accessible during the estimation, the historical records can help to better initialize the parameters. We first run the batch solution with both Kalman filtering and smoothing until EM step converges for the first few timestamps, followed by the online version. We call it the pre-train step.

We summarize our entire algorithm in Algorithm 2. We call it *Evolving Truth* algorithm, denoted by $EvolvT(T^*)$. T^* denotes the history length to run the batch-mode version. We will compare the performance of different T^* in the experiments section. For historical T^* timestamps data, we iteratively update Kalman recursions $\mu_{t|T^*}$, $V_{t|T^*}$ based on Eq. (7), (8) and parameters based on Eq. (9), (10), (13), (14) and (15) until all of them converge. At the pre-train step, we assume the source quality matrix Σ to be consistent for stable initialization.After first T^* pre-train timestamps, we update truths and parameters in an incremental way and only scan the remaining data once from time T^* to T. Truths are updated based on Kalman forward recursions $\mu_{t|t}$, $V_{t|t}$ in Eq. (7). Parameters are updated following Eq. (9), (10), (13), (14) and (15), but replacing smoothing estimates $(\mu_{t|T}, V_{t|T}, V_{t-1,t|T})$ by filtering estimates $(\mu_{t|t}, V_{t|t}, V_{t-1,t|t})$. Here, the source quality matrix Σ is actually changing over time.

IV. Experiments

In this section we evaluate the effectiveness and efficiency of our model. All the experiments are conducted on a laptop with 4 GB RAM, 1.4 GHz Intel Core i5 CPU, and OS X 10.11.6, with Python 3.6.

A. Experiment Setup

1) Datasets: We adopt the market capitalization data, fight arrival data, weather forecast data and pedestrian counts data to evaluate the algorithms.

- **Market capitalization data (Stock)**. The market capitalization data consist of 1000 stock symbols from 55 sources on trading days in July 2011 [33]. The ground truth for evaluation is built on NASDAQ100 stocks collected by taking the majority values provided by five stock data providers: nasdaq.com, yahoo finance, google finance, bloomberg and MSN finance.
- **Weather forecast data (Weather)**. We collect the highest temperature weather forecast data for 88 U.S. cities from 6 websites: wunderground.com, worldweatheronline.com, openweathermap.org, DarkSky.net, APIXU.com and yahoo.com. The data last for 2 months, from June 8th, 2017 to August 8th, 2017 (61 days). We also collect the actual highest temperature (°F) observations as the

Algorithm 2: Evolving Truth algorithm, $EvolvT(T^*)$

```
Data: The observation vector $v_{j,t}$ for object $o_j$ from $S$ sources
      at time $t$.
{Data preprocessing step:};
Follow Algorithm 1;
{Training step:(if $T^* > 0$)}
while $||para\langle r\rangle - para\langle r-1\rangle||_2 > \delta_{em}$;   // $para\langle r\rangle$ is
the vector of all parameters to be
estimated at $r$-th iteration.
do
    E-Step:
    for block b in 1:B tcp*B: number of independent blocks do
        for $h$ in $1 : T^*$ do
            update filtering estimates $\mu^b_{h|h}$ and $V^b_{h|h}$ based on
            Eq. (7) in blocks
        end
        for $h$ in $(T^*, T^* - 1, \ldots, 1)$ do
            update smoothing estimates $\mu^b_{h|T^*}$, $V^b_{h|T^*}$ and
            $V^b_{T^*-1,T^*|T^*}$ based on Eq. (8) in blocks
        end
    end
    M-Step:
    Update source quality $\Sigma$ based on Eq. (9) or (10)
    Update initial state parameters $\pi_1$ and $V_1$ based on Eq. (13)
    Update transition matrix $A$ based on Eq. (14)
    Update truth covariance matrix $\Gamma$ based on Eq. (15)
end
{Incremental updating step:};
for t in T*:T*+T do
    for block b in 1:B do
        update filtering estimates $\mu^b_{t|t}$ and $V^b_{t|t}$ based on Eq.
        (7) in blocks
    end
    Repeat M-Step above and replace smoothing estimates by
    Kalman estimates if they are used to update parameters
end
```

ground truth.

- **Pedestrian data (Pedestrian)**. Given by Dublin City Council [3] , the data consist of daily pedestrian counts of four streets in 2015. There are many sources that may provide the pedestrian counts, such as sensors from traffic light, surveillance cameras, infrared beam counters, etc. Since it is not easy to collect the real data from the aforementioned sources, we simulate six different sources by varying the Gaussian noise level with different variances. We use data from November 1st to December 31st and set the variances as 1, 1.44, 1.96, 2.56, 3.24, 4, respectively.
- **Flight arrival data (Flight)**. The flight arrival time contains 3000 flights from 38 sources over 1-month period (31 days) (December 2011) from [33]. We normalize the arrival time into minutes. Ground truth for evaluation is provided by corresponding airline websites.

2) Evaluation metrics: We use mean absolute error (MAE) and root mean square error (RMSE) to measure the correctness of all the truth discovery algorithms.

3) Compared methods: A large portion of the existing methods are only working on stable truths. To demonstrate the advantages of modeling dynamic truths, we treat the streaming data in a batch way to run these methods.

The following algorithms are designed for categorical truth, where distance between answers is not measurable in Euclidean distance. Thus, the true answer is selected from one of the candidate answers. Since our paper is focused on numerical truth, we treat numerical truth as one of the candidate to fit in the models. **TruthFinder** [1] and **AverageLog** [17] iteratively estimate the truths and source quality using additive or multiplicative ways. In **Investment** and **PooledInvestment** [17], each source uniformly invests its quality among the answers they provide, and its quality is a weighted sum of the credibility of those answers. **3Estimates** [34] extends the framework further by introducing an additional factor, i.e. difficulty of the question when evaluating the truths and source quality.

For these methods modeling continuous data, we list them as follows. **Median** and **Mean** are two naive methods that do not consider the source quality and take median and mean at each timestamp independently. **GTM** [28] is a probabilistic graphical model designed for continuous data in static truth discovery. **DynaTD+ALL** is an incremental method with both decay and smoothing factors and the state-of-the-arts algorithm reported in [10]. We use the same data preprocessing algorithm for it.

For our proposed method, *EvolvT*, we develop a set of different versions by varying the pre-train step T^*. We take three sets of (T^*) to illustrate their difference. $EvolveT(0)$ is a fully online version starting with no historical data and randomly initialized parameters. Parameters such as source quality, transition matrix, and object covariance matrix are updated along time. $EvolvT(t) (t \neq 0)$ is to use historical t timestamps to initialize the parameters, then conduct the $O(T)$ algorithm to infer the truths left. $EvolvT$ is a fully batch-mode version where we can observe all streaming data and estimate the truths at all timestamps with fixing source quality along time. For all baseline methods, we use the suggested parameters, initialization and convergence conditions in the original papers. For our model, the parameters are set to $\nu = 2$, $\Phi = (S + \nu + 1 * I)$, $\alpha_i = \beta_i = 10$ for each source s_j, $\boldsymbol{\mu}_1 = \mathbf{0}$, V_1 and A to identity matrix. We evaluate $Evolve$ in the batch mode without any historical data. As for T^*, we set it to [0, 5, 10], and evaluate all methods from the 11th timestamp.

B. Experimental Results

In this section, we empirically demonstrate the effectiveness and efficiency of our algorithm, $EvolvT(T^*)$, and illustrate the impact of different factors to the estimation performance.

1) Dynamic Truth Inference: Table I shows the performance comparison of the baseline models, our batch version $Evolve$ and online version $Evovle(T^*)$ on stock, flight, weather and pedestrian datasets. We observe that on stock, weather and pedestrian datasets, the performance follows the same pattern while on the flight dataset, the performance is different. We will specifically discuss the flight dataset next.

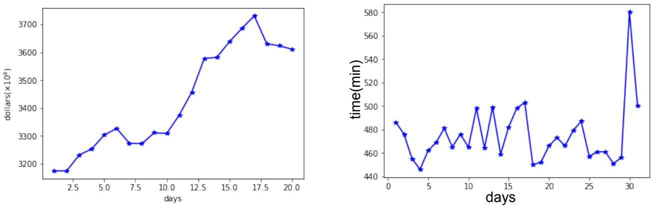
[3] https://data.gov.ie/dataset/pedestrian_footfall

TABLE I

Performance Comparison of baseline methods, our batch algorithm $EvolvT$ and online algorithm $EvolvT(T^*)$ on four datasets. Stock, weather and pedestrian datasets evolve smoothly along time, while flight dataset have more sudden changes due to flight delays and cancels.

	Stock			Flight			Weather			Pedestrian		
Methods	MAE	RMSE	Time(s)	MAE	RMSE	Time(s)	MAE	RMSE	Time(s)	MAE	RMSE	Time(s)
TruthFinder	3.49	9.44	98.12	1.19	5.85	50.04	2.81	3.33	2.11	1.79	2.28	1.11
3Estimates	6.05	27.70	208.99	28.90	119.90	146.75	3.34	4.42	1.76	1.79	2.28	1.11
AverageLog	3.60	9.70	31.53	0.98	4.85	22.20	2.53	3.65	0.78	1.79	2.28	0.05
Investment	6.05	27.50	44.48	28.91	119.90	30.76	3.34	4.42	0.93	1.79	2.28	0.05
PooledInv	3.23	11.19	50.69	**0.39**	**3.13**	44.68	2.55	3.36	1.41	1.79	2.28	0.08
Median	2.97	13.29	39.93	2.55	6.40	45.44	2.49	3.20	3.08	0.79	0.95	0.18
Mean	4.60	19.26	57.87	5.78	10.75	62.19	2.46	3.25	4.42	0.87	1.01	0.36
GTM	2.24	11.83	103.28	2.70	3.31	125.82	2.55	3.35	0.70	0.98	1.19	0.10
GTM+ours	1.63	9.27	131.32	3.37	6.73	150.35	2.46	3.53	2.75	0.80	0.96	0.11
DynaTD+All	2.05	8.01	8.82	2.89	8.48	0.23	2.97	4.45	10.23	0.78	1.01	0.16
EvolvT	2.05	7.97	3.63	**2.54**	**4.31**	140.1	2.52	3.78	54.78	0.72	0.95	0.12
EvolvT(0)	1.96	8.35	4.01	2.80	9.18	0.49	2.64	3.49	2.33	0.72	0.93	0.03
EvolvT(5)	1.86	7.54	7.93	3.25	8.53	3.85	2.48	3.28	11.38	**0.69**	**0.89**	0.04
EvolvT(10)	**1.54**	**6.91**	3.85	3.31	8.61	22.89	**2.42**	**3.20**	22.89	0.70	0.90	0.06

In general, the methods for numerical data perform better than those for categorical data, and dynamic models perform better than the static numerical models. The reason is that when the person is assigned to count the number of pedestrians, the smaller count difference he makes, the more we should trust this person. While for methods working at categorical data, they treat the counters with the same confidence level regardless of the count difference. On the other hand, due to the inherent property of stock, weather and pedestrian datasets, truths along time have dependency, and considering the historical truths can benefit the estimation of current truth. Also, our method is parameter-free in estimating the current truth compared to DynaTD+All. We notice that our methods are significantly better than all the other algorithms. With pretrain step of historical records, our method gets better performance than random initialization $EvolvT(0)$. The batch-mode version performs slightly better than online version, but would cost more time due to the convergence requirements.



(a) Market capitalization of stock AAPL in 20 days

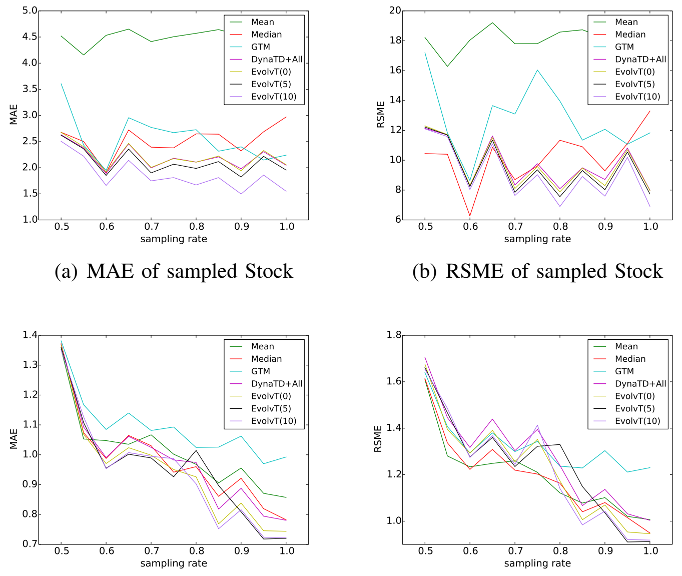
(b) Arrival time of flight AA3979 in 30 days

Fig. 2. Observations of stock AAPL market capitalization change and flight AA3979 arrival time change. The arrival time of flight AA3979 has been normalized to minutes.

2) *Discussion on flight dataset:* The first thing to consider is whether we treat flight arrival time as numerical value or categorical value. One interesting finding is that though flight arrival time is real-valued parameter, whose difference can be measured using Euclidean distance, the actual distribution of flight arrival time observations do not follow a normal distribution. Due to the delays or accidental incidents, some websites may not immediately update the information timely, or not update it until the flight actually arrives, leading to scattered distribution centered at some discrete value. Thus, the optimal solution, with high probability, will be one of the time provided by certain sources. Table I shows that TruthFinder, AverageLog, and PooledInvestment have better performance than the Gaussian-based method, GTM. Secondly, it is important to know when the historical truth can benefit the current truth estimation. Comparing GTM with $EvolvT$, we find that the only dataset GTM can win our model on is flight dataset. The common part is that both methods make Gaussian assumption on the observation from each source, but the difference lies in whether to use historical truth as a smoother. If we take a further look at the truth fluctuation in Figure 2, we find that the market capitalization of stocks usually evolve smoothly, while the arrival time of the flight changes sharply due to flight delays or cancels.

3) *Efficiency:* We also report the running time of all the algorithms. For efficiency, the running time (s) of our proposed model is close to Median and Mean. Iterative methods usually takes 10 times more times to converge at each timestamp, while our single-pass O(T) version gains better performance with even shorter running time. Our algorithm can largely reduce the running time by single-pass O(T) algorithm without loss in performance. The reason behind it is that algorithm can keep track of the parameter information from the history such that source quality, object dependency and prediction power are properly inherited, while the batch-mode algorithms do not use the history information, making the running time long.

4) *Missing observations:* To illustrate the robustness of our algorithm to missing observations, we randomly remove some of the observations of sources. Since we have shown in Table I that the methods specifically working on numerical truth work consistently better than those on



(a) MAE of sampled Stock

(b) RSME of sampled Stock

(c) MAE of sampled Pedestrian

(d) RSME of sampled Pedestrian

Fig. 3. MAE, RSME of sampled Stock, Pedestrian datasets

categorical truth, we only run part of the baseline algorithms for further comparison on missing observations. The sampling rate demonstrates the proportion that we keep out of all the data. We range the sampling rate in $[0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 1.]$. Figure 3 shows the MAE and RSME changes along with the sampling rate. We can observe that our method $Evolve(10)$ performs the best in terms of MAE in most cases on both stock and pedestrian datasets. One interesting observation is that Median performs well in terms of RSME when we only keep half of the data on both datasets. It is possible that as the number of data points significantly drops, there are no sufficient data to estimate the parameters of the truth discovery models.

5) Effectiveness of data pre-processing: To demonstrate the effectiveness of our data pre-processing step, we replace the outlier detection and normalization step of GTM, denoted by GTM+ours. GTM considers the points that do not center around the mean of the observations within k standard deviation, while our data pre-processing uses median and absolute deviation. Table I shows that our pre-processing reduces the errors of GTM on stock, weather and pedestrian datasets.

6) Source dependency: We compare the performance with and without the source dependency assumption. Table II lists the performance of dependent sources models. We find that assuming all sources are correlated would not have a good estimation on truths. The major reason is that the number of parameters is the square of the number of sources in the dependency case. With limited number of observations, it is hard to estimate all the parameters with high confidence.

For source dependency study, we rank all the sources by their quality at the last timestamp, i.e. $1/\sigma_i^2$, from top to low, and separate them into three different groups. The sources in highest quality group are listed in Figure 4(a). The mappping from source index to name is listed in Table III. (2:bloomberg, 21:tmx-quotemedia, 10:investoguide, 24:yahoo-finance) have highest correlation in this group. Figure 4(b) and 4(c) show groups with second lowest and lowest source quality. (4:cnn-money, 15:optimum) are highly correlated. (1:barrons[4], 12:marketwatch[5], 17:screamingmedia) are highly correlated. We further check the the origins of these three websites, and find barrons and marketwatch are all owned by Dow Jones & Company, which is an American publishing and financial information firm, and screamingmedia is pre-owned by Dow Jones & Company[6]. With high possibility, these websites get information from an identical source.

TABLE II

Performance on stock, flight and weather datasets of the source dependency models. The pre-train step is set to be 0, 5, 10 timestamps, respectively.

	Stock		Flight		Weather	
Methods	MAE	RMSE	MAE	RMSE	MAE	RMSE
dep-EvolvT(0)	2.59	9.24	3.18	15.78	3.60	2.44
dep-EvolvT(5)	2.58	8.74	3.17	16.23	3.78	11.38
dep-EvolvT(10)	2.56	8.65	3.50	16.3	3.40	22.94

As for the flight data, we plot the quality of all sources in Figure 4(d). (6:flights, 7:businesstravellogue, 8:flylouisville) are highly correlated, and (9:flightview, 10:panynj, 11:gofox, 12:foxbusiness, 13:allegiantair, 14:boston) are highly correlated. It is possible that they copy the flight information from each other, or achieve information from similar sources.

For the weather data, we find that within the 6 sources, only APIXU and worldweatheronline are highly correlated, with correlation score=0.9. We further check on the web and find that World Weather Online acquires Apixu platform [7]. Thus, the source correlation is validated, showing that our method can effectively detect the source dependencies.

V. Conclusions

We study the problem of modeling dynamic truth on streaming numerical data. To address the challenges of modeling time-evolving source dependencies and handling missing data, we propose a model that combines hidden Markov model and Kalman filtering. We demonstrate that the model is capable of capturing different key aspects for dynamic truth discovery, and also provide analytical solutions to the parameter inference of the model. Our proposed different versions for parameter inference reduce the computational cost and allow the model to infer latent truth in an online and dynamic setting. Experiments on the real-world datasets demonstrate the effectiveness of our model and its robustness to missing data.

Acknowledgement

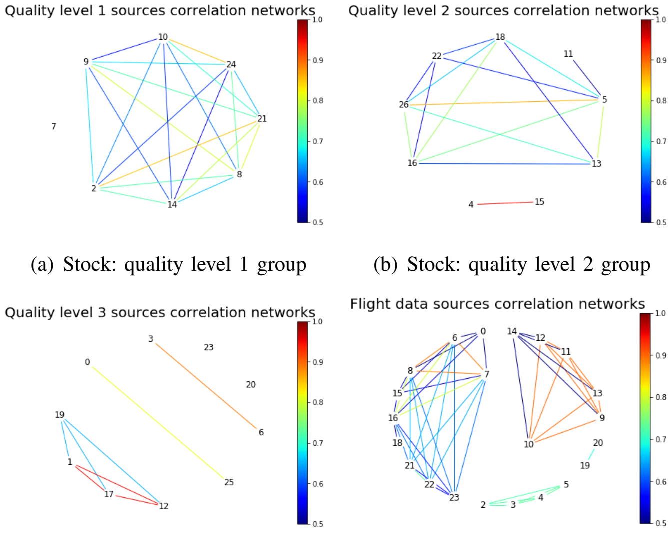
Research was sponsored in part by U.S. Army Research Lab under Cooperative Agreement No. W911NF-09-2-0053

[4] https://en.wikipedia.org/wiki/Barron%27s_(newspaper)

[5] https://en.wikipedia.org/wiki/MarketWatch

[6] http://adage.com/article/btob/dow-jones-sells-screaming-media-yellowbrix/276811/

[7] https://www.facebook.com/worldweatheronline/posts/1160440014032686


Quality level 1 sources correlation networks
10
9
24
7
21
8
2
14
Quality level 2 sources correlation networks
18
22
11
26
5
16
13
4
15
Quality level 3 sources correlation networks
3
23
0
20
6
19
25
1
17
12
Flight data sources correlation networks
6
0
14
12
11
8
7
15
13
16
9
18
10
20
21
19
22
23
5
2
3
4
1.0
0.9
0.8
0.7
0.6
0.5

(a) Stock: quality level 1 group

(b) Stock: quality level 2 group

(c) Stock: quality level 3 group

(d) Flight: all sources

Fig. 4. Group-level source dependency of stock and flight datasets. The bar on the right shows the intensity of the dependency level. Color red indicates the highest and color blue means lowest dependency.

TABLE III
LIST OF SOURCES OF FLIGHT AND STOCK DATASETS WITH SOURCE INDEX AND ITS NAME

stock	0:barchart, 1:barrons, 2:bloomberg, 3:cio-com, 4:cnn-money, 5:eresearch-fidelity-com, 6:finapps-forbes-com, 7:finviz, 8:fool, 9:google-finance, 10:investorguide, 11:marketintellisearch, 12:marketwatch, 13:msn-money, 14:nasdaq-com, 15:optimum, 16:pc-quote, 17:screamingmedia, 18:smartmoney, 19:thestree, 20:tickerspy, 21:tmx-quotemedia, 22:updown, 23:wallstreetsurvivor, 24:yahoo-finance, 25:ycharts-com, 26:zacks
flight	0:aa, 1:flightexplorer, 2:airtravelcenter, 3:myrateplan, 4:helloflight, 5:flytecomm, 6:flights, 7:businesstravellogue, 8:flylouisville, 9:flightview, 10:panynj, 11:gofox, 12:foxbusiness, 13:allegiantair, 14:boston, 15:travelocity, 16:orbitz, 17:weather, 18:mia, 19:mytripandmore, 20:flightarrival, 21:flightaware, 22:wunderground, 23:flightstats, 24:quicktrip, 25:world-flight-tracker, 26:ifly, 27:ua, 28:usatoday, 29:CO, 30:flightwise, 31:iad, 32:mco

(NSCTA), DARPA under Agreement No. W911NF-17-C-0099, National Science Foundation IIS 16-18481, IIS 17-04532, and IIS-17-41317, DTRA HDTRA11810026, and grant 1U54GM114838 awarded by NIGMS through funds provided by the trans-NIH Big Data to Knowledge (BD2K) initiative (www.bd2k.nih.gov).

REFERENCES

[1] X. Yin, J. Han, and P. Yu, "Truth discovery with multiple conflicting information providers on the web," *TKDE*, vol. 20, 2008.

[2] H. Xiao, J. Gao, Z. Wang, S. Wang, L. Su, and H. Liu, "A truth discovery approach with theoretical guarantee," in *KDD*, 2016.

[3] D. Yu, H. Huang, T. Cassidy, H. Ji, C. Wang, S. Zhi, J. Han, C. Voss, and M. Magdon-Ismail, "The wisdom of minority: Unsupervised slot filling validation based on multi-dimensional truth-finding," in *COLING*. ACM, 2014.

[4] L. Liu, X. Ren, Q. Zhu, S. Zhi, H. Gui, H. Ji, and J. Han, "Heterogeneous supervision for relation extraction: A representation learning approach," *EMNLP*, 2017.

[5] C. Zhang, G. Zhou, Q. Yuan, H. Zhuang, Y. Zheng, L. Kaplan, S. Wang, and J. Han, "Geoburst: Real-time local event detection in geo-tagged tweet streams," in *SIGIR*, 2016, pp. 513–522.

[6] L. Y. Z. C. W. T. W. Y. G. J. S. L. Zhang Hengtong, Ma Fenglong, "Leveraging the power of informative users for local event detection," in *ASONAM*, 2018.

[7] S. Mukherjee, G. Weikum, and C. Danescu-Mizil, "People on drugs: credibility of user statements in health communities," in *KDD*, 2014.

[8] Z. Wang, F. Han, and H. Liu, "Sparse principal component analysis for high dimensional multivariate time series," in *Artificial Intelligence and Statistics*, 2013, pp. 48–56.

[9] M. Yu, Z. Yang, T. Zhao, M. Kolar, and Z. Wang, "Provable gaussian embedding with one observation," in *NIPS*, 2018, pp. 48–56.

[10] Y. Li, Q. Li, J. Gao, L. Su, B. Zhao, W. Fan, and J. Han, "On the discovery of evolving truth," in *KDD*. ACM, 2015, pp. 675–684.

[11] T. Li, Y. Gu, X. Zhou, Q. Ma, and G. Yu, "An effective and efficient truth discovery framework over data streams." in *EDBT*, 2017, pp. 180–191.

[12] X. Dong, L. Berti-Equille, and D. Srivastava, "Truth discovery and copying detection in a dynamic world," *PVLDB*, 2009.

[13] L. Yao, L. Su, Q. Li, Y. Li, F. Ma, J. Gao, and A. Zhang, "Online truth discovery on time series data," in *Proceedings of the 2018 SIAM International Conference on Data Mining*. SIAM, 2018, pp. 162–170.

[14] A. Pal, V. Rastogi, A. Machanavajjhala, and P. Bohannon, "Information integration over time in unreliable and uncertain environments," in *WWW*. ACM, 2012, pp. 789–798.

[15] D. Y. Zhang, C. Zheng, D. Wang, D. Thain, X. Mu, G. Madey, and C. Huang, "Towards scalable and dynamic social sensing using a distributed computing framework," in *Distributed Computing Systems (ICDCS)*. IEEE, 2017, pp. 966–976.

[16] D. A. Garcia-Ulloa, L. Xiong, and V. Sunderam, "Truth discovery for spatio-temporal events from crowdsourced data," *Proceedings of the VLDB Endowment*, vol. 10, no. 11, pp. 1562–1573, 2017.

[17] J. Pasternack and D. Roth, "Knowing what to believe (when you already know something)," in *COLING*, 2010.

[18] V. Vydiswaran, C. Zhai, and D. Roth, "Content-driven trust propagation framework," in *Proc. of SIGKDD*, 2011.

[19] X. Liu, X. L. Dong, B. C. Ooi, and D. Srivastava, "Online data fusion," *Proceedings of the VLDB Endowment*, vol. 4, no. 11, pp. 932–943, 2011.

[20] 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 *SIGMOD*, 2014.

[21] 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*, 2014.

[22] B. Zhao, B. Rubinstein, J. Gemmell, and J. Han, "A bayesian approach to discovering truth from conflicting sources for data integration," *PVLDB*, vol. 5, no. 6, pp. 550–561, 2012.

[23] G.-J. Qi, C. C. Aggarwal, J. Han, and T. Huang, "Mining collective intelligence in diverse groups," in *WWW*, 2013.

[24] 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 *KDD*, 2015.

[25] S. Zhi, B. Zhao, W. Tong, J. Gao, D. Yu, H. Ji, and J. Han, "Modeling truth existence in truth discovery," in *KDD*, 2015, pp. 1543–1552.

[26] Z. Wang, Q. Gu, Y. Ning, and H. Liu, "High dimensional expectation-maximization algorithm: Statistical optimization and asymptotic normality," *arXiv preprint arXiv:1412.8729*, 2014.

[27] X. Dong, L. Berti-Equille, and D. Srivastava, "Integrating conflicting data: the role of source dependence," *PVLDB*, vol. 2, no. 1, pp. 550–561, 2009.

[28] B. Zhao and J. Han, "A probabilistic model for estimating real-valued truth from conflicting sources," 2012.

[29] A. C. Harvey, *Forecasting, structural time series models and the Kalman filter*. Cambridge university press, 1990.

[30] J. Durbin and S. J. Koopman, *Time series analysis by state space methods*. OUP Oxford, 2012, vol. 38.

[31] R. H. Shumway and D. S. Stoffer, "An approach to time series smoothing and forecasting using the em algorithm," *Journal of time series analysis*, vol. 3, no. 4, pp. 253–264, 1982.

[32] B. Iglewicz and D. C. Hoaglin, *How to detect and handle outliers*. Asq Press, 1993, vol. 16.

[33] X. Li, X. L. Dong, K. Lyons, W. Meng, and D. Srivastava, "Truth finding on the deep web: Is the problem solved?" in *Proceedings of the VLDB Endowment*, vol. 6, no. 2. VLDB Endowment, 2012, pp. 97–108.

[34] A. Galland, S. Abiteboul, A. Marian, and P. Senellart, "Corroborating information from disagreeing views," in *Proc. of WSDM*, 2010.