Effort-driven Fact Checking

TAM NGUYEN

The Web constitutes a valuable source of information. In recent years, it fostered the construction of large-scale knowledge bases, such as Freebase, YAGO, and DBpedia, each storing millions of facts about society in general, and specific domains, such as politics or medicine. The open nature of the Web, with content potentially being generated by everyone, however, leads to inaccuracies and misinformation, such as fake news and exaggerated claims. Construction and maintenance of a knowledge base thus relies on fact checking, assessing the credibility of facts. Due to the inherent lack of ground truth information, fact checking cannot be done in a purely automated manner, but requires human involvement. In this paper, we propose a framework to guide users in the validation of facts, striving for a minimisation of the invested effort. Specifically, we present a probabilistic model to identify the facts for which manual validation is most beneficial. As a consequence, our approach yields a high quality knowledge base, even if only a sample of a collection of facts is validated. Our experiments with three large-scale datasets demonstrate the efficiency and effectiveness of our approach, reaching levels of above 90% precision of the knowledge base with only a third of the validation effort required by baseline techniques.

1 INTRODUCTION

Extracting factual knowledge from Web data plays an important role in various applications. For example, knowledge bases such as Freebase [fre 2017], YAGO [yag 2017] and DBpedia [dbp 2017] rely on Wikipedia to extract entities and their relations. These knowledge bases store millions of facts, about society in general as well as specific domains such as politics and medicine. Facts can be stored in various formats, reaching from unstructured text segments, through formal concepts and relations, to structured statements. In any case, extraction of factual knowledge first yields candidate facts (aka claims), for which the credibility needs to be assessed. Given the open nature of the Web, where content is potentially generated by everyone, fact extraction faces inaccuracies and misinformation, such as fake news and exaggerated claims. Hence, building a knowledge base from multiple Web sources does not only require conflict resolution and data cleansing [Dong et al. 2012], but also calls for methods to ensure the credibility of the extracted facts, especially in domains dealing with sensitive information, such as healthcare [Mukherjee et al. 2014].

To assess the credibility of facts extracted from the Web, automated methods rely on classification [Lehmann et al. 2012] or sensitivity analysis [Wu et al. 2014]. While these methods scale to the volume of Web data, they are hampered by the inherent ambiguity of natural language, deliberate deception, and domain-specific semantics. Consider the claims of ‘the world population being 7.5 billion’ or ‘antibiotics killing bacteria’. Both represent common-sense facts. Yet, these facts have been generated by complex statistical and survey methods and, therefore, cannot easily be inferred from other basic facts.

For applications that rely on accurate facts, incorporating manual feedback is the only way to overcome the limitations of automated fact checking. However, eliciting user input is challenging due to several reasons. First, user input is expensive (in terms of time, cost, etc.), so that a validation of all claims is infeasible, even if one relies on a large number of users (e.g., by crowdsourcing [Hung et al. 2013]) and ignores the overhead to resolve disagreement among them. Second, claims are not independent, but connected in a network of Web sources. An assessment of their credibility therefore requires effective propagation of user input between correlated claims. Finally, there is a trade-off between the precision of a knowledge base (the ratio of credible facts) and the amount of user input: The more facts are checked manually, the higher the precision. However, user input is limited by some effort budget in most applications.

Against this background, we aim at supporting a user in the validation of facts by means of a pay-as-you-go approach. While user input is incorporated continuously to improve the results of automatic fact checking, our ultimate goal is to instantiate an accurate knowledge base, even if not all claims have been validated. By (i) inferring the credibility of non-validated facts from those that have been validated, and by (ii) guiding a user in the validation process, we reduce the amount of manual effort needed to achieve a specific level of result precision. Both steps, credibility inference and user guidance, are interrelated. On the one hand, inference exploits mutual reinforcing relations between Web sources and claims, which are further justified based on the user input. On the other hand, a user is guided based on the potential effect of the validation of a claim for credibility inference.

Our contributions are summarised as follows:

Author’s address: Tam Nguyen.

2019. Manuscript submitted to ACM

- *Fact checking model:* §2 proposes a model for fact checking that combines automated credibility inference with input from a user who manually validates facts. We also formulate the problem of effort minimisation in fact checking and introduce an iterative approach to guide a user in the validation process.
- *Probabilistic credibility inference:* §3 presents methods to construct a probabilistic knowledge base, to perform incremental inference based on user input, and to construct a trusted set of facts.
- *Probabilistic user guidance:* §4 develops a strategy to choose the claims for which manual validation is most beneficial, dropping traditional assumptions about the trustworthiness of Web sources. We also combine credibility inference and user guidance in a comprehensive validation process.

Furthermore, §5 evaluates our techniques with three large-scale datasets, demonstrating their efficiency and effectiveness. Finally, §6 discusses related work before §7 concludes the paper.

2 MODEL AND APPROACH

Setting. We model the setting of fact checking by a set of data sources $S = \{s_1, \ldots, s_u\}$, a set of documents $D = \{d_1, \ldots, d_m\}$, and a set of candidate facts, or short claims, $C = \{c_1, \ldots, c_n\}$. A source could be a user, a website, a news provider, or a business entity. A document is often textual and provided by some source (e.g., a tweet, a news item, or a forum posting). Sources, documents, and claims jointly represent a fact database, denoted by a tuple $N = \langle S, D, C \rangle$. To construct an accurate knowledge base, the goal is to infer labels for every claim in C, where $c_i = 1$ denotes that a fact c_i is credible, whereas $c_i = 0$ represents the opposite.

User input is modelled by a fact checking function $e : C \rightarrow \{0, 1, \ominus\}$. The label $\ominus$ denotes that a claim has not yet been validated. Our work employs a probabilistic model, where $P(c = 1)$ denotes the probability that claim c is credible. Then, a probabilistic fact database is a tuple $Q = \langle N, e, P \rangle$, with e being a fact checking function and P assigning a credibility probability to all claims.

The goal of fact checking is a deterministic assignment, a grounding function $g : C \rightarrow \{0, 1\}$ that assigns truth values to all claims. It then enables the construction of an accurate knowledge base from the claims that are deemed credible.

Effort minimisation. While our objective is to instantiate a trusted set of credible facts, our work combines automated fact checking with input from users validating claims. However, such validation is costly, in terms of user hiring cost and time. Therefore, user input is commonly limited by an effort budget, which leads to a trade-off between validation accuracy and invested effort.

Going beyond this trade-off, we aim at minimising the user effort invested to reach a given validation goal. We consider fact checking as an iterative process with a user validating the credibility of a claim in each iteration. This process halts either when reaching a validation goal or upon consumption of the available effort budget. The former relates to the desired result quality, e.g., a threshold on the estimated credibility of the deterministic assignment. The latter defines an upper bound for the number of validations by a user and, thus, iterations of the validation process.

Formally, for a fact database $N = \langle S, D, C \rangle$, conducting fact checking leads to a sequence of deterministic assignments $\langle g_0, g_1, \ldots, g_n \rangle$, termed a *validation sequence*. Each g_i represents the assignment obtained after the i-th iteration of the validation process. Given an effort budget b and a validation goal Δ, we refer to a sequence $\langle g_0, g_1, \ldots, g_n \rangle$ as being *valid*, if $n \leq b$ and d_n satisfies Δ. Let $\mathcal{R}(\Delta, b)$ denote a finite set of valid validation sequences that can be created by instantiations of the validation process. Then, a validation sequence $\langle g_0, g_1, \ldots, g_n \rangle \in \mathcal{R}(\Delta, b)$ is *minimal*, if for any validation sequence $\langle g'_0, g'_1, \ldots, g'_m \rangle \in \mathcal{R}(\Delta, b)$ it holds that $n \leq m$.

Problem 1 (Effort Minimisation in Fact Checking).
Let $\langle S, D, C \rangle$ be a fact database and $\mathcal{R}(\Delta, b)$ a set of valid validation sequences for an effort budget b and a goal Δ. The problem of effort minimisation in fact checking *is the identification of a minimal sequence $\langle g_0, g_1, \ldots, g_n \rangle \in \mathcal{R}(\Delta, b)$.*

As detailed above, a validation goal is commonly defined in terms of a threshold on the estimated credibility of the deterministic assignment. However, solving Problem 1 is challenging. Claims are not independent, but subject to mutual reinforcing relations with Web sources and documents. Consequently, the validation of one claim may affect the probabilistic credibility assessment of other facts. Furthermore, sources that try to spread misinformation influence fact checking and may distort the evaluation of a fact's credibility. Finally, there is a computational challenge and finding an optimal solution to Problem 1 quickly becomes intractable: all permutations of all subsets (of size $\leq b$) of claims would have to be explored.

Guided fact checking. To address the problem of effort minimisation in fact checking, we consider a process that guides a user in the validation of claims. The general idea of the validation process is summarised as follows: User input shall be sought solely on the 'most promising' unverified facts, i.e., those for which manual validation is expected to have the largest impact on the estimated credibility of the resulting deterministic assignment.

Let $Q = \langle N, e, P \rangle$, with $N = \langle S, D, C \rangle$, be a probabilistic fact database. Then, the validation process continuously updates a deterministic assignment g of truth values to claims by:
(1) *selecting* a claim c for which feedback shall be sought;
(2) *eliciting* user input on the credibility of c, which is represented by the fact checking function, $e(c)$;
(3) *inferring* the implications of the user input on the probabilistic fact database Q;
(4) *deciding* on the deterministic assignment g that captures the facts that are assumed to be credible.
To realise the above process, steps (1), (3), and (4) need to be instantiated with specific methods. An example for a straight-forward instantiation would be a validation process that *selects* a claim randomly; limits the *inference* to the claim for which feedback has been sought; and *decides* that a claim c is credible, $g(c) = 1$, if and only if it holds $P(c) \geq 0.5$. In the remainder of this paper, we present methods for a more elaborated instantiation of the above process that exploit the mutual reinforcing relations between Web sources and claims to *infer* the implications of user input and *decide* on the deterministic assignment. Furthermore, we also show how to *select* the claims for which manual validation is most beneficial.

3 PROBABILISTIC FACT CHECKING

This section presents a probabilistic model for fact checking (§3.1). Based thereon, we introduce mechanisms for incremental inference (§3.2) and the instantiation of a deterministic assignment (§3.3).

3.1 A Probabilistic Model for Fact Checking

Sources of uncertainty. A probabilistic fact database is constructed based on sources and documents, each encoded using a set of features, such as the frequency of updates at a source or linguistic characteristics of a document. Our model abstracts from the specific nature of these features, but takes into account that the trustworthiness of a source and the language quality of a document often have a strong influence on the credibility of the extracted claims. Each source s_k is associated with a feature vector $\langle f_1^S(s_k), \ldots, f_{m_S}^S(s_k) \rangle$ of m_S source features. In the same vein, $\langle f_1^d(d_j), \ldots, f_{m_D}^d(d_j) \rangle$ is a vector of m_D document features assigned to each document d_j.

Features of sources and documents interact with each other, as well as the credibility of the extracted claims, so that we consider the following relations between them:

- *Causal relation:* Claims are provided in a document by a source. Thus, a claim's credibility depends on both trustworthiness of the source and the language quality of the document. A claim is more likely to be credible, if it is posted by a trustworthy source using confident and objective language. Yet, the intentions of a source, and thus its trustworthiness, may change over different contexts and hence documents.
- *Mutual relation:* Causal relations are complemented by mutual relations that stem from overlapping sets of a sources, documents, and claims. Multiple documents, potentially from multiple sources, may contain the same claim. For example, if a source disagrees with a claim that is accepted by several sources, it shall be regarded as not trustworthy.

To model these relations along with observed variables, we rely on a Conditional Random Field (CRF) model, see Fig. 1. Specifically, causal relations are captured by cliques in the CRF, while mutual relations are modelled by the overlap of cliques, which leads to the factorization of cliques to compute probability distributions.

The Conditional Random Field model. We construct a CRF as an undirected graph of random variables S, D, C for sources, documents, and claims. There are three types of edges between vertices, those between a source and a document, between a document and a claim, and between a source and a claim. Hence, the CRF contains cliques of a source, a document, and a claim. Since the same claim can be provided in different documents, by the same or different sources, any random variable can be part in multiple cliques.

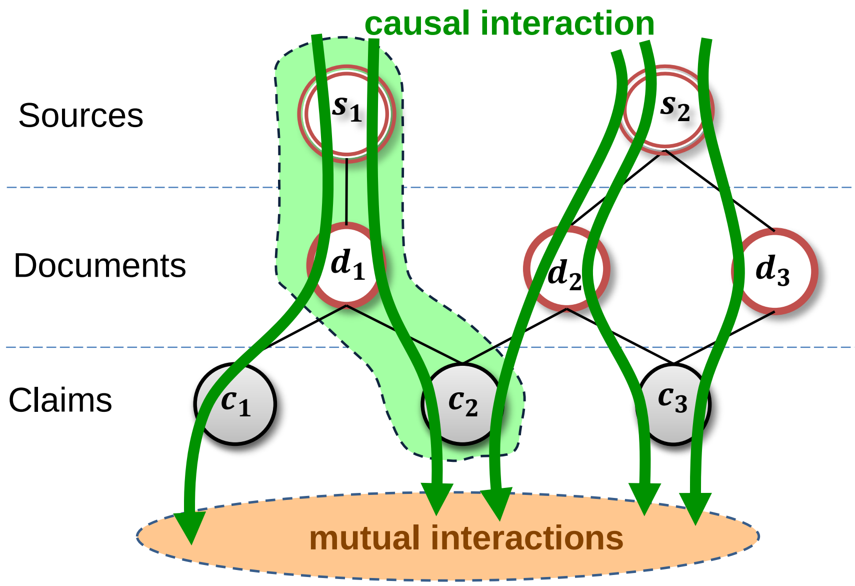

Fig. 1. A fraction of probabilistic fact database

We note that S and P are observed variables, while C represents output variables. Therefore, the model likelihood can be expressed in the form of a conditional distribution:

$$Pr(C|D,S;W) = \frac{1}{Z(D,S;W)} \prod_{\pi=\{c_i,d_j,s_k\}\in\Pi} \phi(c_i,d_j,s_k;W_\pi) \tag{1}$$

where Π is the set of all possible cliques; c_i, d_j, s_k are a claim, document, and source of a clique π, respectively; $Z(D,S;W) = \sum_C \prod_\pi \phi(c_i,d_k,s_k;W_\pi)$ is a normalization constant to ensure that the sum of probabilities over all configurations of C is equal to one; and $W = \bigcup_{\pi\in\Pi} W_\pi$ is the set of model parameters, used to control the effects of individual features. Using this model, we would like to compute the conditional distribution of C, given the source features and document features. This is realised by the log-linear model (aka logistic regression) that expresses the log of a potential function as a linear combination of associated features:

$$\log\ \phi(c_i,d_j,s_k;W_\pi) = \sum_{b=0}^{1} w_{\pi,b} \times I_b(c_i) + \sum_{t=1}^{m_D} w^D_{\pi,t} \times f^D_t(d_j) + \sum_{t=1}^{m_S} w^S_{\pi,t} \times f^S_t(s_k) \tag{2}$$

where $I_b(c_i)$ is the indicator function of the current configuration of c_i, i.e., $I_b(c_i)$ equals to one, if $b = c_i$, and is zero otherwise. Hence, we have different weights for each configuration of C and $W_\pi = \{w_{\pi,b}, w^D_{\pi,t}, w^S_{\pi,t}\}$ is the set of all weights. The above formalisation provides a simple model, in which the features of sources and documents are discriminative indicators for the credibility of the related claims. The weights of different features further enable us to tune their relative importance, as features vary between applications and shall be learned from labelled data.

Handling opposing claims. Different Web documents may contain the same claim with opposite stances—support or refute it [Hasan and Ng 2014]—and a source is considered trustworthy, if it refutes an incorrect claim. A model that only captures that a claim was mentioned in a document by a source would neglect this aspect. However, incorporating such information explicitly would over-complicate the model, as the number of Web documents is commonly much larger than the number of claims (see our real-world datasets in §5). We therefore tackle this challenge by introducing an opposing variable $\neg c$ for each claim c. Then, model complexity increases only slightly: C includes opposing claims, W contains a doubled number of parameters, and any document connects only to the positive or negative variable of a claim. However, as c and $\neg c$ cannot have the same truth value, we enforce a non-equality constraint, as follows:

$$Pr(c, \neg c') = \begin{cases} 0 & \text{if } c = c' \\ Pr(c, \neg c' | D, S; W) & \text{otherwise.} \end{cases} \quad (3)$$

3.2 Incremental Inference with User Input

Using the above formalisation, we need to solve the following optimisation problem to infer model parameters:

$$W^* = \arg\max_W \log Pr(C^L | D, S; W)$$
$$= \arg\max_W \log \sum_{C^U} Pr(C^L, C^U | D, S; W) \quad (4)$$

where C^L and C^U are sets of labelled and unlabelled claims, respectively. The log-likelihood optimisation is convex, since the logarithm is a monotonically increasing function and the probability distribution is in exponential form. However, the problem becomes intractable due to the exponential number of configurations of C^U. Moreover, upon receiving new user input, C^L and C^U change, so that re-computation is needed.

Against this background, we propose an incremental inference algorithm, called *iCRF*, which follows the view maintenance principle [Blakeley et al. 1986]. Estimation of assignment correctness and model parameters exploits the results of the previous iteration of the validation process, thereby avoiding re-computation. As we will show experimentally, this does not only increase inference efficiency, but also yields a better approximation compared to random probability estimation.

The *iCRF* algorithm implements the *conclude* function of the validation process introduced in §2. In the z-th iteration of the process, the input is given by the fact database N and the fact checking function e_z, which has been updated with the user input received in the z-th iteration. When conducting the validation process, we always maintain the state of the previous iteration as follows: If $\hat{c}$ is the claim validated in the z-th iteration of the validation process, we rely on the probabilistic fact database $Q_{z-1} = \langle N, e_{z-1}, P_{z-1} \rangle$ to conduct the reasoning. Then, the set of unlabelled and labelled claims is updated, $C^U_z = C^U_{z-1} \setminus \{\hat{c}\}$ and $C^L_z = C^L_{z-1} \cup \{\hat{c}\}$, and the algorithms returns a new probabilistic fact database $Q_z = \langle N, e_z, P_z \rangle$.

In each iteration of the validation process, the *iCRF* algorithm adopts the Expectation-Maximization (EM) principle [Mc-Callum et al. 2005] for inference. The choice for EM is motivated by its generally fast convergence, computationally efficiency, and particular usefulness when the likelihood is an exponential function (i.e., maximising log-likelihood becomes maximising a linear function). Specifically, we infer the labels of the variables C^U for unlabelled claims and learn the weight parameters W mutually. By relying on an EM-based approach, we can further naturally integrate user input on the credibility of specific claims. This is a major advantage compared to approaches based on gradient-descent [Mukherjee and Weikum 2015] that optimise model parameters, but do not enable the integration of user input and constraints (e.g., on opposing claims as introduced above).

Technically, inference is iterative and alternates between an *Expectation step* (E-step) and a *Maximization step* (M-step), until convergence. EM-based inference is conducted in each iteration of the validation process, while each EM iteration updates the model parameters W. Therefore, in the z-th iteration of this process, we obtain a sequence $W^0_z, W^1_z, \ldots, W^t_z$ of model parameters and a sequence $P^0_z, P^1_z, \ldots, P^t_z$ of credibility probabilities assigned to claims.

E-step: We estimate the credibility probabilities from the current parameter values. The first E-step of the z-th iteration of the validation process is based on parameters W^0_z, given as input from the previous iteration of the validation process, i.e., $W^0_z = W^{t_{z-1}}_{z-1}$, with t_{z-1} as the number of EM iterations in the $z-1$-th iteration of the validation process. In the t-th E-step of the z-th step of the validation process, credibility probabilities are computed as follows:

(1) Obtain a sequence of samples Ω^t by performing Gibbs sampling according to the conditional probability distribution:

$$q^t_z(C^U) = Pr(C^U | C^L, D, S; W^t_z)$$
$$\propto \prod_{\pi = \{c_i, d_j, s_k\} \in \Pi} Pr^{t-1}_z(c) \times \phi(c_i, d_j, s_k; W^t_z) \quad (5)$$

We incorporate non-equality constraints (Eq. 3) into Gibbs sampling using an idea similar to [Schmidt 2009], which, based on matrix factorisation, embeds constraints as factorised functions into the Markov chain Monte Carlo process.

Note that Ω^t is a sequence, as any configuration of C can appear multiple times. We weight the influence of causal interactions (i.e., cliques) by the credibility of their contained claims, so that user input is propagated via mutual interactions (i.e., overlaps) between the cliques.

(2) Compute the probability from the Gibbs samples, for each claim c for which no user input has been received ($e_z(c) = \ominus$):

$$Pr_z^t(c) = \frac{\sum_{C\in\Omega^t} \mathbb{1}_{c=1}}{|\Omega^t|} \tag{6}$$

For all other claims ($e_z(c) \neq \ominus$), the probability is fixed:

$$Pr_z^t(c) = \begin{cases} 1 & \text{if } e_z(c) = 1 \\ 0 & \text{otherwise.} \end{cases} \tag{7}$$

M-step: We compute the new parameter values by maximizing the expectation of log-likelihoods as a weighted average of the probability distribution of current label estimates. That is, in the t-th M-step of the z-th step of the validation process, we have:

$$W_z^{t+1} = \arg\max_{W'} \sum_{C^U} q_z^t(C^U) \log Pr(C^L, C^U | D, S; W') \tag{8}$$

This step is implemented by a L2-regularized Trust Region Newton Method [Lin et al. 2008], suited for large-scale data, where critical information is often sparse (many feature values may be zero).

The process converges when the difference between two consecutive estimates of parameters is insignificant.

3.3 Instantiation of Deterministic Assignment

After incorporating the user input of the z-th iteration of the validation process, a deterministic assignment may be instantiated by predicting the truth values of all claims. Since claims are not independent, we take the truth configuration with maximal joint probability:

$$g_z(C^U) = \arg\max_{C^U} Pr(C^U | C_z^L, D, S; W_z) \tag{9}$$

However, solving this equation is similar to solving a boolean satisfiability problem. Therefore, we may simply leverage the most recent Gibbs sampling result Ω_z obtained during EM for instantiation:

$$g_z(C^U) = \arg\max_{C^U} \sum_{C\in\Omega_z} \mathbb{1}_{C^U=C} \tag{10}$$

We break ties randomly. For validated claims, the user input is directly incorporated, i.e., $g_z(C^L) = e_z(C^L)$.

4 USER GUIDANCE IN FACT GUIDING

Having discussed a model and techniques for (i) inference based on user input and (ii) instantiation of a deterministic assignment, we now turn to strategies to guide a user in the validation. This section first defines a measure of uncertainty for a probabilistic fact database (§4.1). Subsequently, we introduce two approaches to guide the selection of claims for validation (§4.2 and §4.3), before combining them in a hybrid approach (§4.4).

4.1 Uncertainty of a Probabilistic Fact Database

The model of a probabilistic fact database, as constructed above, enables us to quantify the uncertainty related to credibility inference in order to guide a user in the validation process. Let $Q = \langle N, e, P\rangle$, with $N = \langle C, D, S\rangle$, be a probabilistic fact database. Recall that Q defines the likelihood of credibility of each claim, i.e., $P = \cup_{c\in C} Pr(c)$. Then, the overall uncertainty of the database is computed by the Shannon entropy [Shannon 2001] over a set of claims as random variables:

$$H_C(Q) = -\sum_{C\in\Omega} Pr(C;W) \log Pr(C;W) \tag{11}$$

where Ω are all possible configurations of C. In our iCRF model, it can be computed exactly by [Reyes 2013; Reyes and Neuhoff 2009]:

$$H(C;W) = \Phi(W) - \mathbb{E}_W[t(C)]^T W \tag{12}$$

where $\Phi(W) = \sum_C \prod_\pi \phi(c_i, d_k, s_k; W)$ is called the partition function and $\mathbb{E}_W[t(C)] = \nabla\Phi(W)$. Since our model is an acyclic graph with no self statistics, the partition function is computed exactly using Ising methods [Reyes 2013], which run in polynomial time.

We can further scale-up uncertainty computation by approximating the entropy in linear time, as follows:

$$H_C(Q) = -\sum_{c \in C}[Pr(c)\log Pr(c) + (1 - Pr(c))\log(1 - Pr(c))] \tag{13}$$

where the probability of each claim is obtained after the EM iterations (i.e., Eq. 6 and Eq. 7). However, this approximation neglects the mutual dependencies between claims and, thus, potentially underestimates the effects of user input on their credibility.

4.2 Uncertainty-driven User Guidance

A first heuristic to guide the selection of claims for validation aims at the maximal reduction in uncertainty under the assumption of trustworthy sources. It exploits the benefit of validating a single claim using the notion of information gain from information theory [Russell and Norvig 2003].

To capture the impact of user input on a claim c, we define a conditional variant of the entropy measure introduced earlier. Informally, it measures the expected entropy of the probabilistic fact database under specific validation input:

$$H_C(Q \mid c) = Pr(c) \times H_C(Q^+) + (1 - Pr(c)) \times H_C(Q^-) \tag{14}$$

where $Q^+ = conclude(N, e')$ (and $Q^- = conclude(N, e')$) is constructed by the incremental learning with $e'(c) = 1$ (and $e'(c) = 0$) and $e'(c') = e(c')$ for $c' \in (C \setminus \{c\})$.

To take a decision on which claim to select, we assess the expected difference in uncertainty before and after incorporating input for a claim. The respective change in entropy is the information gain that quantifies the potential benefit of knowing the true value of an unknown variable [Russell and Norvig 2003], i.e., the truth value in our case:

$$IG_C(c) = H_C(Q) - H_C(Q \mid c). \tag{15}$$

The information gain can guide the selection of a claim: we chose the one that is expected to maximally reduce the uncertainty of the probabilistic fact database. This is formalized by a selection function for uncertainty-driven user guidance:

$$select_C(C') = \underset{c \in C'}{\arg\max}\ IG_C(c) \tag{16}$$

where $C' \subseteq C$ is the set of non-validated claims. While this formulation is sound for ranking a whole set of claims, validated claims will be ranked last (since their credibility probability is either one or zero). Also, we do not need to rank the opposing claim $\neg c$ of a claim c, as their conditional entropies in Eq. 14 will be equivalent.

4.3 Source-driven User Guidance

User guidance as introduced above assumes that sources are trustworthy—an assumption that is often violated in practice. To tackle this issue, we model source trustworthiness by explicitly aggregating over all claims made by a source. More precisely, the likelihood that a source is trustworthy is measured as the fraction of its claims that are considered as true in the last EM iteration:

$$Pr(s = 1) = \frac{\sum_{c \in C_s} \mathbb{1}_{c=1}}{|C_s|} \tag{17}$$

where $C_s = \{c \in C | (c, s) \in \Pi\}$ is the set of claims connected to s in the CRF model.
Then, the uncertainty of source trustworthiness values is defined as:

$$H_S(Q) = -\sum_{s \in S}[Pr(s)\log Pr(s) + (1 - Pr(s))\log(1 - Pr(s))] \tag{18}$$

The conditional entropy when a claim c is validated is:

$$H_S(Q|c) = Pr(c) \times H_S(Q^+) + (1 - Pr(c)) \times H_S(Q^-) \tag{19}$$

where, as above, $Q^+ = conclude(N, e')$ (and $Q^- = conclude(N, e')$) is constructed by the incremental learning with $e'(c) = 1$ (and $e'(c) = 0$) and $e'(c') = e(c')$ for $c' \in (C \setminus \{c\})$.

As for the case of the first heuristic, we further capture the information gain as the difference in entropy and, based thereon, define the selection function for source-driven user guidance:

$$IG_S(c) = H_S(Q) - H_S(Q|c) \tag{20}$$

$$select_S(C') = \arg\max_{c \in C'} IG_S(c) \tag{21}$$

where $C' \subseteq C$ is the set of non-validated claims. Again, we do not need to rank opposing claims.

4.4 A Combined Approach to User Guidance

There is a trade-off between the application of the uncertainty-driven and the source-driven strategies for user guidance. Focusing solely on the former may lead to contamination of the claims from trustworthy sources by unreliable sources. An excessively source-driven approach, in turn, is undesirable as it may increase the overall user efforts significantly. Therefore, we propose a dynamic weighting procedure that, in each iteration of the validation process, helps to choose among the two strategies.

Weighting procedure. Two factors affect the choice of strategies:

Ratio of untrustworthy sources. If there is a high number of unreliable sources, the source-driven strategy is preferred. With little user input, detection of unreliable sources is difficult, though, so that the uncertainty-driven strategy is favoured in the beginning.

Error rate. The deterministic assignment g_i captures the assignments considered to be correct in the i-th iteration of the validation process. If g_i turns out to be mostly incorrect, we have evidence of unreliable sources and, thus, favour the source-driven strategy.

Both factors combined lead to a dynamic strategy. Initially, with little user input, the strategy is primarily chosen on the error rate of the deterministic assignment. At later stages, the number of inferred unreliable sources becomes the dominant factor.

The above idea is formalised based on the ratio of unreliable sources in the i-th iteration of the validation process, which is $r_i = (|\{s \in S | Pr(s) < 0.5\}|)/(|S|)$. The error rate of the deterministic assignment is computed by comparing the user input for claim c in the i-th iteration with the truth value that has been assigned to c in g_{i-1}, i.e., in the previous iteration. Here, we leverage the probability $P_{i-1}(c)$ of the probabilistic fact database $Q_{i-1} = \langle N, e_{i-1}, P_{i-1} \rangle$, of the previous iteration. If g_{i-1} assigns a credibility $v \in \{0, 1\}$ to c, the error rate is computed as:

$$\epsilon_i = 1 - Pr_{i-1}(c = v) \tag{22}$$

Using the ratio of unreliable sources r_i and the error rate ϵ_i, we define a normalised score for choosing the source-driven strategy:

$$z_i = 1 - e^{-(\epsilon_i (1-h_i) + r_i h_i)} \tag{23}$$

where $h_i = (i)/(|C|)$ is the ratio of user input. This score mediates the trade-off between the error rate ϵ_i and the ratio of untrustworthy sources r_i by the ratio of user input h_i. When the ratio h_i is small, the ratio of untrustworthy sources has less influence and the error rate is the dominant factor. When the ratio h_i is large, the ratio of unreliable sources becomes a more dominant factor.

Validation procedure. Combining the model and inference mechanism introduced in §3 with the above strategies to guide a user, we present our complete validation process for fact checking in Alg. 1. As long as the validation goal is not reached and the user effort budget has not been exhausted (line 5), selection of the claim for which user input shall be sought is done either by the source-driven or the uncertainty-driven strategy. The choice between strategies is taken by comparing factor z_i to a random number (line 7), which implements a roulette wheel selection [Goldberg 1989]. The second step (lines 11 and 12) elicits user input for the selected claim and computes the error rate according to Eq. 22. The third step (line 13-16) incorporates the user input. That is, we update the fact checking function e_{i+1}; infer the implications of user input by means of function *conclude*, which yields a new probabilistic fact database; and decide on the new deterministic assignment g_{i+1} capturing which facts are considered credible. Further, the trustworthiness of each source is updated and the ratio of unreliable sources r_i is calculated to compute score z_{i+1} (lines 18-19), which is used in the next iteration to choose between the selection strategies.

Algorithm 1: Validation process for fact checking

input : an probabilistic fact database, $Q = \langle N, e, P \rangle$, with $N = \langle C, D, S \rangle$, a validation goal Δ, and a user effort budget b.
output : the truth assignment g.

$e_0 \leftarrow (c \mapsto \ominus, c \in C)$;
$P_0 \leftarrow conclude(N, e_0)$;
$g_0 \leftarrow decide(P_0)$;
$i, z_0 \leftarrow 0$;
while *not* $\Delta \wedge i \leq b$ **do**
  // (1) Selecting a claim to validate
  $x \leftarrow random(0, 1)$;
  if $x < z_i$ **then** // Choosing the source-driven strategy
    $c \leftarrow select_S(\{c' \in C \mid e_i(c') = \ominus\})$
  else // Choosing the uncertainty-driven strategy
    $c \leftarrow select_C(\{c' \in C \mid e_i(c') = \ominus\})$
  // (2) Eliciting user input
  Elicit user input $v \in \{0, 1\}$ on c;
  $\epsilon_i = 1 - P_{i-1}(c = v)$; // Calculate error rate ϵ_i
  // (3) Incorporating user input
  $e_{i+1} \leftarrow (c \mapsto v \wedge c' \mapsto e_i(c'), c' \in C, c' \neq c)$;
  $Q_{i+1} \leftarrow conclude(N, e_{i+1})$;
  $g_{i+1} \leftarrow (c' \mapsto decide(P_{i+1}), c' \in C, e_{i+1}(c') = \ominus \wedge c' \mapsto e_{i+1}(c'), c' \in C, e_{i+1}(c') \neq \ominus)$;
  $i \leftarrow i + 1$;
  Update source trustworthiness $Pr(s), \forall s \in S$;
  Calculate ratio of unreliable sources r_i;
  $z_{i+1} = 1 - e^{-\left(\epsilon_i \left(1 - \frac{i}{|C|}\right) + r_i \frac{i}{|C|}\right)}$;
return d_i;

5 EVALUATION

In this section, we evaluate the proposed approach experimentally, using real-world datasets. We first discuss the experimental setup (§5.1), before turning to an evaluation of the following aspects of our approach:

- The runtime performance of the presented approach (§5.2).
- The efficacy of our *iCRF* model (§5.3).
- The effectiveness of user guidance (§5.4).

Since different baselines are not applicable for all experimental setting, we defer the description of baselines to each evaluation.

5.1 Experimental Setup

Datasets. We utilize the following datasets (see also Table 1):

- *Wikipedia:* This dataset contains a list of proven hoaxes and fictitious people from Wikipedia links [wik 2017]. In total, there are 1955 sources, 3228 documents, and 157 labelled claims.
- *Healthcare forum:* The dataset contains 291276 claims about side-effects of drugs extracted from 2.8M documents of 15K users on a healthcare forum (healthboards.com) [hea 2017]. We consider 529 claims that have been labelled by health experts. The claims are part of 48083 documents from 11206 sources.
- *Snopes:* This dataset contains a wide range of claims from news websites, social media, e-mails, etc [sno 2017]. The credibility of these claims has been assessed by Snopes' editors. The dataset comprises 80421 documents of 23260 sources that contain 4856 labelled claims.

Table 1. Datasets

Dataset	#Sources	#Documents	#Claims
wiki	1955	3228	157
health	11206	48083	529
snopes	23260	80421	4856

For those datasets, we derive credibility scores as follows. If a source is a website, we rely on ranking scores [Ding et al. 2003]. If a source is an author, features include personal information (age, gender) and activity logs (number of posts).

Credibility of documents is assessed using common linguistic features such as stylistic indicators (e.g. usage of modals, named entities, inferential conjunction) and affective indicators (e.g. sentiments, thematic words) [Olteanu et al. 2013].

In our experiments, we use the ground truth of the datasets to simulate user input. Model parameters are initialised with a value of 0.5, following the maximum entropy principle.

Evaluation Metrics. In addition to the uncertainty of the probabilistic fact database defined in §4, we relied on the following measures:

Relative user efforts (E_i): the number of user-validated claims i relative to the number of all claims $|C|$, i.e., $E = i/|C|$.

Precision (P_i): the correctness of the deterministic assignment at each validation step. Let $g^* : C \to \{0, 1\}$ be the correct assignment of claim credibility. Then, the precision of the deterministic assignment g_i at the i-th validation step is

$$P_i = \frac{|\{c \in C \mid g_i(c) = g^*(c)\}|}{|C|}.$$

This definition of precision is different from the one in information retrieval and binary classification [Russell and Norvig 2003]. Since the user interest is a trusted set of facts, the correctness of obtained facts shall be evaluated.

Percentage of precision improvement (R_i): a normalised version of precision, measuring relative improvements to illustrate the effect of user input. With P_0 as the initial precision, measure is defined at the i-th validation step by

$$R_i = \frac{P_i - P_0}{1 - P_0}.$$

Experimental environment. Experimental results have been obtained on an Intel Core i7 system (3.4GHz, 12GB RAM). All except the experiments on early termination (**??**) ran until the actual termination of the validation process.

5.2 Runtime Performance

Being based on direct user interactions, our approach relies on a good runtime performance. Therefore, a first experiment measures the response time, denoted by Δt, of our approach during one iteration of Alg. 1, i.e., the waiting time of a user after validating one claim. This includes the time to infer implications of user input and to select the next best claim to validate.

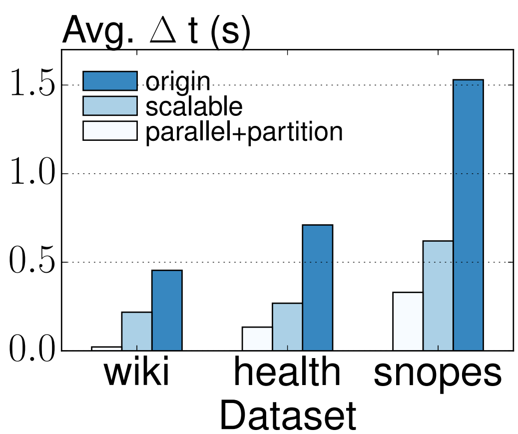

Fig. 2. Time vs. dataset

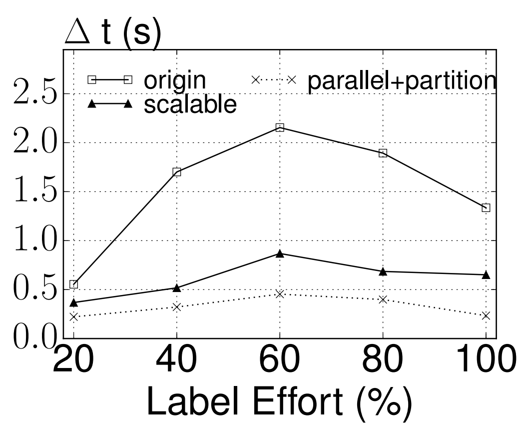

Fig. 3. Time vs. user effort

Fig. 2 shows the observed response time, averaged over 10 runs, when using the plain algorithm (*origin*), with uncertainty estimation as introduced in §4.1 (*scalable*), and with the computational optimisations (*parallel+partition*). With larger dataset size (*wiki* to *snopes*), the response time increases. However, with computational optimisations, the average response time stays below half a second, which enables immediate user interactions. Fig. 3 further illustrates for the largest dataset, *snopes*, how the response time evolves during the validation process when averaging the response time over equal bins of user effort. The time peaks between 40% and 60% of user effort, since at these levels, user input enables the most conclusions on the credibility of claims.

5.3 Efficacy of *iCRF* model

We now turn to the overall efficacy of our *iCRF* model and evaluate the estimated probabilities of credibility assignments. Since we use probabilistic information to guide validation, the probabilities should reflect the ground truth, i.e. true values of claims. For each claim, our iCRF model should assign a higher probability to correct credibility values than to incorrect ones. In the experiment, we keep track of the correct assignments and their associated probabilities while varying the user effort (0%, 20%, 40%).

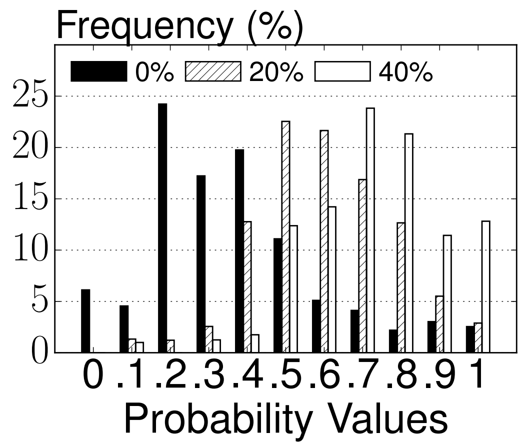

Fig. 4. Guidance benefits

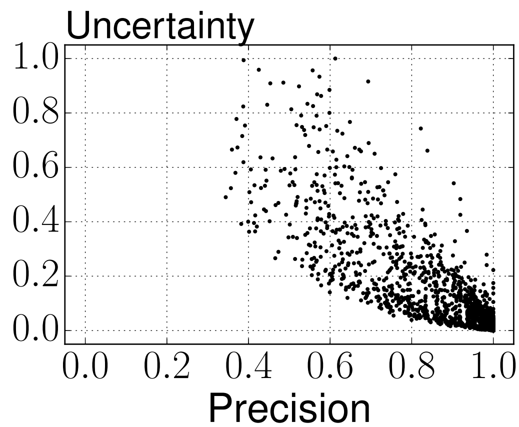

Fig. 5. Uncert. vs. prec.

Fig. 4 shows the correctness of our model via a histogram of the probability distribution over all datasets. It illustrates how often the probability assigned to a claim falls into a specific bin. Increasing the amount of user input, the range covering most of the correct credibility values shifts from lower probability bins to higher ones. Even with little user effort (20%), the number of correct assignments that are assigned a value ≥ 0.5 is very high. This highlights that user input indeed enables better assessment of the credibility of claims.

5.4 Effectiveness of User Guidance

Relation between uncertainty and precision. We verify the underlying assumption of our approach to user guidance, i.e., that the uncertainty of a fact database, see §4, is correlated with the precision of the deterministic assignment. In this experiment, the uncertainty-driven guidance strategy was applied to all datasets (100 runs each), until precision reaches 1.0. Fig. 5 depicts the results in terms of precision and normalized uncertainty (i.e., uncertainty divided by the maximum value of the run). There is a strong correlation between both measures (Pearson's correlation coefficient is -0.8523). Hence, uncertainty is indeed a truthful indicator of the result correctness.

Guidance strategies. Turning to the guidance strategies, we mimic the user by means of the ground-truth, until precision reaches 1.0. We compare the proposed approach (*hybrid*) with two methods: *random*, which selects a claim randomly for validation; and *baseline*, which selects the claim that is most 'problematic' in terms of the entropy of its probability. Intuitively, this baseline method outperforms random selection since it targets claims that are on the edge of being considered credible and thus the major sources of model uncertainty.

Fig. 6 shows the results for the *health* and *snopes* datasets (*wiki* yields similar trends). Our approach (*hybrid*) clearly outperforms random selection and the baseline method. For example, using the *snopes* dataset, our approach leads to a precision above 0.9 with input on only 31% of the claims, whereas the other methods require validation of at least 84% to reach the same level of precision. Note that *snopes* has low initial precision due to the frequency of fake news and rumours on news websites and social media. Yet, the resulting issues are quickly fixed with user input that eliminates many claims that related to each other and jointly not credible.

6 RELATED WORK

Truth finding on the Web. Given a set of data items claimed by multiple sources, the truth finding (aka fact checking) problem is to determine the true values of each claimed item [Dong and Naumann 2009]. Existing work in this space, also

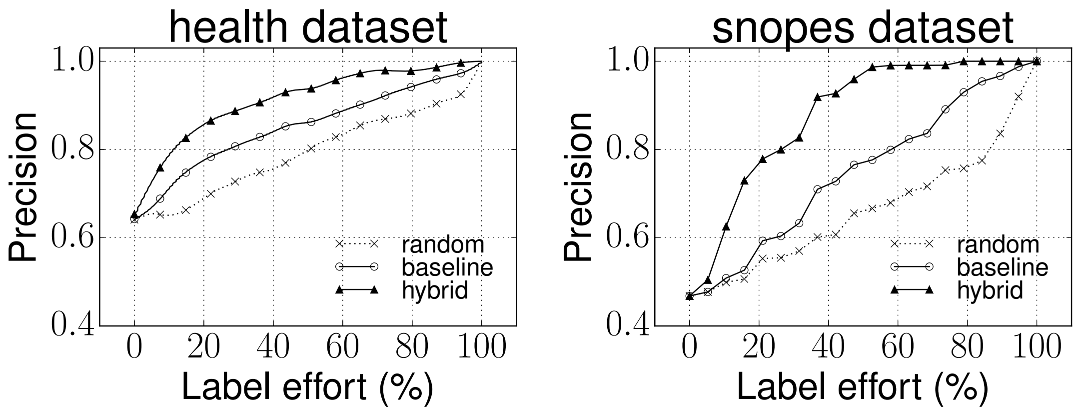

Fig. 6. Effectiveness of guiding

considers mutual reinforcing relations between sources and data items, e.g., by Bayesian models [Cong et al. 2018; Duong et al. 2017; Hung et al. 2015, 2014b, 2018; Tran et al. 2011; Zhao et al. 2012], maximum likelihood estimation [Wang et al. 2012], and latent credibility analysis [Pasternack and Roth 2013]. However, these techniques neglect posterior knowledge on input and rely on domain-specific information about sources and data, such as the dependencies between sources [Dong et al. 2010] and temporal data evolution [Dong et al. 2009].

Truth finding is also known as *knowledge verification* [Li et al. 2017a] and *credibility analysis* [Mukherjee and Weikum 2015]. Existing automatic techniques mostly look at features of data, such as number of relevant articles, mentions, keywords, and popularity, which are noisy and can be easily dominated by information cascades [Friggeri et al. 2014; Li et al. 2017a]. Again, posterior knowledge on user input cannot be incorporated. Also, approaches based on gradient-descent [Mukherjee and Weikum 2015] only optimise model parameters, but neglect external probability constraints. Fact extraction may be performed by diverse data representations, e.g., knowledge bases [Dong et al. 2014], web tables [Cafarella et al. 2008], semi-structured data [Etzioni et al. 2004], or free text [Banko et al. 2007]. Other work considers co-occurrence information and evidential logs [Li et al. 2017a, 2011]. Yet, they are limited to quantitative information and do not employ logical reasoning and semantics. For example, in [Li et al. 2017a] identifies unpopular facts based on the number of mentions in Web data. Our work is orthogonal to all these works. First our model relies on an abstract data representation, which is not specific to a particular domain. Second, our principles of user guidance can be adapted for many of the aforementioned truth finding techniques, as the notion of probabilistic uncertainty is generic [Shannon 2001].

User guidance. Guiding users or experts has been studied in data integration, data repair, crowdsourcing, and recommender systems [Hung et al. 2017, 2014a; Jeffery et al. 2008; Li et al. 2017b; Parameswaran et al. 2012; Yakout et al. 2011; Yin et al. 2016a,b]. Most of this work proposes a decision theoretic framework to rank candidate data for validation to improve the quality of a dataspace. Following this line, we rely on models from the fields of Decision Theory and Active Learning [Rubens et al. 2015; Russell and Norvig 2003]. Despite the similarities in the applied models, our approach differs from the aforementioned ones in several ways. First, unlike existing work that focuses on structured data that is deterministic and traceable, we cope with Web data that is unreliable and potentially non-deterministic. Second, existing work relies on two main sources of information (data and data provider), whereas we incorporate individual features as well as causal and mutual relations between heterogeneous data (sources, documents, and claims). Third, the benefit of user input depends on the application domain and, unlike the above works, our approach is tailored to the specific characteristics of Web data.

Our setting is different from Active Learning [Settles 2010], as we do not require any training data for a user to begin the validation process. Moreover, we incrementally incorporate user input without devising a model from scratch upon receiving new labels. However, we note that stopping criteria for feedback processes have been studied in Active Learning, e.g. using held-out labels [Mozafari et al. 2014] and performance estimation [Laws and Schätze 2008]. However, these methods are applicable only for specific classifiers and do not incorporate human factors. By means of a unified probabilistic model, we have been able to propose several criteria for early termination that turned out to be very effective.

7 CONCLUSIONS

In this paper, we proposed an approach to overcome the limitations of existing methods for automatic and manual fact checking. We introduced an iterative validation process, which, based on a probabilistic model, selects a claim for which validation is most beneficial, infers the implications of the user input on the fact database as a whole, and enables grounding of the credibility values of claims at any time. We further proposed different strategies to guide users and presented optimisations that increase the efficiency and robustness of the validation process. Our evaluation showed that our approach outperforms respective baselines methods significantly, saving up to 53% of user effort when striving for 90% result precision.

REFERENCES

2017. DBPedia. http://www.dbpedia.org. (2017).
2017. Freebase. http://www.freebase.com. (2017).
2017. http://resources.mpi-inf.mpg.de/impact/peopleondrugs/data.tar.gz. (2017).
2017. http://resources.mpi-inf.mpg.de/impact/web_credibility_analysis/Snopes.tar.gz. (2017).
2017. http://resources.mpi-inf.mpg.de/impact/web_credibility_analysis/Wikipedia.tar.gz. (2017).
2017. YAGO. http://www.mpi-inf.mpg.de/yago. (2017).
Michele Banko, Michael J Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. 2007. Open Information Extraction from the Web.. In *IJCAI*. 2670–2676.
Jose A Blakeley, Per-Ake Larson, and Frank Wm Tompa. 1986. Efficiently updating materialized views. In *SIGMOD*. 61–71.
Michael J Cafarella, Alon Halevy, Daisy Zhe Wang, Eugene Wu, and Yang Zhang. 2008. Webtables: exploring the power of tables on the web. In *VLDB*. 538–549.
Phan Thanh Cong, Nguyen Thanh Toan, Nguyen Quoc Viet Hung, and Bela Stantic. 2018. Minimizing Efforts in Reconciling Participatory Sensing Data. In *WIMS*. 49.
Chris Ding, Xiaofeng He, Parry Husbands, Hongyuan Zha, and Horst Simon. 2003. PageRank, HITS and a unified framework for link analysis. In *SDM*. 249–253.
Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. 2014. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *KDD*. 601–610.
Xin Luna Dong, Laure Berti-Equille, Yifan Hu, and Divesh Srivastava. 2010. Solomon: Seeking the truth via copying detection. In *VLDB*. 1617–1620.
Xin Luna Dong, Laure Berti-Equille, and Divesh Srivastava. 2009. Truth discovery and copying detection in a dynamic world. In *VLDB*. 562–573.
Xin Luna Dong and Felix Naumann. 2009. Data fusion: resolving data conflicts for integration. In *VLDB*. 1654–1655.
Xin Luna Dong, Barna Saha, and Divesh Srivastava. 2012. Less is more: Selecting sources wisely for integration. In *VLDB*. 37–48.
Chi Thang Duong, Quoc Viet Hung Nguyen, Sen Wang, and Bela Stantic. 2017. Provenance-Based Rumor Detection. In *ADC*. 125–137.
Oren Etzioni, Michael Cafarella, Doug Downey, Stanley Kok, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S Weld, and Alexander Yates. 2004. Web-scale information extraction in knowitall:(preliminary results). In *WWW*. 100–110.
Adrien Friggeri, Lada A Adamic, Dean Eckles, and Justin Cheng. 2014. Rumor Cascades.. In *ICWSM*. 101–110.
David E. Goldberg. 1989. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman.
Kazi Saidul Hasan and Vincent Ng. 2014. Why are You Taking this Stance? Identifying and Classifying Reasons in Ideological Debates.. In *EMNLP*. 751–762.
Nguyen Quoc Viet Hung, Chi Thang Duong, Nguyen Thanh Tam, Matthias Weidlich, Karl Aberer, Hongzhi Yin, and Xiaofang Zhou. 2017. Argument discovery via crowdsourcing. *VLDB J.* (2017), 511–535.
Nguyen Quoc Viet Hung, Nguyen Thanh Tam, Vinh Tuan Chau, Tri Kurniawan Wijaya, Zoltán Miklós, Karl Aberer, Avigdor Gal, and Matthias Weidlich. 2015. SMART: A tool for analyzing and reconciling schema matching networks. In *ICDE*. 1488–1491.
Nguyen Quoc Viet Hung, Nguyen Thanh Tam, Ngoc Tran Lam, and Karl Aberer. 2013. An Evaluation of Aggregation Techniques in Crowdsourcing. In *WISE*. 1–15.
Nguyen Quoc Viet Hung, Nguyen Thanh Tam, Zoltán Miklós, and Karl Aberer. 2014b. Reconciling Schema Matching Networks Through Crowdsourcing. *EAI* (2014), e2.
Nguyen Quoc Viet Hung, Nguyen Thanh Tam, Zoltán Miklós, Karl Aberer, Avigdor Gal, and Matthias Weidlich. 2014a. Pay-as-you-go reconciliation in schema matching networks. In *ICDE*. 220–231.
Nguyen Quoc Viet Hung, Kai Zheng, Matthias Weidlich, Bolong Zheng, Hongzhi Yin, Nguyen Thanh Tam, and Bela Stantic. 2018. What-if Analysis with Conflicting Goals: Recommending Data Ranges for Exploration. In *ICDE*. 1–12.
Shawn R. Jeffery, Michael J. Franklin, and Alon Y. Halevy. 2008. Pay-as-you-go user feedback for dataspace systems. In *SIGMOD*. 847–860.
Florian Laws and Hinrich Schätze. 2008. Stopping criteria for active learning of named entity recognition. In *ICCL*. 465–472.
Jens Lehmann, Daniel Gerber, Mohamed Morsey, and Axel-Cyrille Ngonga Ngomo. 2012. Defacto-deep fact validation. In *ISWC*. 312–327.
Furong Li, Xin Luna Dong, Anno Langen, and Yang Li. 2017a. Knowledge verification for long-tail verticals. In *VLDB*. 1370–1381.
Xian Li, Weiyi Meng, and Clement Yu. 2011. T-verifier: Verifying truthfulness of fact statements. In *ICDE*. 63–74.
Yaguang Li, Han Su, Ugur Demiryurek, Bolong Zheng, Tieke He, and Cyrus Shahabi. 2017b. PaRE: A System for Personalized Route Guidance. In *WWW*. 637–646.
Chih-Jen Lin, Ruby C Weng, and S Sathiya Keerthi. 2008. Trust region newton method for logistic regression. *JMLR* (2008), 627–650.
Andrew McCallum, Kedar Bellare, and Fernando Pereira. 2005. A Conditional Random Field for Discriminatively-trained Finite-state String Edit Distance. In *UAI*. 388–395.
Barzan Mozafari, Purna Sarkar, Michael Franklin, Michael Jordan, and Samuel Madden. 2014. Scaling Up Crowd-Sourcing to Very Large Datasets: A Case for Active Learning. In *VLDB*. 125–136.
Subhabrata Mukherjee and Gerhard Weikum. 2015. Leveraging joint interactions for credibility analysis in news communities. In *CIKM*. 353–362.
Subhabrata Mukherjee, Gerhard Weikum, and Cristian Danescu-Niculescu-Mizil. 2014. People on drugs: credibility of user statements in health communities. In *KDD*. 65–74.
Alexandra Olteanu, Stanislav Peshterliev, Xin Liu, and Karl Aberer. 2013. Web credibility: Features exploration and credibility prediction. In *ECIR*. 557–568.

Aditya G Parameswaran, Hector Garcia-Molina, Hyunjung Park, Neoklis Polyzotis, Aditya Ramesh, and Jennifer Widom. 2012. Crowdscreen: Algorithms for filtering data with humans. In *SIGMOD*. 361–372.
Jeff Pasternack and Dan Roth. 2013. Latent credibility analysis. In *WWW*. 1009–1020.
Matthew G Reyes. 2013. Covariance and entropy in Markov random fields. In *ITA*. 1–6.
Matthew G Reyes and David L Neuhoff. 2009. Entropy bounds for a Markov random subfield. In *ISIT*. 309–313.
Neil Rubens, Mehdi Elahi, Masashi Sugiyama, and Dain Kaplan. 2015. Active learning in recommender systems. In *Recommender systems handbook*. 809–846.
Stuart J. Russell and Peter Norvig. 2003. *Artificial Intelligence: A Modern Approach*. Pearson Education.
Mikkel Schmidt. 2009. Linearly constrained bayesian matrix factorization for blind source separation. In *NIPS*. 1624–1632.
Burr Settles. 2010. Active learning literature survey. *University of Wisconsin, Madison* 52, 55-66 (2010), 11.
Claude E Shannon. 2001. A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review* (2001), 3–55.
Le Hung Tran, Quoc Viet Hung Nguyen, Ngoc Hoan Do, and Zhixian Yan. 2011. *Robust and hierarchical stop discovery in sparse and diverse trajectories*. Technical Report. EPFL.
Dong Wang, Lance Kaplan, Hieu Le, and Tarek Abdelzaher. 2012. On truth discovery in social sensing: A maximum likelihood estimation approach. In *IPSN*. 233–244.
You Wu, Pankaj K Agarwal, Chengkai Li, Jun Yang, and Cong Yu. 2014. Toward computational fact-checking. In *VLDB*. 589–600.
Mohamed Yakout, Ahmed K Elmagarmid, Jennifer Neville, Mourad Ouzzani, and Ihab F Ilyas. 2011. Guided data repair. In *VLDB*. 279–289.
Hongzhi Yin, Zhiting Hu, Xiaofang Zhou, Hao Wang, Kai Zheng, Nguyen Quoc Viet Hung, and Shazia Wasim Sadiq. 2016a. Discovering interpretable geo-social communities for user behavior prediction. In *ICDE*. 942–953.
Hongzhi Yin, Xiaofang Zhou, Bin Cui, Hao Wang, Kai Zheng, and Nguyen Quoc Viet Hung. 2016b. Adapting to User Interest Drift for POI Recommendation. *TKDE* (2016), 2566–2581.
Bo Zhao, Benjamin IP Rubinstein, Jim Gemmell, and Jiawei Han. 2012. A bayesian approach to discovering truth from conflicting sources for data integration. In *VLDB*. 550–561.