TextTruth: An Unsupervised Approach to Discover Trustworthy Information from Multi-Sourced Text Data

Hengtong Zhang[1], Yaliang Li[2], Fenglong Ma[1], Jing Gao[1], Lu Su[1]
[1]SUNY Buffalo, Buffalo, NY USA
[2]Tencent Medical AI Lab, Palo Alto, CA USA
{hengtong, fenglong, jing, lusu}@buffalo.edu , yaliangli@tencent.com

ABSTRACT

Truth discovery has attracted increasingly more attention due to its ability to distill trustworthy information from noisy multi-sourced data without any supervision. However, most existing truth discovery methods are designed for structured data, and cannot meet the strong need to extract trustworthy information from raw text data as text data has its unique characteristics. The major challenges of inferring true information on text data stem from the multifactorial property of text answers (i.e., an answer may contain multiple key factors) and the diversity of word usages (i.e., different words may have the same semantic meaning). To tackle these challenges, in this paper, we propose a novel truth discovery method, named "TextTruth", which jointly groups the keywords extracted from the answers of a specific question into multiple interpretable factors, and infers the trustworthiness of both answer factors and answer providers. After that, the answers to each question can be ranked based on the estimated trustworthiness of factors. The proposed method works in an unsupervised manner, and thus can be applied to various application scenarios that involve text data. Experiments on three real-world datasets show that the proposed TextTruth model can accurately select trustworthy answers, even when these answers are formed by multiple factors.

CCS CONCEPTS

• **Information systems → Data mining**;

KEYWORDS

Truth discovery; unsupervised learning; text mining

ACM Reference Format:
Hengtong Zhang, Yaliang Li, Fenglong Ma, Jing Gao, Lu Su. 2018. TextTruth: An Unsupervised Approach to Discover Trustworthy Information from Multi-Sourced Text Data. In *KDD '18: The 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, August 19–23, 2018, London, United Kingdom.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3219819.3219977

1 INTRODUCTION

In the big data era, tremendous data can be accessed on various online platforms, such as *Amazon Mechanical Turk*, *Stack Exchange* and *Yahoo Answers*. However, such multi-sourced data are usually contributed by non-expert online users, thus there may exist errors or even conflicts in the data. Therefore, how to automatically infer trustworthy information (i.e., the truths) from such noisy and conflicting data is a challenging problem.

To address this challenge, truth discovery methods have been proposed [4, 5, 8, 12–15, 19–21, 26, 27, 29, 36, 38, 43], which aim to estimate trustworthy information from conflicting data by considering user reliability degrees. Truth discovery approaches follow two fundamental principles: (1) If a user provides much trustworthy information or true answers, his/her reliability is high; (2) If an answer is supported by many reliable users, this answer is more likely to be true. Though yielding reasonably good performance, most existing truth discovery methods are designed for structured data, and are difficult to be directly applied to *text data*, which are unstructured and noisy. This significantly narrows the application domain of these truth discovery methods, as a large ratio of the multi-sourced data are text. Actually, there are several unique characteristics of natural language that hinder the existing truth discovery methods from being successfully applied to text data.

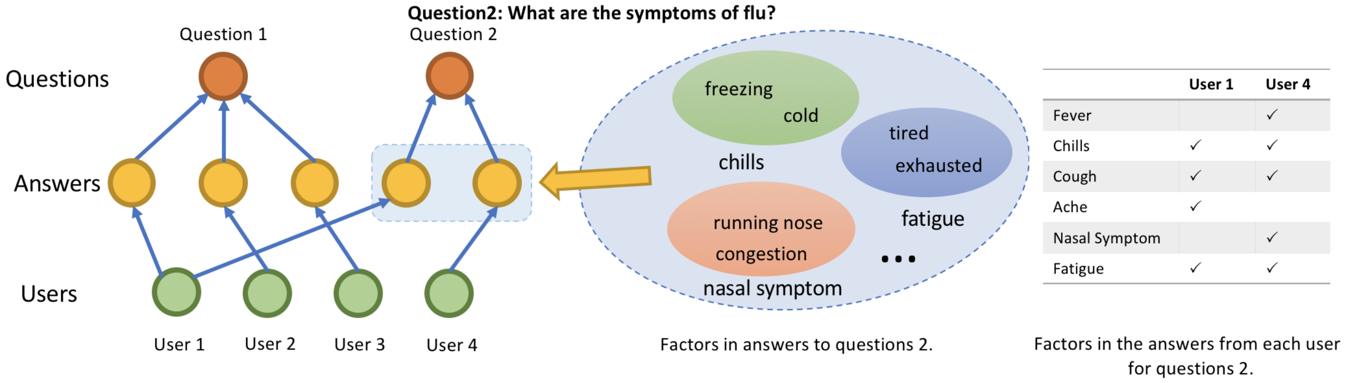
Figure 1 gives an illustration of these two characteristics of text data. First, the answer to a factoid question [1] may be *multifactorial*, and it is usually hard for a given text answer to cover all the factors. For the question '*What are the symptoms of flu?*', the correct answer should contain the following factors: *fever*, *chills*, *cough*, *nasal symptom*, *ache*, and *fatigue*. Even if the answer provided by a user covers two factors, such as *cough* and *chills*, the existing truth discovery methods may determine this answer to be totally wrong and assign a low reliability degree to this user. This is because these methods treat the whole answer as an integrated unit. However, if we take the fine-grained answer factors into consideration, the answer provided by this user is partially correct, which implies that we should give some credits to the user by increasing his/her reliability degree. Thus, how to identify partially correct answers and model factors of text answers is critical for the task of truth discovery on text data.

The second characteristic of text data is the *diversity of word usages*. Answers provided by online users may convey a very similar meaning with different keywords. For example, users may use words such as *tired* or *exhausted* to describe the symptom of *fatigue*. However, existing truth discovery approaches may treat them as

[1] **Note**: This paper merely focuses on finding trustworthy answers for **factoid questions**. Factoid questions are defined as questions that can be answered with **simple facts** expressed in **short text answers**.[11]

KDD '18, August 19–23, 2018, London, United Kingdom

ACM ISBN 978-1-4503-5552-0/18/08…$15.00
https://doi.org/10.1145/3219819.3219977



Factors in answers to questions 2.

	User 1	User 4
Fever		✓
Chills	✓	✓
Cough	✓	✓
Ache	✓	
Nasal Symptom		✓
Fatigue	✓	✓

Factors in the answers from each user for questions 2.

Figure 1: An Illustration of questions, answers, answer factors and keywords. The left diagram illustrates the relationship among questions, answers and users. The middle diagram shows an example of keywords and their answer factors. The right table demonstrates the factors in the answers of user 1 and user 4, respectively.

totally different answers. Thus, it is of great importance to model the diversity among answers in the text data when inferring trustworthy information.

In order to tackle the aforementioned challenges for inferring trustworthy information from text data, in this paper, we propose a model named "*TextTruth*", which takes the keywords in each answer as inputs and outputs a ranking for the answer candidates based on their trustworthiness. Specifically, we first transform the keywords in text answers into pre-trained computable vector representations. Due to the fact that an answer may contain multiple factors, the "answer-level" or coarse-grained representations may not be able to capture the partially correct answers. Thus, we need to convert the whole answer into fine-grained factors. Then, we model the diversity of answers by clustering the keywords with similar semantic meanings. By doing so, we can estimate the trustworthiness of each answer factor instead of the whole answer and infer the correctness of each factor in the answer.

Compared with existing truth discovery methods, the advantages of the proposed TextTruth are two-fold: First, by evaluating the trustworthiness of each answer factor, the proposed model can naturally handle the partial correctness phenomenon of text answers. Second, by modeling answer keywords in the form of vector representations, we can make the factors within the answers computable such that the ubiquitous usage diversity issue on text data is addressed.

Experiments on three real-world datasets demonstrate that the proposed TextTruth model can improve the performance of finding trustworthy answers in text data compared with the state-of-the-art truth discovery approaches. We also provide case studies to demonstrate that the proposed method can provide interpretable labels for answer factors in real-world answers. The major contributions of this paper are as follows:

- We identify the unique challenges of discovering true information from multi-sourced text data, i.e., partially correct answers and word usage diversity.
- We propose a probabilistic model called TextTruth, which can extract fine-grained factors from each answer. Such design can naturally handle the partial correctness of answers.
- The proposed TextTruth model can jointly learn semantic clusters (i.e., factors) for answer keywords and infer the reliability of each user as well as the trustworthiness of each answer factor. The answers can thus be ranked based on the trustworthiness of their factors.
- We empirically show that the proposed model outperforms the state-of-the-art truth discovery methods for the task of answer ranking on three real-world datasets.

The rest of the paper is organized as follows: Section 2 is a survey of related work. In Section 3, we formally define the problem discussed in this paper. Then we describe the proposed TextTruth model, and provide a method for parameter estimation in Section 4. In Section 5, we conduct a series of experiments and case studies on real-world datasets. We conclude the paper in Section 6.

2 RELATED WORK

We survey the related work from three aspects: truth discovery, community question answering and answer selection.

Truth Discovery: The research topic of truth discovery, which aims to identify trustworthy information from conflicting multi-source data, has become a hot topic in recent years. A large variety of methods have been proposed to handle various scenarios such as: different data types [5, 13, 42, 44], source dependencies [5, 15, 43], fine-grained source reliability [20], entity/object dependency [22] and long-tail data [12, 39]. Among them, there are two truth discovery scenarios that are related to the problem studied in this paper. Firstly, as previously discussed, there may exist multiple factors in a text answer. Such setting could be related to the problem of multi-truth discovery [35, 44]. However, there are some significant differences. In [35, 44], the input from each user is structured categorical data. Hence, the methods proposed in these two papers cannot be directly extended to unstructured text data, where answers may be partially correct and contain diverse word expressions. Secondly, there is also some existing work that focuses on unstructured text inputs. For example, [6] specifies a confidence-aware source reliability estimation approach, which takes the SVO triples extracted from webpages as inputs. However, the ultimate goal of that paper is to reduce conflicting information in the process of knowledge base construction, which is different from our paper. In [32, 33], the authors transform twitter texts into structured data and apply truth discovery methods to find trustworthy tweets. However, in [32, 33], the semantic meanings of texts are not taken into consideration during the truth discovery process. In [16, 17], the authors study the task of verifying the truthfulness of fact statements utilizing Web sources. These work and this paper both conduct trustworthiness analysis in the proposed methods. However, the truthfulness verification task is different from ours, and the methods in [16, 17] assume the access to external supporting information that is not required by our proposed method.

To the best of our knowledge, the only previous work that incorporates semantic meanings into the truth discovery procedure is [18]. However, this work can only handle single word answers and the problem settings are different from this paper which handles multi-factor answers.

Collaborative Question Answering: This paper is also related to the problem of collaborative question answering (CQA). The existing work in this field can be categorized into two groups. The first group of work [3, 10] explicitly extracts features from crowdsourced answers and transforms the answer quality estimation task into classification problems or ranking problems. However, this line of approaches usually require high-quality training sets and a variety of useful features to train the model. Such information, unfortunately, is not always available in real-world applications. Another group of methods [40, 45] transform the problem of answer quality estimation into an expert finding problem. These methods infer the quality of answers based on the answer providers. However, these methods require external information on either asker-answerer interactions or explicit features like voting information. The different problem settings and solutions naturally distinguish these work from this paper.

Answer Selection: Answer selection, which aims to choose the most suitable answer from a set of candidate sentences, is an important task in the field of question answering (QA). Traditional answer selection approaches are mainly based on lexical features [37, 41]. Neural networks based models are proposed to represent the meaning of a sentence in a vector space and then compare the question and answer candidates in this hidden space [7, 34], and have shown great improvement in answer selection. In [28, 31], attention mechanism is introduced into answer selection models to enhance the sentence representation learning. However, these models are all supervised. The model proposed in this paper is different from these approaches, as it does not require labeled data for training.

3 PROBLEM FORMULATION

In this paper, we consider a general truth discovery scenario for factoid text questions and answers. Before introducing the problem formulation, we first define some basic terminologies that will be used in the rest of the paper:

Definition 3.1 (Question). A *question* q contains N_q words and can be answered by users.

Definition 3.2 (Answer). An *answer* given by user u to question q is denoted as a_{qu}.

Definition 3.3 (Answer Keyword). *Answer keywords* are domain-specific content words / phrases in answers. The m-th answer keyword of the answer given by user u to question q is denoted as x_{qum}.

Definition 3.4 (Answer Factor). *Answer factors* are the key points of the answers, which are represented as clusters of *answer keywords*. The k-th answer factor in the answers to question q is denoted as c_{qk}.

For each question, there can be different answers provided by different users. These answers may consist of complex sentences with multiple factors and can be partially correct. This setting can support a broad range of text data. Formally, the problem discussed in this paper can be defined as:

Definition 3.5 (Problem Definition). Given a set of *users* $\{u\}_1^U$, a set of questions $\{q\}_1^Q$ and a set of answers $\{a_{qu}\}_{q,u=1,1}^{Q,U}$, where U denotes the number of users and Q stands for the number of questions. The goal of this paper is to extract highly-trustworthy answers and highly-trustworthy key factors in answers for each question.

4 METHODOLOGY

In this section, we first offer an overview of the proposed TextTruth model, and then explain in detail each component of it.

4.1 Overview

When applying truth discovery methods to find the trustworthy answers to complex natural language questions, semantic correlations among answers should be taken into consideration, so that user reliability can be accurately estimated. However, learning accurate vector representations for the whole answers is difficult especially when the context corpus of these answer paragraphs is not sufficiently large. Moreover, due to the complexity of natural language, the meaning of an answer is too complicated to be represented by a single vector. To tackle such challenges, we rely on more fine-grained semantic units (i.e., answer factors) in each answer to determine the trustworthiness of each answer.

In this paper, for each question, we first extract the *keywords* in each answer and learn their vector representations. Then we cluster these word/phrase-level keywords into semantic clusters (i.e., factors). These factors represent all the possible key points in the answers to a question and can be used to determine the trustworthiness of an answer. For the keywords within each cluster, as they share very similar semantic meanings, their trustworthiness should be almost the same. In addition, users may have different reliabilities, which can be reflected in the answers they provided.

Based on the above ideas, we propose a two-step method to estimate the trustworthiness of each answer. In the first step, we specify a probabilistic model to model the generation of keywords with user reliabilities taken into consideration in Section 4.2. The generative model, which consists of three major components, jointly learns the answer factors and their truth label. The generative model first generates a mixture of answer factors and their semantic parameters. After that, the model generates two-fold user reliability variables, which model the comprehensiveness and accuracy of answer factors provided by a specific user. These two variables

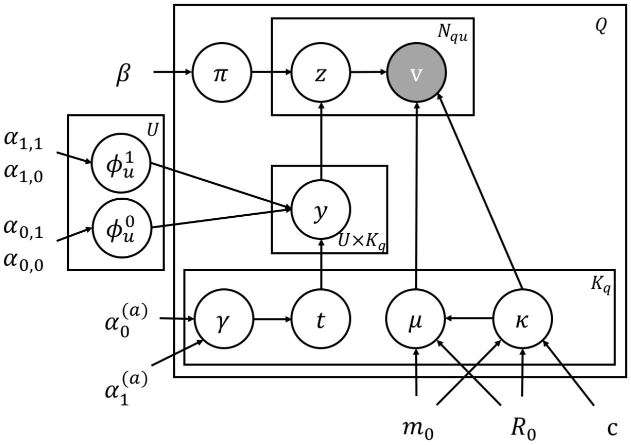


Figure 2: Plate notation for the proposed TextTruth Model. In the graph, white circles denote the latent variables, gray circles stand for the observations, while others stand for the hyper-parameters.

capture a whole spectrum of the user reliability. Finally, the model selects an answer factor based on the semantics, the trustworthiness of the answer factor as well as the reliability of the user that provides the answer, and generate the keyword embedding vector via a von Mises-Fisher (vMF) distribution. The vMF distribution is centralized at the semantic centroid of that answer factor. This way, the design of answer factor and user reliability takes the multifactorial characteristics of answers into consideration. Meanwhile the keyword embedding vector generation also captures the diversity of word usages. These designs make the model capable of capturing the unique characteristics of text data. In section 4.3, we design a straightforward scoring mechanism to evaluate the trustworthiness score of each answer. We provide the parameter estimation of the proposed method in Section 4.4.

4.2 Generative Model

We develop a probabilistic model to jointly learn the answer factors and the truth labels of each answer factor for every question. For an answer a_{qu}, we extract domain-specific answer keywords and get their normalized [2] vector representations [23]. The set of all the vector representations is denoted as $\{\mathbf{v}_{qum}\}$, which also serves as the observation of the probabilistic model. Figure 2 shows the plate notation of the proposed model. The generative model consists of three major components, which are listed as follows:

I. Answer Factor Modeling: The model first generate the mixture of factors according to the Dirichlet distribution, which is commonly used to generate mixture models. Formally, the mixture distribution $\boldsymbol{\pi}_q$ is generated as:

$$\boldsymbol{\pi}_q \sim \text{Dirichlet}(\boldsymbol{\beta}). \tag{1}$$

Here, $\boldsymbol{\beta}$ is a K_q-dimensional vector, where K_q denotes the number of factors in the answers to question q.

For the k-th answer factor under question q, we model its trustworthiness via a binary truth label t_{qk}. Specifically, the model first generates the prior truth probability γ_{qk}. It determines the prior distribution of how likely each factor is to be true, from a Beta distribution with hyper-parameter $\alpha_1^{(a)}$ and $\alpha_0^{(a)}$:

$$\gamma_{qk} \sim \text{Beta}(\alpha_1^{(a)}, \alpha_0^{(a)}). \tag{2}$$

Then the truth label t_{qk} is generated from a Bernoulli distribution with parameter γ_{qk}:

$$t_{qk} \sim \text{Bernoulli}(\gamma_{qk}). \tag{3}$$

Finally, to model the semantic characteristic of each answer factor, we define the centroid parameter $\boldsymbol{\mu}_{qk}$ and concentrate parameter κ_{qk} of vMF distributions from its conjugate prior distribution $\Phi(\boldsymbol{\mu}_{qk},\ \kappa_{qk}; \boldsymbol{m}_0, R_0, c)$ [25], i.e.:

$$\boldsymbol{\mu}_{qk},\ \kappa_{qk} \sim \Phi(\boldsymbol{\mu}_{qk},\ \kappa_{qk}; \boldsymbol{m}_0, R_0, c), \tag{4}$$

where $\Phi(\boldsymbol{\mu}_{qk},\ \kappa_{qk}; \boldsymbol{m}_0, R_0, c)$ is defined as:

$$\Phi(\boldsymbol{\mu}_{qk},\ \kappa_{qk}; \boldsymbol{m}_0, R_0, c) \propto \{C_D(\kappa_{qk})\}^c \exp(\kappa_{qk} R_0 \boldsymbol{m}_0^T \boldsymbol{\mu}_{qk}).$$

Here, $C_D(\kappa) = \frac{\kappa^{D/2-1}}{I_{D/2-1}(\kappa)}$, and $I_{D/2-1}(\cdot)$ is the modified *Bessel function* of the first kind. In practice, there may be few answers that are totally irrelevant to the question. Since the answer factors in irrelevant answers are usually supported by very few users, they will not be regarded as trustworthy.

II. User Reliability Modeling: The reliability of each user is inferred according to the answers they provide. As aforementioned, the answer of a user u may merely cover part of the trustworthy answer factors, and at the same time may consist of untrustworthy answer factors. For instance, some users may only provide the factors that they are very confident of. On the contrary, other users may cover a broad collection of answer factors with different trustworthinesses in their answers. This naturally motivates us to use a two-fold score like [44] to model the reliability of a user.

Suppose we know all the answer factors and their truth labels in advance, for all the questions and their answers, we use TP_u and FP_u to denote the number of trustworthy and untrustworthy answer factors that are covered by the answers from user u (i.e., the number of true positive and false positive factors), respectively. Similarly, we use FN_u and TN_u to denote the number of trustworthy and untrustworthy answer factors that are not covered by the answers from user u (i.e., the number of false negative and true negative factors), respectively. Based on these statistics, we can intuitively use the false positive rate (defined as: $\frac{FP_u}{FP_u+TN_u}$), and the true positive rate (defined as: $\frac{TP_u}{TP_u+FN_u}$) to fully characterize u's reliability.

Let's resume the discussion of the proposed model. During the generative process, the answer factors and their truth labels are not known in advance. Inspired by [44], we also define two-fold user reliability variables ϕ_u^0 and ϕ_u^1 to model the false positive rate and the true positive rate of factors that are covered by the answers of user u. Specifically, for each user u, we generate ϕ_u^0 and ϕ_u^1 from two Beta distributions with hyper-parameters $(\alpha_{0,1}, \alpha_{0,0})$ and $(\alpha_{1,1}, \alpha_{1,0})$, respectively. Here, $\alpha_{0,1}$ and $\alpha_{0,0}$ are the prior false positive count and true negative count, respectively. Similarly, $\alpha_{1,1}$

[2] The normalized vector of $\mathbf{v}$ is given by $\hat{\mathbf{v}} = \frac{\mathbf{v}}{|\mathbf{v}|}$, where $|\mathbf{v}|$ is the $l2$-norm of $\mathbf{v}$.

and $\alpha_{1,0}$ stand for the prior true positive count and the false negative count of each source, respectively. Formally:

$$\begin{aligned} \phi_u^0 &\sim \text{Beta}(\alpha_{0,1}, \alpha_{0,0}) \qquad \text{(False Positive Rate)} \\ \phi_u^1 &\sim \text{Beta}(\alpha_{1,1}, \alpha_{1,0}) \qquad \text{(True Positive Rate).} \end{aligned} \tag{5}$$

III. Observation Modeling: As aforementioned, we use the vector representations of keywords as observations. For the m-th word representation from user u for question q, we specify the following generation process.

Firstly, we define a binary indicator $y_{u,qk}$, which denotes whether the k-th factor of the answers to question q should be covered by user u, based on the reliability of u. For question q, if its truth label $t_{qk} = 1$, the probability of user u covering the k-th factor in its answer follows a Bernoulli distribution with reliability parameter ϕ_u^1. Otherwise, if its truth label $t_{qk} = 0$, the probability follows a Bernoulli distribution with reliability parameter ϕ_u^0. Formally, this process can be written as:

$$\begin{aligned} y_{u,qk} &\sim \text{Bernoulli}(\phi_u^0) \qquad \text{If } t_{qk} = 0, \\ y_{u,qk} &\sim \text{Bernoulli}(\phi_u^1) \qquad \text{If } t_{qk} = 1. \end{aligned} \tag{6}$$

To this point, we have determined the set of answer factors that should be covered by the answer a_{qu}, with the reliability of u taken into consideration.

Then, for the m-th keyword in the answer a_{qu}, its factor label z_{qum} is drawn from a probability density function defined as:

$$P(z_{qum} = k | \boldsymbol{\pi}_q, y_{u,qk}) \propto \begin{cases} \pi_{qk} & \text{if } y_{u,qk} = 1, \\ 0 & \text{if } y_{u,qk} = 0. \end{cases} \tag{7}$$

The density function jointly considers the answer factor mixture distribution and the set of binary indicators $y_{u,q\cdot}$. This means that both semantics and user reliabilities are used to determine the factor label of a specific answer keyword.

With the factor labels determined, the model samples keywords vectors that describe the semantic meaning of its corresponding factor. Note that this procedure should not involve the reliability of a user. The vector representation of a keyword (i.e. $\mathbf{v}_{qum}$) is randomly sampled from a vMF distribution with parameter $\boldsymbol{\mu}_{qk}, \kappa_{qk}$:

$$\mathbf{v}_{qum} \sim \text{vMF}(\boldsymbol{\mu}_{qk}, \kappa_{qk}). \tag{8}$$

Specifically, for a D-dimensional unit semantic vector $\mathbf{v}$ that follows vMF distribution, its probability density function is given by:

$$p(\mathbf{v}_{qum} | \boldsymbol{\mu}_{qk}, \kappa_{qk}) = C_D(\kappa_{qk}) \exp(\kappa_{qk} \boldsymbol{\mu}_{qk}^T \mathbf{v}_{qum}). \tag{9}$$

The vMF distribution has two parameters: the mean direction $\boldsymbol{\mu}_{qk}$ and the concentration parameter $\kappa_{qk} (\kappa_{qk} > 0)$. The distribution of $\mathbf{v}_{qum}$ on the unit sphere concentrates around the mean direction $\boldsymbol{\mu}_{qk}$, and is more concentrated if κ_{qk} is larger. In our scenario, the mean vector $\boldsymbol{\mu}$ acts as a semantic focus on the unit sphere, and produces relevant semantic embeddings around it. The superiority of the vMF distribution over other continuous distributions (e.g., Gaussian) for modeling textual embeddings has also been shown in the field of clustering [1] and topic modeling [9].

The overall generative process is summarized in Algorithm 1.

```
Algorithm 1: Generative Process of TextTruth
for each question $q$ do
    Draw mixture $\pi_q \sim \text{Dirichlet}(\boldsymbol{\beta})$;
    for each answer factor $k$ do
        Draw centroid and concentration:
          $\boldsymbol{\mu}_{qk},\ \kappa_{qk} \sim \Phi(\boldsymbol{m}_0, R_0, c)$;
        Draw truth parameter: $\gamma_{qk} \sim \text{Beta}(\alpha_0^{(a)}, \alpha_1^{(a)})$;
        Draw a truth label: $t_{qk} \sim \text{Bernoulli}(\gamma_{qk})$;
    end
end
for each user $u$ do
    Draw: $\phi_u^0 \sim \text{Beta}(\alpha_{0,1}, \alpha_{0,0})$,    $\phi_u^1 \sim \text{Beta}(\alpha_{1,1}, \alpha_{1,0})$;
end
for each answer $a_{qu}$ do
    for each answer factor $k$ do
        Draw binary label: $y_{u,qk} \sim \text{Bernoulli}(\phi_u^{t_{qk}})$;
    end
    for each keyword $m$ do
        Draw a answer factor label: $P(z_{qum} = k | \boldsymbol{\pi}, y_{u,qk})$;
        Draw keyword embedding:
          $\mathbf{v}_{qum} \sim \text{vMF}(\boldsymbol{\mu}_{qz_{qum}}, \kappa_{qz_{qum}})$;
    end
end
```

4.3 Trustworthy-Aware Answer Scoring

Intuitively, the trustworthiness of an answer should be evaluated by the volume of correct information it provides. Hence, we propose a straightforward scoring mechanism to evaluate the trustworthiness score of each answer. Given the inferred truth labels for each answer factor of question q, we score the answers according to the number of answer keywords in the answer a_{qu} that are related to the factor with truth label $t_{qk} = 1$, i.e.:

$$score_{qu} = \sum_{k=1}^{K_q} N_{u,qk} \mathbb{I}(t_{qk} = 1), \tag{10}$$

where K_q is the number of answer factors for question q, $N_{u,qk}$ denotes the number of keywords that are provided by user u and are clustered into factor k. $\mathbb{I}(t_{qk} = 1) = 1$ if $t_{qk} = 1$, and $\mathbb{I}(t_{qk} = 1) = 0$ if $t_{qk} = 0$. Note that there are many alternative ways of designing scoring functions.

4.4 Model Fitting

In this section, we present the approach to estimating the latent variables and the user reliability parameters.

Latent Variable Estimation: We use MCMC method to infer the latent variables t, z, y and κ. As one can see, the values of y and z have a large impact on the final results, and they may be sensitive to the initialization. Therefore, we make an approximation in latent variable estimation to make the process stable. The detailed steps are specified in the following paragraphs.

First, using conjugate distributions, we are able to analytically integrate out the model parameters and only sample the cluster assignment variable z. This is done as follows:

$$\begin{aligned} & P(z_{qum} = k \mid z_{q,\neg um}, \boldsymbol{\beta}, \boldsymbol{m}_0, R_0, c) \\ & \propto P(z_{qum} = k \mid z_{q,\neg um}, \boldsymbol{\beta}) \\ & \quad \times P(\mathbf{v}_{qum} | \mathbf{v}_{q,\neg um}, z_{qum} = k, \boldsymbol{z}_{q,\neg um}, \boldsymbol{m}_0, R_0, c), \end{aligned} \tag{11}$$

where $\mathbf{v}_{q,\neg um}$ stands for the set of all the keywords in the answers for question q, except the m-th keyword from user u.

Then we can derive the expressions for the two terms in Eq. (11). The first term $P(z_{qum} = k \mid z_{q,\neg um}, \boldsymbol{\beta})$ can be written as:

$$P(z_{qum} = k | \boldsymbol{z}_{q,\neg um}, \boldsymbol{\beta}) \propto N_{qk\neg um} + \beta_k, \tag{12}$$

where $N_{qk,\neg um}$ denotes the number of answer keywords under the k-th factor of question q except current keyword $\mathbf{v}_{qum}$. The second term in Eq. (11) is similar to the form of *vMF Mixture Model*, which can be written as:

$$\begin{aligned} & P(\mathbf{v}_{qum} | \mathbf{v}_{q,\neg um}, z_{qum} = k, \boldsymbol{z}_{q,\neg um}, \boldsymbol{m}_0, R_0, c) \\ & \propto \frac{C_D(\kappa_{qk}) C_D(||\kappa_{qk}(R_0 \boldsymbol{m}_0 + \mathbf{v}_{qk\neg um})||_2)}{C_D(||\kappa_{qk}(R_0 \boldsymbol{m}_0 + \mathbf{v}_{qk})||_2)}, \end{aligned} \tag{13}$$

where $\mathbf{v}_{qk}$ denotes the sum of all the vector representations of keywords in factor k for question q. The concentration parameters κ_{qk} are sampled from the following distribution:

$$P(\kappa_{qk} | \boldsymbol{\kappa}_{q\neg k}, \boldsymbol{m}_0, R_0, c) \propto \frac{(C_D(\kappa_{qk}))^{c+N_{qk}}}{C_D(\kappa_{qk} || R_0 \boldsymbol{m}_0 + \mathbf{v}_{qk} ||_2)}. \tag{14}$$

The conditional distribution of κ_{qk} is again not of a standard form, we use a step of Metropolis Hasting sampling (with log-normal proposal distribution) to sample κ_{qk}. To this point, we get the full expression of Eq. (11). In the circumstance when the model fitting efficiency becomes a concern, the sampling process specified by Eq. (11) can be approximated via the method specified in [30], which also produces satisfactory results.

Here, we make an approximation by removing the impact of y in terms of determining the value z. For the answer provided by user u for questions q, $y_{u,qk}$ is determined via:

$$y_{u,qk} = \begin{cases} 0 & \text{If } \nexists\, m \text{ satisfies } z_{qum} = k, \\ 1 & \textit{Otherwise}. \end{cases} \tag{15}$$

Finally, we move on to sample the truth label for each answer factor under each question t_{qk} via the following posterior distribution:

$$\begin{aligned} & P(t_{qk} = x \mid \boldsymbol{t}_{q,\neg k}, \boldsymbol{z}_q, \boldsymbol{y}_q, \alpha_{0,0}, \alpha_{0,1}, \alpha_{1,0}, \alpha_{1,1}, \alpha_0^{(a)}, \alpha_1^{(a)}) \\ & \propto \alpha_x^{(a)} \prod_{u \in U_q} \frac{\alpha_{x, y_{u,qk}} + n_{u,x,y_{u,qk}}}{\alpha_{x,0} + \alpha_{x,1} + n_{u,x,0} + n_{u,x,1}}, \end{aligned} \tag{16}$$

where U_q is the set of users that provide answer for question q. Here, $x \in \{0, 1\}$. $n_{u,0,0}$, $n_{u,0,1}$, $n_{u,1,0}$ and $n_{u,1,1}$ denote the number of true negative, false positive, false negative and true positive factors provided by user u, respectively.

User Reliability Estimation: With t, y, κ and z determined, we are able to obtain the closed-form solution for ϕ_u^0 and ϕ_u^1 by setting the partial derivatives of the negative log-likelihood respective to ϕ_u^0 and ϕ_u^1 to zero:

$$\phi_u^0 = \frac{\alpha_{0,1} + n_{u,0,1}}{\alpha_{0,0} + \alpha_{0,1} + n_{u,0,1} + n_{u,0,0}}, \tag{17}$$

$$\phi_u^1 = \frac{\alpha_{1,1} + n_{u,1,1}}{\alpha_{1,0} + \alpha_{1,1} + n_{u,1,0} + n_{u,1,1}}, \tag{18}$$

where $n_{u,0,0}$, $n_{u,0,1}$, $n_{u,1,0}$ and $n_{u,1,1}$ are user reliability statistics, which denote the number of true negative, false positive, false negative and true positive factors provided by user u, respectively. Moreover, these statistics also allow us to calculate other user reliability metrics, e.g., precision score of a user:

$$prec_u = \frac{\alpha_{1,1} + n_{u,1,1}}{\alpha_{0,1} + \alpha_{1,1} + n_{u,0,1} + n_{u,1,1}}. \tag{19}$$

This score is also used in the experiment section to validate the estimated user reliability.

5 EXPERIMENTS

In this section, we empirically validate the performance of the proposed method from the following aspects: Firstly, we compare the performance of the proposed method with the state-of-the-art truth discovery methods as well as a couple of retrieval based schemes to demonstrate the advantage of utilizing fine-grained semantic units of answers for better answer trustworthiness estimation. After that, we provide a case study to show that the results produced by the proposed method are highly interpretable. Finally, we validate the estimated user reliabilities with groundtruth to further prove that the proposed method can make a good estimation of user reliabilities.

5.1 Datasets

SuperUser Dataset & ServerFault Dataset: These two datasets are collected from the community question answering (CQA) websites *SuperUser.com* and *ServerFault.com*, respectively. These two websites are mainly focused on the questions about general daily computer usages and server administration, respectively. The task on these datasets is to extract the most trustworthy answer to each question. We use the answers' votes from *SuperUser.com* and *ServerFault.com* as the groundtruths for evaluation.

Student Exam Dataset [24]: This dataset is collected from introductory computer science assignments with answers provided by a class of undergraduate students in the University of North Texas. 30 students submit answers to these assignments. For each assignment, the students' answers are collected via an online learning environment. The task on this dataset is to extract Top-K (K is set to 1-10 in this paper) trustworthy student answers for each question. The groundtruth answers are given by the instructors. All the answers are independently graded by two human judges, using an integer scale from 0 (completely incorrect) to 5 (perfect answer). The statistics of these three datasets are shown in Table 1.

Table 1: Data Statistics.

Item	SuperUser	ServerFault	Student Exam
# of Questions	3379	7621	80
# of Users	1036	1920	30
# of Answers	16014	40373	2273

Pre-Processing: For all the datasets, we discard all code blocks, HTML tags, and stop words in the text. Answer keywords are extracted using entity dictionary and Stanford POS-Tagger[3]. To

[3] https://nlp.stanford.edu/software/tagger.shtml

train word vector representations, we utilize all the crawled texts as the corpus. Skip-gram architecture in package gensim[4] is used to learn the vector representation of every answer keyword. The dimensionality of word vectors is set to 100, context window size is set to 5, and the minimum occurrence count is set to 20. For more details on the embedding algorithm, please refer to [23].

5.2 Experiment Protocols

5.2.1 Comparison Methods. We compare the proposed TextTruth model against several state-of-the-art truth discovery and retrieval-based answer selection approaches.

Bag-of-Word (BOW) Similarity: The bag-of-word vectors of questions and their answers are extracted. Answers are ranked according to the similarity values between the question vector and its corresponding answer vectors.

Topic Similarity: We utilize Latent Dirichlet Allocation (i.e. LDA [2]) to extract a 100-dimension topic representation for each question and its corresponding answers. Similar to BOW, answers are ranked according to the cosine similarity to the question.

CRH [13] + Topic Dist.: CRH is an optimization based truth discovery framework which can handle both categorical and continuous data. The goal of the optimization problem is to minimize the weighted loss of the aggregation results. In the experiment, we use the topic distributions as the representations of the whole answers to be fed to CRH.

CRH [13] + Word Vec.: This baseline approach is similar to *CRH + Topic Dist.* except that the inputs are changed to the average word vectors of answers. These word vector representations are learned as in [23].

CATD [12] + Topic Dist.: CATD is another optimization based truth discovery framework which considers the long-tail phenomena in the data. The optimization objective is similar to that of CRH. However, the upper bounds of user reliability are used for weight loss calculation. Similar to *CRH + Topic Dist.*, we use the topic distributions as the representations of the whole answers to be fed to CATD.

CATD [12] + Word Vec.: This baseline approach is similar to *CATD + Topic Dist.* except that the inputs are changed to the average word vectors of answers. The word vector representations are the same as those in *CRH + Word Vector.*

For each baseline approach, we implement it and set its parameters according to the method recommended by the original papers.

5.2.2 Evaluation Metrics. Due to the differences in dataset characteristics, evaluation metrics for three datasets are slightly different. On CQA datasets, we report the precisions of returned best answers from each method for each question. On student test dataset, we report the average score of returned top-K (K is set to 1-10 in this paper) trustworthy answers from each method for each question.

5.3 Performance and Analysis

The results are shown in Figure 3 and Table 2. For student exam dataset, we only show the results on exam 1 3 data. The results on rest exams follow the same tendency. As one can see, the proposed method TextTruth consistently outperforms all the baseline methods. By outperforming various retrieval-based approaches and state-of-the-art truth discover approaches, the proposed TextTruth demonstrates its great advantages on natural language data.

Table 2: Results on ServerFault Dataset & SuperUser Dataset.

Method	ServerFault	SuperUser
BOW Similarity	0.2077	0.1944
Topic Similarity	0.2462	0.2462
CATD + Topic Dist.	0.2311	0.2308
CATD + Word Vec.	0.1821	0.2234
CRH + Topic Dist.	0.2453	0.2453
CRH + Word Vec.	0.1847	0.2231
TextTruth	**0.3985**	**0.4019**

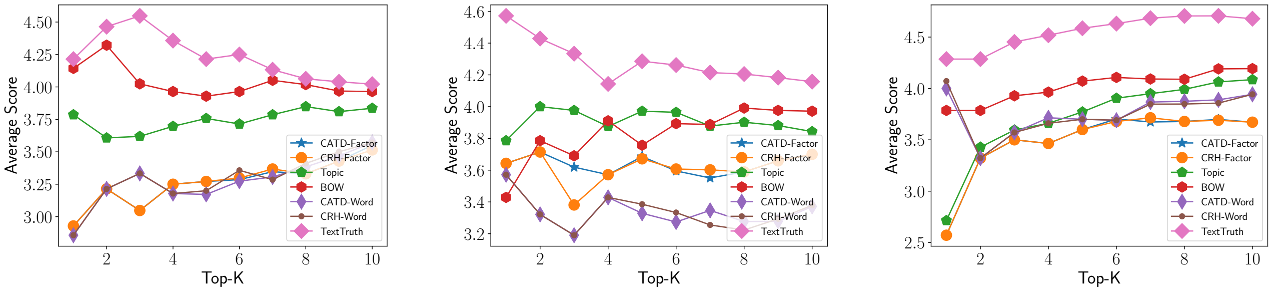
The reasons why the proposed TextTruth surpasses all the baseline methods are as follows. First, retrieval-based approaches (i.e., *BOW Similarity* and *Topic Similarity*) rank the answers merely based on the semantic similarity between the question and answers. However, a question itself does not necessarily cover all the semantics that should be covered in ideal answers. Therefore, retrieval-based methods only discover relevant answers instead of trustworthy answers. On the other hand, although existing truth discovery methods can capture user reliability for answer ranking, the performance is not very satisfactory. This is because these truth discovery approaches treat the answers as an integrated semantic unit, and ignore the fact that the semantic meaning of each answer may be complicated. Therefore, single vector representations fail to capture the innate correlations among these answers. To make things worse, CRH and CATD regard the weighted aggregation of these single vector representations as the “true” semantic representation to evaluate user reliabilities. However, answers from different users may involve distinct aspects of answers. Therefore, aggregating semantic representation of answers with distinct aspects only produces an inaccurate representation, which cannot be used to correctly estimate the reliabilities of users. The inaccurate user reliability estimation would further lead to incorrect aggregated results.

In contrast to existing approaches, the proposed TextTruth regards each answer as a collection of fine-grained semantic units (i.e., factors), which are represented by separated keyword vector representations. Based on these semantic units, TextTruth discovers the innate factors of each answer by grouping keywords into factors, and evaluates the trustworthiness of each answer on the top of these factors. As mentioned in the above paragraph, the major reason why existing truth discovery methods cannot produce satisfactory results is that these methods cannot aggregate the semantic representation of answers with distinct aspects effectively. Instead, the proposed TextTruth evaluate the users' reliabilities according to whether their answers contain keywords from the factors that are regarded to be correct (or incorrect). Therefore, the trustworthiness of each answer is better evaluated, which leads to the best result.

5.4 Case Study

To better evidence the analysis above, we give a case study on a question in the exam dataset. The question is related to the data structure. The result of the case study is shown in Table 3. In Table 3

[4] https://pypi.python.org/pypi/gensim, an implementation of Word2Vec



(a) Exam 1 **(b) Exam 2** **(c) Exam 3**

Figure 3: Performance on Exam Datasets.

Table 3: Case Study of Real Question and Answers.

	Content
Question	What is a tree?
Groundtruth Answer	A collection of nodes, which has a special node called root, and the rest of the nodes are partitioned into one or more disjoint sets, each set being a tree.
Top Answer 1	A tree is a finite set of one or more nodes with a specially designated node called the root and the remaining nodes are partitioned into disjoint sets where each of these sets is a tree.
Top Answer 2	A a finite collection of nodes, where it starts, with an element, called the root„ which has children, and its children have children until you get to the leaves which are the last elements and have to children
Untrustworthy Answer	It is a list of numbers in a list made by comparing values of nodes already in the tree and adding to the appropriate spot. Its a list made up of nodes with left and right points.

words in blue color are keywords that are estimated to be trustworthy, while words in red color are keywords that are estimated to be untrustworthy or unrelated. The groundtruth answer is provided by the instructors.

As one can see, the proposed method can automatically select keywords that are meaningful to the questions, such as “node”, “tree” and “root”. Moreover, we can observe that the top-ranked answers have more true keywords than low-ranked untrustworthy answers. These phenomena again demonstrate that the results of the proposed model are both effective and interpretable. The case study also demonstrates why existing approaches fail to produce satisfactory results. First, the question itself merely consists of one keyword ‘tree’. Therefore, retrieval-based methods, rank ‘*Untrustworthy Answer*’ over ‘*Top Answer 2*’, because it contains exactly the same keyword that exists in the question. This indicates that we cannot rely merely on relevance to find trustworthy answers. Second, we can see that the correct keywords involve multiple aspects (i.e., factors). These factors shape a comprehensive description of a tree. Such phenomenon is very common in natural language questions and answers, but cannot be successfully handled by the existing methods. That is why the proposed method can produce better results than the state-of-the-art truth discovery methods.

5.5 User Reliability Validation

The quantitative results and the case study shown above have demonstrated that the proposed method can outperform other baseline methods. In this section, we *further* exhibit the estimated user reliabilities by the proposed approach. As there are no direct user reliability values on the CQA dataset, we only investigate the estimated user reliabilities on the student exam dataset. Specifically, we use the average score of a student’s answer to each question as the groundtruth reliability. Intuitively, the learned two-fold user reliability parameters (i.e., ϕ^0 and ϕ^1) are not directly proportional to the true user reliability; we use the metric *prec* defined in Eq. (19) for user reliability validation. Due to space limitation, we only show one example, which comes from the mid-result of TextTruth on exam 10, in Figure 4. In Figure 4, each point denotes a user. The Y-axis is the user reliability groundtruth and the X-axis is the estimated user precision score. As one can see, the estimated user reliability score (X) typically increases when the groundtruth user

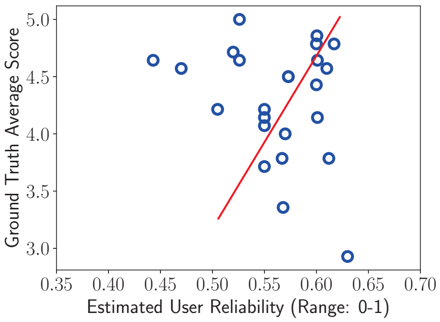


Figure 4: Estimated User Reliability V.S. Ground Truth User Score.

reliability (Y) increases which means that the proposed TextTruth successfully captures the reliabilities of users.

6 CONCLUSIONS

As an emerging topic, truth discovery has shown its effectiveness in a wide range of applications with structured data. However, existing methods all suffer on unstructured text data, due to the semantic ambiguity of natural languages and the complexity of text answers. To tackle these challenges, in this paper, we propose a probabilistic model named TextTruth that takes vector representations of key factors extracted from answers as inputs and outputs the ranking of answers based on the trustworthiness of key factors within each answer. Specifically, the model jointly learns the clustering label and truth label for each answer factor cluster through modeling the generative process of answer factors' embedding representations. Experimental results on three real-world datasets prove the effectiveness of the proposed TextTruth model. Furthermore, case studies illustrate that the learned labels are interpretable.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous referees for their valuable comments and suggestions. This work was sponsored in part by US National Science Foundation under grants IIS-1319973, IIS-1553411, CNS-1652503 and CNS-1737590. The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing any funding agency.

REFERENCES

[1] Kayhan Batmanghelich, Ardavan Saeedi, Karthik Narasimhan, and Sam Gershman. 2016. Nonparametric spherical topic modeling with word embeddings. *arXiv preprint arXiv:1604.00126* (2016).

[2] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *JMLR* 3, Jan (2003), 993–1022.

[3] Mohamed Bouguessa, Benoît Dumoulin, and Shengrui Wang. 2008. Identifying authoritative actors in question-answering forums: the case of yahoo! answers. In *Proc. of SIGKDD*. 866–874.

[4] Alexander Philip Dawid and Allan M Skene. 1979. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Applied statistics* (1979), 20–28.

[5] Xin Luna Dong, Laure Berti-Equille, and Divesh Srivastava. 2009. Integrating conflicting data: the role of source dependence. *PVLDB* 2, 1 (2009), 550–561.

[6] Xin Luna Dong, Evgeniy Gabrilovich, Kevin Murphy, Van Dang, Wilko Horn, Camillo Lugaresi, Shaohua Sun, and Wei Zhang. 2015. Knowledge-based trust: Estimating the trustworthiness of web sources. *PVLDB* 8, 9 (2015), 938–949.

[7] Minwei Feng, Bing Xiang, Michael R Glass, Lidan Wang, and Bowen Zhou. 2015. Applying deep learning to answer selection: A study and an open task. In *Proc. of IEEE Workshop on ASRU*. 813–820.

[8] Alban Galland, Serge Abiteboul, Amélie Marian, and Pierre Senellart. 2010. Corroborating information from disagreeing views. In *Proc. of WSDM*. ACM, 131–140.

[9] Siddharth Gopal and Yiming Yang. 2014. Von Mises-Fisher Clustering Models.. In *Proc. of ICML*.

[10] Liangjie Hong and Brian D Davison. 2009. A classification-based approach to question answering in discussion boards. In *Proc. of SIGIR*. 171–178.

[11] Dan Jurafsky and James H. Martin. 2017. *Speech and language processing : an introduction to natural language processing, computational linguistics, and speech recognition (3rd Edition)*.

[12] Qi Li, Yaliang Li, Jing Gao, Lu Su, Bo Zhao, Murat Demirbas, Wei Fan, and Jiawei Han. 2014. A confidence-aware approach for truth discovery on long-tail data. *PVLDB* 8, 4 (2014), 425–436.

[13] Qi Li, Yaliang Li, Jing Gao, Bo Zhao, Wei Fan, and Jiawei Han. 2014. Resolving conflicts in heterogeneous data by truth discovery and source reliability estimation. In *Proc. of SIGMOD*. 1187–1198.

[14] Xian Li, Xin Luna Dong, Kenneth Lyons, Weiyi Meng, and Divesh Srivastava. 2012. Truth finding on the deep web: is the problem solved? *PVLDB* 6, 2 (2012), 97–108.

[15] Xian Li, Xin Luna Dong, Kenneth B Lyons, Weiyi Meng, and Divesh Srivastava. 2015. Scaling up copy detection. In *Proc. of ICDE*. 89–100.

[16] Xian Li, Weiyi Meng, and T Yu Clement. 2016. Verification of Fact Statements with Multiple Truthful Alternatives.. In *WEBIST (2)*. 87–97.

[17] Xian Li, Weiyi Meng, and Clement Yu. 2011. T-verifier: Verifying truthfulness of fact statements. In *Data Engineering (ICDE), 2011 IEEE 27th International Conference on*. 63–74.

[18] Yaliang Li, Nan Du, Chaochun Liu, Yusheng Xie, Wei Fan, Qi Li, Jing Gao, and Huan Sun. 2017. Reliable Medical Diagnosis from Crowdsourcing: Discover Trustworthy Answers from Non-Experts. In *Proc. of WSDM*. 253–261.

[19] Yaliang Li, Qi Li, Jing Gao, Lu Su, Bo Zhao, Wei Fan, and Jiawei Han. 2015. On the discovery of evolving truth. In *Proc. of SIGKDD*. 675–684.

[20] Fenglong Ma, Yaliang Li, Qi Li, Minghui Qiu, Jing Gao, Shi Zhi, Lu Su, Bo Zhao, Heng Ji, and Jiawei Han. 2015. Faitcrowd: Fine grained truth discovery for crowdsourced data aggregation. In *Proc. of SIGKDD*. 745–754.

[21] Fenglong Ma, Chuishi Meng, Houping Xiao, Qi Li, Jing Gao, Lu Su, and Aidong Zhang. 2017. Unsupervised Discovery of Drug Side-effects from Heterogeneous Data Sources. In *Proc. of SIGKDD*. ACM, 967–976.

[22] Chuishi Meng, Wenjun Jiang, Yaliang Li, Jing Gao, Lu Su, Hu Ding, and Yun Cheng. 2015. Truth Discovery on Crowd Sensing of Correlated Entities. In *Proc. of SenSys*.

[23] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in NIPS*. 3111–3119.

[24] Michael Mohler, Razvan Bunescu, and Rada Mihalcea. 2011. Learning to grade short answer questions using semantic similarity measures and dependency graph alignments. In *Proc. of ACL*. 752–762.

[25] Gabriel Nuñez-Antonio and Eduardo Gutiérrez-Peña. 2005. A Bayesian analysis of directional data using the projected normal distribution. *Journal of Applied Statistics* 32, 10 (2005), 995–1001.

[26] Jeff Pasternack and Dan Roth. 2011. Making better informed trust decisions with generalized fact-finding. In *IJCAI*. 2324–2329.

[27] Jeff Pasternack and Dan Roth. 2013. Latent credibility analysis. In *Proc. of WWW*. 1009–1020.

[28] Cicero dos Santos, Ming Tan, Bing Xiang, and Bowen Zhou. 2016. Attentive pooling networks. *arXiv preprint arXiv:1602.03609* (2016).

[29] Anish Das Sarma, Xin Luna Dong, and Alon Halevy. 2011. Data integration with dependent sources. In *Proc. of EDBT*.

[30] J. Straub, T. Campbell, J. P. How, and J. W. Fisher. 2015. Small-variance nonparametric clustering on the hypersphere. In *Proc. of CVPR*. 334–342.

[31] Ming Tan, Bing Xiang, and Bowen Zhou. 2015. LSTM-based Deep Learning Models for non-factoid answer selection. *arXiv preprint arXiv:1511.04108* (2015).

[32] Dong Wang, Md Tanvir Amin, Shen Li, Tarek Abdelzaher, Lance Kaplan, Siyu Gu, Chenji Pan, Hengchang Liu, Charu C Aggarwal, Raghu Ganti, et al. 2014. Using humans as sensors: an estimation-theoretic perspective. In *Proc. of IPSN*. 35–46.

[33] Dong Wang, Lance Kaplan, Hieu Le, and Tarek Abdelzaher. 2012. On truth discovery in social sensing: A maximum likelihood estimation approach. In *Proc. of IPSN*. 233–244.

[34] Di Wang and Eric Nyberg. 2015. A Long Short-Term Memory Model for Answer Sentence Selection in Question Answering.. In *ACL (2)*. 707–712.

[35] Xianzhi Wang, Quan Z Sheng, Xiu Susie Fang, Lina Yao, Xiaofei Xu, and Xue Li. 2015. An integrated bayesian approach for effective multi-truth discovery. In *Proc. of CIKM*. 493–502.

[36] Yaqing Wang, Fenglong Ma, Lu Su, and Jing Gao. 2017. Discovering Truths from Distributed Data. In *Proc. of ICDM*. IEEE, 505–514.

[37] Zhiguo Wang, Haitao Mi, and Abraham Ittycheriah. 2016. Sentence Similarity Learning by Lexical Decomposition and Composition. *arXiv preprint arXiv:1602.07019* (2016).

[38] Jacob Whitehill, Ting-fan Wu, Jacob Bergsma, Javier R Movellan, and Paul L Ruvolo. 2009. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *Advances in NIPS*. 2035–2043.

[39] Houping Xiao, Jing Gao, Qi Li, Fenglong Ma, Lu Su, Yunlong Feng, and Aidong Zhang. 2016. Towards confidence in the truth: A bootstrapping based truth discovery approach. In *Proc. of SIGKDD*. 1935–1944.

[40] Liu Yang, Minghui Qiu, Swapna Gottipati, Feida Zhu, Jing Jiang, Huiping Sun, and Zhong Chen. 2013. Cqarank: jointly model topics and expertise in community question answering. In *Proc. of CIKM*. 99–108.

[41] Xuchen Yao, Benjamin Van Durme, and Peter Clark. 2013. Answer Extraction as Sequence Tagging with Tree Edit Distance. In *Proc. of NAACL-HLT*. 858–867.

[42] Xiaoxin Yin, Jiawei Han, and Philip S Yu. 2008. Truth discovery with multiple conflicting information providers on the web. *TKDE* 20, 6 (2008), 796–808.

[43] Hengtong Zhang, Qi Li, Fenglong Ma, Houping Xiao, Yaliang Li, Jing Gao, and Lu Su. 2016. Influence-Aware Truth Discovery. In *Proc. of CIKM*. 851–860.

[44] Bo Zhao, Benjamin IP Rubinstein, Jim Gemmell, and Jiawei Han. 2012. A bayesian approach to discovering truth from conflicting sources for data integration. *PVLDB* 5, 6 (2012), 550–561.

[45] Guangyou Zhou, Siwei Lai, Kang Liu, and Jun Zhao. 2012. Topic-sensitive probabilistic model for expert finding in question answer communities. In *Proc. of CIKM*. 1662–1666.