Compact Explanation of Data Fusion Decisions

Xin Luna Dong*
Google Inc.
lunadong@google.com

Divesh Srivastava
AT&T Labs–Research
divesh@research.att.com

ABSTRACT

Despite the abundance of useful information on the Web, different Web sources often provide conflicting data, some being out-of-date, inaccurate, or erroneous. *Data fusion* aims at resolving conflicts and finding the truth. Advanced fusion techniques apply iterative MAP (Maximum A Posteriori) analysis that reasons about trustworthiness of sources and copying relationships between them. Providing explanations for such decisions is important for a better understanding, but can be extremely challenging because of the complexity of the analysis during decision making.

This paper proposes two types of explanations for data-fusion results: *snapshot explanations* take the provided data and any other decision inferred from the data as evidence and provide a high-level understanding of a fusion decision; *comprehensive explanations* take only the data as evidence and provide an in-depth understanding of a fusion decision. We propose techniques that can efficiently generate correct and compact explanations. Experimental results show that (1) we generate correct explanations, (2) our techniques can significantly reduce the sizes of the explanations, and (3) we can generate the explanations efficiently.

Categories and Subject Descriptors

H.2 [**Database Management**]: Heterogeneous Databases

Keywords

Data fusion, copy detection, explanation

1. INTRODUCTION

Despite the abundance of useful information on the Web, different Web sources often provide conflicting data, some being out-of-date, inaccurate, or erroneous. A recent study [18] shows that even for stock and flight, where people usually obtain data from the Web and the quality of the data can have a big effect on people's daily lives, inconsistent data are provided for 70% of the data items. Resolving such conflicts and finding the values that best reflect the real world is extremely important for cleaning Web content, constructing knowledge bases, and improving user experiences.

Data fusion (surveyed in [2, 18]) aims at resolving conflicts and finding the truth. It has been shown that simply choosing the value provided by the most sources often leads to incorrect results [18].

*Research conducted at AT&T Labs-Research.

WWW 2013, May 13–17, 2013, Rio de Janeiro, Brazil.
ACM 978-1-4503-2035-1/13/05.

Table 1: Data from five sources on the affiliation of five DB researchers. False values are in italic font.

	S_1	S_2	S_3	S_4	S_5
Stonebraker	MIT	*berkeley*	MIT	MIT	*MS*
Dewitt	MSR	msr	*UWisc*	*UWisc*	*UWisc*
Bernstein	MSR	msr	MSR	MSR	MSR
Carey	UCI	*at&t*	*BEA*	*BEA*	*BEA*
Halevy	Google	google	*UW*	*UW*	*UW*

State-of-the-art fusion techniques consider in addition (i) trustworthiness of the providers such that data provided by more trustworthy sources are trusted more [8, 9, 12, 20, 21, 23, 24, 25, 26, 27], and (ii) copying relationships between the providers such that copied data are ignored in truth finding [1, 6, 8, 9].

In real systems, simply presenting data-fusion results is often insufficient. It is natural to ask "*Why is this value rather than some other value provided by other sources considered true?*" Only when we facilitate "what" with "why", can we achieve a better understanding of the data-fusion decision, which is not only valuable for data consumers, but also useful for diagnosis.

Explaining such decisions is important, but challenging. First, *Bayesian analysis*, specifically, *MAP (Maximum A Posteriori) analysis*, is conducted for decision making, including deciding the true value, judging whether a source copies from another, and so on. Unlike conventional (provenance-style) reasoning, MAP analysis considers all alternate choices, computes the inverse probability of the observed data conditioned on each choice, and then computes the probability of each alternative accordingly. We are not aware of any existing techniques that explain MAP reasoning ([11, 17] explained evidence propagation in Bayesian networks, which is different). As we illustrate next, an exhaustive description of the underlying MAP analysis can be hard to understand and frustrating as an explanation.

EXAMPLE 1.1. *Consider data provided by five sources on the affiliation of five DB researchers (Table 1). Source S_1 provides all correct affiliations; S_2 provides affiliation names in lower case; S_4 and S_5 copy from S_3, while S_5 provides the value for* Stonebraker *independently. We are able to find all correct affiliations if we apply the MAP analysis in [8], but it is natural to ask "Why is* UCI *considered as the correct affiliation of* Carey*?" Suppose we know the accuracy of the sources and probability of copying between sources (we explain in Sec.2 how we may obtain them), a detailed (and possibly agonizing) explanation can go like this.*

Three values are provided for Carey's affiliation. If UCI is true, then we reason as follows. (1) Source S_1 provides the correct value. Since S_1 has accuracy .97, the probability that it provides this correct value is .97. (2) Source S_2 provides a wrong value. Since S_2 has accuracy .61, the probability that it provides a wrong value is $1 - .61 = .39$. If we assume there are 100 uniformly distributed wrong values in the domain, the probability that S_2 provides the

particular wrong value AT&T is $\frac{.39}{100} = .0039$. (3) Source S_3 provides a wrong value. Since S_3 has accuracy .4, the probability that it provides BEA is $\frac{1-.4}{100} = .006$.[1] (4) Source S_4 either provides a wrong value independently or copies this wrong value from S_3. It has probability .98 to copy from S_3, so probability $1 - .98 = .02$ to provide the value independently; in this case, its accuracy is .4 so the probability that it provides BEA is .006. (5) Source S_5 either provides a wrong value independently or copies this wrong value from S_3 or S_4. It has probability .99 to copy from S_3 and probability .99 to copy from S_4, so probability $(1 - .99)(1 - .99) = .0001$ to provide the value independently; in this case, its accuracy is .21, so the probability that it provides BEA is .0079. Thus, the probability of our observed data conditioned on UCI being true is $.97 * .0039 * .006 * .006^{.02} * .0079^{.0001} = 2.1 * 10^{-5}$.

If AT&T is true, the probability of our observed data is $9.9 * 10^{-7}$ (details skipped). If BEA is true, the probability of our observed data is $4.6 * 10^{-7}$. If none of the provided values is true, the probability of our observed data is $6.3 * 10^{-9}$. Thus, UCI has the *maximum a posteriori* probability to be true (its conditional probability is .91 according to the Bayes Rule).

Obviously, such an explanation gives too many details unnecessarily and is extremely verbose, so is very difficult to understand.

A much simpler explanation might be "(1) S_1, the provider of value UCI, has the highest accuracy, and (2) copying is very likely between S_3, S_4, and S_5, the providers of value BEA*". For most purposes, this level of detail is adequate (further details can be provided on demand). However, automatically extracting such* key *evidence is not easy.* □

The second challenge for explanation comes from the iterative reasoning in inter-dependent tasks in data fusion, such as quantifying trustworthiness of sources, detecting copying between sources, and finding correct values. Existing work on explaining iterative reasoning (*e.g.*, [22]) provides exhaustive answers, such as finding all extraction patterns that contribute to an extracted tuple in data extraction, but does not show how to explain the iterative process.

EXAMPLE 1.2. *Continue with Ex.1.1. Given the proposed explanation, natural subsequent questions might be (1) why S_1 is considered as having a higher accuracy than other sources and (2) why copying is considered likely between $S_3 - S_5$.*

Careful choices need to be made in answering these questions. Taking the copying between S_3 and S_4 as an example, the explanation might be "S_3 and S_4 share all five values, and especially, make the same three mistakes UWisc, BEA, UW; this is unusual for independent sources, so copying is likely". *This explanation would further trigger explanation for why* UWisc, BEA, UW *are wrong. However, recall that one reason for* BEA *to be considered wrong (*i.e.*,* UCI *being correct) is the copying between $S_3 - S_5$, so we end up with a circular explanation, which is undesirable.*

On the other hand, if we provide a provenance-style explanation and trace back the iterations (see Fig.1, which we shall explain later), the explanation again can be verbose and repetitive, containing a lot of highly similar fragments. □

In this paper we propose two types of explanations. For a high-level understanding of a fusion decision, we provide *snapshot explanations* that take the provided data and any *other* decision inferred from the data as evidence. The explanation in Ex.1.1 is a snapshot explanation. For an in-depth understanding of a fusion decision, we provide *comprehensive explanations* that take only the provided data as evidence and explain any decision that requires inference over the data. This paper focuses on how to *find and organize evidence that we would show in each type of explanation*; how to present the evidence (*i.e.*, which words and layout to use, whether to use text, tables, or graphs) to improve the understandability and user studies to measure this impact are beyond the scope of this paper.

We have three goals in producing such explanations. First, the evidence we show should be consistent with the MAP analysis and give the *correct* reasoning. For example, MAP analysis considers various alternate choices and reasons about them using all available positive and negative evidence, so showing only the positive evidence to explain a decision is inappropriate. Second, rather than providing a big chunk of evidence that contains every detail of the MAP analysis, which can be long and overwhelming, it is desirable that the evidence lists are *succinct*; indeed, succinctness has been a goal for explanation in the literature [14, 15]. Third, explanations are often generated at runtime on demand; thus, the evidence should be selected *efficiently*.

To the best of our knowledge, this paper is the first that aims at explaining data fusion decisions made by iterative MAP analysis. In particular, we make the following contributions.

1. We propose *explaining our decisions* by snapshot explanations, which list both positive and negative *evidence* considered in MAP. We show how we efficiently shorten such explanations by categorizing and aggregating evidence and selectively removing *unimportant* evidence.
2. We propose *explaining our (snapshot) explanations*[2] by comprehensive explanations, which construct a DAG (directed acyclic graph) where children nodes represent evidence for the parent nodes according to the iterations. We show how we efficiently shorten such explanations by considering only the *critical* points at which we change our decision in the iterations.
3. We show through experiments on real-world data that (i) we generate correct explanations, (ii) our techniques can significantly reduce the size of the explanations, and (iii) our algorithms are efficient.

We have implemented our techniques for snapshot explanations in SOLOMON[3] [7] and demonstrated a text presentation and a graph presentation for the same set of selected evidence. Our techniques apply to data fusion approaches that conduct MAP analysis or iterative reasoning [1, 6, 8, 12, 20, 21, 24, 25, 26, 27]; however, the core ideas, including how to explain iterative MAP analysis and how to efficiently shorten such explanations, are novel and not discussed in any previous work. Our ideas for snapshot explanations can be adapted to explain other types of MAP decisions (*e.g.*, classification), and our ideas for comprehensive explanations can be applied in explaining iterative reasoning involving confidence or probabilities (*e.g.*, iterative data extraction).

In this paper, Sec.2 defines our problem and briefly reviews data fusion techniques. Sec.3-4 describe snapshot and comprehensive explanations. Sec.5 presents experiments. Sec.6 discusses related work and Sec.7 concludes.

2. PRELIMINARIES

This paper studies how to explain iterative MAP analysis. We consider two types of explanations: *Snapshot explanations* provide

[1] We have omitted repeating some words and some details that would appear in such a detailed explanation to save space.

[2] Lord Byron wrote in *Don Juan* "I wish he would explain his explanation."

[3] http://www2.research.att.com/∼yifanhu/SourceCopying/

Table 2: Main notations in this paper.

Notation	Meaning
$\mathcal{S}$	The set of sources in the data set.
$\mathcal{D}$	The set of data items in consideration.
Φ	Observation of data by sources S and S'.
$\Phi_D(S)$	Observation of data by S on item D.
$P(\Phi\|S\perp S')$	Probability of S's and S''s data conditioned on S and S' being independent (similar for condition $S \to S'$ or $S' \to S$).
$P_{ind}(\Phi(S))$	Probability of S's data conditioned on S being independent of S'.
$P(\Phi(S)\|S \to S')$	Probability of S's data conditioned on S being a copier of S'; abbreviated from $P(\Phi(S)\|S \to S', \Phi(S'))$.

a high-level understanding of a fusion decision; *comprehensive explanations* provide an in-depth understanding of a fusion decision.

DEFINITION 2.1. *Let W be a MAP decision in data fusion.*

- *A* snapshot explanation *for W takes the provided data and all decisions in fusion except W as evidence and explains how W is reached.*
- *A* comprehensive explanation *for W takes only the data as evidence and explains how W is reached.* □

We next briefly review advanced data fusion techniques; notations are summarized in Table 2.

Overview: Consider a set $\mathcal{D}$ of *data items*, each representing a particular aspect of a real-world object (*e.g.*, the affiliation of a researcher) and having a single true value. Also consider a set $\mathcal{S}$ of *data sources* that provide data on these data items. For the same item, different sources may provide conflicting values. Data fusion aims at *finding the true value for each item according to the provided values*.

Advanced fusion techniques [6, 8, 24] find the true value on item $D \in \mathcal{D}$ by MAP: it computes the inverse probability that the observed data on D are provided conditioned on each value in D's domain being true, and selects the value with the highest probability. The probability computation considers the following aspects.

1. *Source accuracy:* The probability that a source $S \in \mathcal{S}$ provides a true value depends on its *accuracy*: the higher the accuracy, the higher the probability (opposite for a false value). The accuracy of S is computed as the average probability of S's values being true [8, 24].
2. *Copying relationship:* We wish to consider only independently provided values. Copying is considered likely if we observe a lot of common unpopular data, especially common false values, since it is typically much less likely for independent sources to share such data.

There is inter-dependence between truth discovery, copy detection, and source accuracy; techniques in [1, 8, 12, 21, 24] conduct iterative computation until the results converge.

In this paper we illustrate our techniques on explaining no-copying between two sources. We thus give more details on MAP analysis for copy detection.

Copy detection: Let Φ be our observation of the data provided by sources $S, S' \in \mathcal{S}$. Let $S \to S'$ denote that S copies from S' and $S \perp S'$ denote that S and S' do not copy from each other; then, $P(S \to S') + P(S' \to S) + P(S \perp S') = 1$ (no-loop copying is assumed in previous work; that is, $S \to S'$ and $S' \to S$ do not happen together). Assuming $0 < \alpha < .5$ is the *a priori* probability of a source copying from another and $\beta = 1 - 2\alpha$, we obtain the following equation according to the Bayes rule.

$$P(S \perp S'|\Phi) = \frac{\beta P(\Phi|S \perp S')}{\alpha P(\Phi|S \to S') + \alpha P(\Phi|S' \to S) + \beta P(\Phi|S \perp S')}. \quad (1)$$

Let $P_{ind}(\Phi(S))$ be the probability of S providing its data conditioned on it being independent of S', we have $P(\Phi|S \perp S') = P_{ind}(\Phi(S))P_{ind}(\Phi(S'))$ as both sources provide its data independently. Let $P(\Phi(S)|S \to S')$ (abbreviated from $P(\Phi(S)|S \to S', \Phi(S'))$ for space consideration) denote the probability of S providing its data conditioned on it being a copier of S', we have $P(\Phi|S \to S') = P(\Phi(S)|S \to S')P_{ind}(\Phi(S'))$, as S' provides its data independently. Assuming independence between different data items and denoting the observation for S on D by $\Phi_D(S)$, we have

$$P(\Phi|S \perp S') = \Pi_{D \in \mathcal{D}} P_{ind}(\Phi_D(S))P_{ind}(\Phi_D(S')); \quad (2)$$

$$P(\Phi|S \to S') = \Pi_{D \in \mathcal{D}} P(\Phi_D(S)|S \to S')P_{ind}(\Phi_D(S')). \quad (3)$$

When computing $P_{ind}(\Phi_D(S))$, [6] considers (but is not limited to) three aspects: the probability of S providing data on D, that of S providing the observed value, $\Phi_{D.val}(S)$, and that of S using the observed format, $\Phi_{D.fmt}(S)$. The product of them is taken:

$$P_{ind}(\Phi_D(S)) = P_{ind}(\Phi_D(S) \neq \emptyset) \cdot P_{ind}(\Phi_{D.val}(S)) \cdot P_{ind}(\Phi_{D.fmt}(S)). \quad (4)$$

We skip details of probability computation (see [6]), as it is unimportant in this paper. Note that a source may appear to copy from another source when there is actually a co-copying or transitive copying relationship; [6] shows how to adjust probability computation for this case and we can easily adapt our approach accordingly.

When computing $P(\Phi_D(S)|S \to S')$, note that a copier may or may not copy on a particular data item, and if it copies the value, it may or may not keep the same format. [6] considers the *selectivity* (probability of copying on a data item), denoted by s, and the probability of keeping the same format in copying, denoted by k ($0 \leq s, k \leq 1$ and [6, 8] discussed how to set them). As an example, in case S provides the same value as S' but uses a different format, we would consider the possibility that S provides the item independently (with probability $1 - s$) and the possibility that S copies it from S' but changes the format (with probability $s(1 - k)$); thus,

$$P(\Phi_D(S)|S \to S') = (1-s)P_{ind}(\Phi_D(S)) + s(1-k)P_{ind}(\Phi_{D.fmt}(S)). \quad (5)$$

As another example, in case S provides a different value, we would only consider the possibility that S provides the item independently (with probability $1 - s$):

$$P(\Phi_D(S)|S \to S') = (1 - s)P_{ind}(\Phi_D(S)). \quad (6)$$

Consider the case that S provides a rare data item, provides a particular false value, or uses an unpopular format. When S' has the same behavior, this probability conditioned on $S \to S'$ can be much higher than that conditioned on $S \perp S'$, so such observations serve as strong evidence for copying.

EXAMPLE 2.2. *Continue with Ex.1.1 and consider S_1 and S_2. They share neither rare data items nor false values and they use different formats, so copying is unlikely. With $\alpha = .25$, $s = k = .8$, the MAP analysis goes as follows.*

We start with $P(\Phi|S_1 \perp S_2)$, which requires computing $P_{ind}(\Phi_D(S_1))$ and $P_{ind}(\Phi_D(S_2))$ for each $D \in \mathcal{D}$ (Eq.(2)). All values S_1 provides are correct. Assuming we have decided that the accuracy of S_1 is .97, then the probability for S_1 to provide a true value is .97. On the other hand, as S_1 provides all data items and uses consistent formatting, the probability of providing a particular item and

Table 3: List explanation for no-copying between S_1 and S_2.

	Score	Evidence
Pos	3.2	S_1 provides different values from S_2 on 2 items
	3.06	Among the items for which S_1 and S_2 provide the same value, S_1 uses different formats for 3 items
	.7	The *a priori* belief is that S_1 is more likely to be independent of S_2
Neg	.06	S_1 provides the same *true* value for 3 items as S_2

that of using the format on a data item are both 1. Thus, for each $D \in \mathcal{D}$ *we have* $P_{ind}(\Phi_D(S_1)) = 1 * .97 * 1 = .97$ *(Eq.(4)). In a similar way, assuming* S_2 *has accuracy .61 and there are 100 uniformly distributed false values, we compute* $P_{ind}(\Phi_D(S_2)) = .61$ *if* S_2 *provides a true value on D, and* $P_{ind}(\Phi_D(S_2)) = \frac{1-.61}{100} = .0039$ *if* S_2 *provides a false value on D. Thus,* $P(\Phi|S_1 \perp S_2) = (.97^5) * (.61^3 * .0039^2) = 3 * 10^{-6}$.

Next consider $P(\Phi|S_1 \rightarrow S_2)$*, which requires computing* $P(\Phi_D(S_1)|S_1 \rightarrow S_2)$ *and* $P_{ind}(\Phi_D(S_2))$ *for each* $D \in \mathcal{D}$ *(Eq.(3)). Source* S_1 *shares three values with* S_2 *and they are all correct. According to Eq.(5), the probability for such item D is* $P(\Phi_D(S_1)|S_1 \rightarrow S_2) = (1 - .8) * .97 + .8 * (1 - .8) * 1 = .354$. *On the other hand,* S_1 *provides two different values from* S_2 *and each of them is true. According to Eq.(6), the probability for such data item D is* $P(\Phi_D(S_2)|S_1 \rightarrow S_2) = (1 - .8) * .97 = .194$. *Thus,* $P(\Phi|S_1 \rightarrow S_2) = (.354^3 * .194^2) * (.61^3 * .0039^2) = 5.8 * 10^{-9}$.

Similarly, $P(\Phi|S_2 \rightarrow S_1) = 2.3 * 10^{-7}$. *According to Eq.(1),* $P(S_1 \perp S_2|\Phi) = \frac{.5*3*10^{-6}}{.5*3*10^{-6}+.25*5.8*10^{-9}+.25*2.3*10^{-7}} = .96$*, so no-copying is very likely.* □

Note again that the reasoning in the example is how a detailed description of the MAP analysis would look like (many details already skipped) for a no-copying decision. Obviously it is overwhelming, especially when only a high-level understanding is needed. We next show how we can explain such decisions more elegantly.

3. EXPLAINING THE DECISION

We start with snapshot explanations: given a decision W, we take the data and all decisions made at the convergence round except W as input and explain W. Snapshot explanations are often sufficient by themselves, and are also important building blocks for comprehensive explanations as we show shortly. We describe how we generate the explanation that strictly follows the MAP analysis (Sec.3.1-3.2), then show how to shorten it (Sec.3.3-3.4).

3.1 List explanation

MAP analysis considers all possible choices, collects *evidence* and computes the probability for each of them. To explain a decision W, rather than showing only the positive evidence for W, we shall show for each alternative W' that the accumulated evidence for W is stronger than that for W'. We thus propose the following form for a snapshot explanation.

DEFINITION 3.1 (LIST EXPLANATION). *The* list explanation *for a decision* W *versus an alternative* W' *in MAP analysis is in the form* $(\mathbf{L}^+, \mathbf{L}^-)$*, where* $\mathbf{L}^+$ *is the list of positive evidence for* W *and* $\mathbf{L}^-$ *is the list of negative evidence for* W *(but positive for* W'*). Each evidence* $l \in \mathbf{L}^+ \cup \mathbf{L}^-$ *is associated with a score, denoted by* $s(l)(> 0)$*. A snapshot explanation for* W *in MAP contains a set of list explanations, one for each alternate choice* W'*.* □

Ideally, a list explanation should be *correct* and *complete*. A list explanation is correct if the sum of the scores of positive evidence is higher than that for negative evidence (so the *a posteriori* probability for W is higher than that for W'). A list explanation is complete if all evidence considered in the MAP analysis is included.

Table 4: List explanation for no-copying between S_1 and S_2 strictly following the MAP analysis.

	Score	Evidence
Pos	1.6	S_1 provides a different value from S_2 on Stonebraker
	1.6	S_1 provides a different value from S_2 on Carey
	1.0	S_1 uses a different format from S_2 although shares the same (true) value on Dewitt
	1.0	S_1 uses a different format from S_2 although shares the same (true) value on Bernstein
	1.0	S_1 uses a different format from S_2 although shares the same (true) value on Halevy
	.7	The *a priori* belief is that S_1 is more likely to be independent of S_2

Obviously, a complete list explanation must be correct as it strictly reflects the MAP analysis; however, as we show soon, such an explanation is often huge in size. In Sec.3.4 we show how we can relax the completeness requirement and shorten a list explanation to be *correct and comparable to the complete list explanation*.

EXAMPLE 3.2. *Table 3 shows the list explanation for "*S_1 does not copy from S_2*" versus "*S_1 copies from S_2*" in Ex.1.1. There are three pieces of positive evidence showing no-copying and one piece of negative evidence showing copying. The explanation is correct:* $3.2 + 3.06 + .7 = 6.96 > .06$. *The explanation is also complete, showing all evidence considered in the MAP analysis.* □

3.2 Generating list explanations

We next describe how we generate a list explanation strictly following the MAP analysis. We illustrate the main idea on no-copying and then generalize the algorithm.

Recall that between two sources there are three possible relationships: $S \perp S'$, $S \rightarrow S'$ and $S' \rightarrow S$. Thus, the snapshot explanation includes two list explanations. According to the MAP analysis (Eq.(1)), for $S \rightarrow S'$ we shall show $\beta P(\Phi|S \perp S') > \alpha P(\Phi|S \rightarrow S')$ and similar for $S' \rightarrow S$. As we assume independence of data items, we need to show the following (derived from Eq.(2-3)).

$$\Pi_{D \in \mathcal{D}} P_{ind}(\Phi_D(S)) > \Pi_{D \in \mathcal{D}} P(\Phi_D(S)|S \rightarrow S') \cdot \frac{\alpha}{1 - 2\alpha}. \quad (7)$$

Recall that we compare the *sum* of the scores for positive and negative evidence; we thus rewrite (7) as follows.

$$\sum_{D \in \mathcal{D}} \ln P_{ind}(\Phi_D(S)) > \sum_{D \in \mathcal{D}} \ln P(\Phi_D(S)|S \rightarrow S') + \ln \frac{\alpha}{1 - 2\alpha}. \quad (8)$$

Each data item D appears in the computation of both sides of the inequality. We decide if it supports $S \perp S'$ or $S \rightarrow S'$ by comparing $P_{ind}(\Phi_D(S))$ and $P(\Phi_D(S)|S \rightarrow S')$. If the former is larger, D is positive evidence for no-copying with score $\ln \frac{P_{ind}(\Phi_D(S))}{P(\Phi_D(S)|S \rightarrow S')}$; if the latter is larger, D is negative evidence with score $\ln \frac{P(\Phi_D(S)|S \rightarrow S')}{P_{ind}(\Phi_D(S))}$; otherwise, D is not evidence for either decision. By moving all D's that form positive evidence to the left side of the inequality and all D's that form negative evidence to the right side, we rewrite Eq.(8) as

$$\sum_{P_{ind}(\Phi_D(S)) > P(\Phi_D(S)|S \rightarrow S')} \ln \frac{P_{ind}(\Phi_D(S))}{P(\Phi_D(S)|S \rightarrow S')} > \sum_{P_{ind}(\Phi_D(S)) < P(\Phi_D(S)|S \rightarrow S')} \ln \frac{P(\Phi_D(S)|S \rightarrow S')}{P_{ind}(\Phi_D(S))} + \ln \frac{\alpha}{1 - 2\alpha} \quad (9)$$

Finally, the term $\ln \frac{\alpha}{1-2\alpha}$ represents the evidence coming from the *a priori* belief (α, β are not involved in any other part of Eq.(9)). This evidence is negative if $\alpha > 1 - 2\alpha$ ($\alpha > \frac{1}{3}$).

Obviously, the explanation is complete and correct: $P(S \perp S'|\Phi) > P(S \rightarrow S'|\Phi)$ if and only if the scores of positive evidence sum up to be higher than those of negative evidence.

EXAMPLE 3.3. *Consider explaining $S_1 \perp S_2$ in Ex.1.1. The list explanation w.r.t. $S_1 \rightarrow S_2$ is shown in Table 4.*

For item Stonebraker*, denoted by D_1, S_1 provides a different value from S_2. Recall from Ex.2.2 that $P_{ind}(\Phi_{D_1}(S_1)) = .97$ and $P(\Phi_{D_1}(S_1)|S_1 \rightarrow S_2) = .194$. Thus, D_1 serves as positive evidence for no-copying with score $\ln \frac{.97}{.194} = 1.6$. We compute the same score for item* Carey.

For item Dewitt*, denoted by D_2, S_1 provides the same value as S_2 but uses a different format. Recall that $P_{ind}(\Phi_{D_2}(S_1)) = .97$ and $P(\Phi_{D_2}(S_1)|S_1 \rightarrow S_2) = .354$. Thus, D_2 also serves as positive evidence for no-copying and the score is $\ln \frac{.97}{.354} = 1.0$. We compute the same score for items* Bernstein *and* Halevy.

Finally, the a priori *belief when $\alpha = .25$ serves as positive evidence with score $|\ln \frac{.25}{1-2*.25}| = .7$.*

In total, there are 6 pieces of positive evidence and no negative evidence. Note that by equation transformation and evidence extraction, the explanation is already much simpler than the description of MAP analysis in Ex.2.2. □

Generalization: We explain a general MAP decision W as follows (application on other fusion decisions shown in [10]).

1. List each alternate choice other than W.
2. Generate a list explanation for each choice W'.
   (a) Write and expand the inequality between inverse probability for W and that for W' to show that W has a higher *a posteriori* probability.
   (b) Take the logarithm of each side of the inequality.
   (c) For each involved element (*e.g.*, data item for copy detection), compare the probability computed on each side and decide if it serves as positive or negative evidence.
   (d) Add evidence according to *a priori* probabilities.

3.3 Categorizing and aggregating evidence

The current explanation scheme lists each data item as a piece of evidence. Since there can be a lot of data items in practice, the explanation can be long and overwhelming. We observe from Table 4 that a lot of evidence looks similar; a natural thought is to categorize and aggregate the evidence. We do so in two steps.

Evidence separation: Since our observation on a data item D consists of three *aspects*: existence of the item, provided value(s), and used format(s) (see Eq.(4)), we divide evidence on D into three, one for each aspect. This enables categorization on each aspect instead of on combinations of aspects.

Accordingly, we need to split the score on D for different aspects, denoted by $score_{ext}(D)$, $score_{val}(D)$, and $score_{fmt}(D)$. We compute (1) $sc_1 = score_{ext}(D)$, (2) $sc_2 = score_{ext}(D) + score_{val}(D)$, and (3) $sc_3 = score_{ext}(D) + score_{val}(D) + score_{fmt}(D)$ (sc_3 actually equals the overall score on D), and then infer $score_{ext}(D)$, $score_{val}(D)$, and $score_{fmt}(D)$. Consider the case of both sources providing the same value v as an example. We have

$$sc_1 = \ln \frac{P_{ind}(\Phi_D(S) \neq \emptyset)}{P(\Phi_D(S) \neq \emptyset | S \rightarrow S')} = \ln \frac{P_{ind}(\Phi_D(S) \neq \emptyset)}{s + (1-s)P_{ind}(\Phi_D(S) \neq \emptyset)} \quad (10)$$

$$sc_2 = \ln \frac{P_{ind}(\Phi_D(S) \neq \emptyset) P_{ind}(\Phi_{D.val}(S))}{P(\Phi_D(S) \neq \emptyset, \Phi_{D.val}(S) | S \rightarrow S')}$$

$$= \ln \frac{P_{ind}(\Phi_D(S) \neq \emptyset) P_{ind}(\Phi_{D.val}(S))}{s + (1-s)P_{ind}(\Phi_D(S) \neq \emptyset) P_{ind}(\Phi_{D.val}(S))}; \quad (11)$$

$$sc_3 = \ln \frac{P_{ind}(\Phi_D(S))}{P(\Phi_D(S) | S \rightarrow S')}. \quad (12)$$

A positive score shows that the specific aspect serves as positive evidence for no-copying and vice versa. Note that even if D as a whole serves as positive evidence, it is not necessary that

Table 5: Score and count for each category (combination of aspect and class) in Ex.3.4.

Aspect	Class 1	Class 2	Class 3	Class 4
Exist	0	0	0	0
Value	0	-.06, 3	0	3.2, 2
Format	0	0	0	3.06, 3

$score_{ext}(D)$, $score_{val}(D)$, and $score_{fmt}(D)$ are all positive. As shown in Ex.3.3, item Dewitt (D_2) serves as positive evidence. However, we compute $score_{ext}(D_2) = \ln \frac{1}{.8+.2*1} = 0$, $score_{val}(D_2) = \ln \frac{1*.97}{.8+.2*1*.97} - 0 = -.02$, $score_{fmt}(D_2) = 1 - (-.02) - 0 = 1.02$. Thus, providing D_2 is neither positive nor negative evidence, sharing the same value is *negative* evidence for no-copying, and using different formats is *positive* evidence. Exposing such hidden evidence is an extra benefit of evidence separation.

Classification: Now for each aspect we can classify the data according to the feature of the data and *why* it serves as positive or negative evidence. Take the value aspect as an example. There are four *classes*: (1) *sharing false value*; (2) *sharing true value*; (3) *providing a different value that is more likely to be provided if the source is a copier but provides this item independently* (*e.g.*, if S copies high-accuracy data from S' but all of its independently provided data are wrong, then for a value different from S''s, S has a probability of 1 to provide a wrong value conditioned on it being a copier of S' but a lower probability conditioned on being independent); and (4) *providing other different values*. Among these classes, the last forms positive evidence for no-copying and the others form negative evidence. The first two classes are essentially the same, but by separating them we can distinguish strong evidence and weak evidence. We have similar classes for the other aspects but in general the classes for different aspects can be different.

Finally, each class of an aspect forms a *category* of evidence; we can aggregate evidence in the same category and sum up the scores. Evidence collection and categorization can be done together in one scanning of the data items. We present details in [10] and illustrate the algorithm by an example.

EXAMPLE 3.4. *Continue with Ex.3.3. We first consider each data item, compute the scores and classify the reasons. Take D_2 as an example. Recall that we compute $score_{ext}(D_2) = 0$, $score_{val}(D_2) = -.02$ and $score_{fmt}(D_2) = 1.02$. For the value aspect, S_1 and S_2 provide the same true value, falling in Class 2; for the format aspect, S_1 and S_2 use different formats, falling in Class 4.*

We then aggregate evidence in the same category, resulting with 2 pieces of positive evidence (not including the a priori *belief evidence) and 1 piece of negative evidence (see Table 5). Table 3 gives the corresponding list explanation, containing only 4 instead of 6 pieces of evidence.* □

With evidence categorization and aggregation, the amount of evidence is not determined by the number of data items, but by the number of categories. In our experiments, the evidence lists can be shortened by orders of magnitude.

Generalization: We categorize and aggregate evidence for a general MAP decision as follows.

1. Manually enumerate the aspects according to the terms in probability computation (such as Eq.(4)).
2. Manually enumerate the classes of reasons why a particular aspect serves as positive or negative evidence given the feature of the observation.
3. For each involved element, split the scores for different aspects and classify the reason for each aspect.
4. Aggregate evidence in each category.

3.4 Shortening lists

Although evidence aggregation can significantly reduce the amount of evidence, the result lists can still contain twenty or thirty pieces of evidence, much of which may be "unimportant" and removable, as we show next.

EXAMPLE 3.5. *Consider the following list explanation (we show only scores).*

$$\mathbf{L}^+ = \{1000, 500, 60, 2, 1\}; \quad \mathbf{L}^- = \{950, 50, 5\}. \tag{13}$$

Obviously, removing the evidence whose scores are below 100 still shows that the positive evidence is much stronger than the negative evidence. However, if we further remove the negative evidence with score 950, it gives the wrong impression that there is no negative evidence. On the other hand, if we further remove the positive evidence with score 500, it gives the wrong impression that the positive evidence is only slightly stronger. □

We can certainly show only top-k evidence or evidence whose score is above a given threshold θ. However, using the same k and θ everywhere may cause over-shortening for some instances, where the explanation is incorrect, and under-shortening for some other instances, where further shortening will generate a shorter but still correct explanation. We next propose two better solutions that generate explanations being *correct and comparable to the complete explanation*. Both of them remove evidence from the end of the list, as typically the lower the score, the less important is the evidence; on the other hand, each follows a different principle of what is considered as *comparable* to the complete list explanation. Both methods can be applied for list explanation of *any* MAP decision.

Tail cutting First, given a shortened list explanation, we can guess the bound of the accumulated scores: the minimal accumulated score for all positive evidence happens when each removed positive evidence has score 0; the maximal score for negative evidence happens when each removed negative evidence has a score as high as the lowest remaining score for negative evidence. If even in such a worst case, the remaining positive evidence is still stronger, we consider the explanation as *comparable*. In Ex.3.5, if we remove the last negative evidence and inform in the explanation that "*there is 1 more piece of negative evidence with lower score*", then the removed score is at most 50, so the negative evidence (total score $950 + 50 + 50 = 1050$) is still weaker. We can further remove the last 3 pieces of positive evidence and the positive evidence is still stronger (total score $1000 + 500 = 1500 > 1050$). According to this intuition, we shall solve the following optimization problem.

DEFINITION 3.6 (TAIL-CUTTING PROBLEM). *Consider the following list explanation (we show only scores):*

$$\mathbf{L}^+ = \{x_1, x_2, \ldots, x_n\}; \quad \mathbf{L}^- = \{y_1, y_2, \ldots, y_m\}. \tag{14}$$

The tail-cutting problem *minimizes* $s + t, 1 \le s \le n, 1 \le t \le m$, *under the constraint*

$$\sum_{i=1}^{s} x_i > \sum_{j=1}^{t} y_j + y_t(m - t). \quad \square \tag{15}$$

In this definition, constraint (15) compares the minimum positive score, obtained when all removed scores are 0, with the maximum negative score, obtained when the $m-t$ pieces of removed evidence all have the maximum possible score y_t, and so guarantees that the shortened list is comparable to the complete explanation.

Algorithm CUTTAIL (pseudo-code in [10]) proceeds in three steps.

1. Iteratively try $s = n, n-1, \ldots, 1$, as far as $\sum_{i=1}^{s} x_s > \sum_{j=1}^{m} y_j$; the resulting s is the minimum when we do not remove any negative evidence ($t = m$) and serves as the starting point for the next step.
2. Iteratively try $t = m, m-1, \ldots,$ as far as $\sum_{i=1}^{n} x_i > \sum_{j=1}^{t} y_j + y_t(m - t)$ (at this point removing any more negative evidence cannot satisfy constraint (15), even if we add back all positive evidence). For each t, increase s when needed to guarantee (15), and record $s + t$.
3. Return the s and t with minimal $s + t$.

PROPOSITION 3.7. *Algorithm* CUTTAIL *solves the* TAIL-CUTTING *problem (finds the optimal solution) in time* $O(m + n)$. □

EXAMPLE 3.8. *Consider applying* CUTTAIL *to the full list explanation in Ex.3.5. The algorithm first removes the last 3 pieces of evidence from* $\mathbf{L}^+$*; at this point,* $1000 + 500 > 950 + 50 + 5$ *and* $s + t = 2 + 3 = 5$*. It then tries* $t = 2$*, where increasing* s *is not needed (*$1000 + 500 > 950 + 50 + 50 * 1$*); so* $s + t = 2 + 2 = 4$*. When it tries* $t = 1$*, even if all positive evidence is added back, we still have* $1000 + 500 + 60 + 2 + 1 < 950 + 950 * 2$*, so it stops. Finally, it returns* $s = 2, t = 2$ *as the result.* □

Difference keeping In our second method, we consider the shortened list as comparable to the complete one if it keeps the difference between the accumulated scores for the two evidence lists; this corresponds to dividing both the numerator and the denominator of Eq.(1) by a constant or dividing both sides of the inequation of Eq.(7) by a constant. In other words, we wish that the sum of the scores for removed positive evidence is nearly the same as that for removed negative evidence. Meanwhile, we wish to make the lists as short as possible, equivalent to making the sum of the removed scores as large as possible. Thus, we solve the following problem.

DEFINITION 3.9 (DIFFERENCE-KEEPING PROBLEM). *Consider the same list explanation as in Defn.3.6. Let* $X = \sum_{i=s+1}^{n} x_i$ *and* $Y = \sum_{j=t+1}^{m} y_j$ *(*$1 \le s \le n, 1 \le t \le m$*). The* difference-keeping problem *minimizes*

$$\frac{|X - Y| + \beta_1}{\max(X, Y) + \beta_2}, \tag{16}$$

where $0 < \beta_1, \beta_2 \le \min\{x_1, \ldots, x_n, y_1, \ldots, y_m\}$ *are small positive numbers, under the constraint*

$$\sum_{i=1}^{s} x_i > \sum_{j=1}^{t} y_j. \quad \square \tag{17}$$

In the objective function (16), we use β_1 and β_2 to guarantee non-zero numerator and denominator. Constraint (17) guarantees correctness of the explanation. Note that one may add other constraints such as giving upper bounds of X and Y to guarantee that at most a given fraction of evidence is removed; giving upper bounds of s and t to control maximum length of the lists; or use $\max(X, Y)^\gamma, \gamma > 0$, in (16) to control how much we wish to emphasize *small* list length.

A naive way of solving this problem tries each combination of m and n and can take time $O(mn(m + n))$. We now sketch an algorithm that solves the problem in only linear time. Recall that we remove evidence from the end of the lists, so we call a removed subset of evidence a *suffix sublist*. The key idea is that for each suffix sublist L, there is a *key evidence* $k(L)$ in the other list, such that the longest suffix without $k(L)$ has lower or the same accumulated score as L and the shortest suffix with $k(L)$ has higher accumulated score than L. Then, for each suffix, we only need to examine these two suffix sublists from the other list. Consider Ex.3.5 and the suffix list $\{5\}$ from $\mathbf{L}^-$. Its key evidence in $\mathbf{L}^+$ is the evidence with score 60. The longest suffix without this evidence, $\{2, 1\}$,

Table 6: Applying KEEPDIFF in Ex.3.5.

Rnd	Remove from $\mathbf{L}^+$	Remove from $\mathbf{L}^-$	Difference	Objective
0	$\emptyset$	$\emptyset$	0	$\frac{1}{1} = 1$
1	$\{1\}$	$\emptyset$	1	$\frac{2}{2} = 1$
2	$\{1, 2\}$	$\emptyset$	3	$\frac{4}{4} = 1$
3	$\{1, 2\}$	$\{5\}$	2	$\frac{3}{6} = .5$
4	$\{1, 2\}$	$\{5, 50\}$	52	$\frac{53}{56} = .95$
5	$\{1, 2, 60\}$	$\{5, 50\}$	8	$\frac{9}{64} = .14$

has a lower score than 5, and the shortest suffix with the evidence, $\{60, 2, 1\}$, has a higher score. Obviously, any other suffix sublist in $\mathbf{L}^+$ has a higher difference from $\{5\}$ than these two.

According to this intuition, Algorithm KEEPDIFF (pseudo-code in [10]) scans $\mathbf{L}^+$ and $\mathbf{L}^-$ bottom-up as follows.

1. Remove evidence with the lowest score.
2. Check constraint (17), compute the objective function, and record the solution if its value is lower than the recorded lowest value.
3. Decide the evidence to remove at the next round as follows: (1) find the next suffix (the current suffix plus the next evidence) of the other list; (2) find its key evidence in the current list; and (3) pick evidence from the current list until reaching the key evidence.
4. Repeat Steps 2-3 until reaching the first evidence of a list.

PROPOSITION 3.10. *Algorithm* KEEPDIFF *solves the* DIFFERENCE-KEEPING *problem in time* $O(m + n)$. □

EXAMPLE 3.11. *Continue with Ex.3.5. Table 6 shows removed evidence and the value of the objective function in each round; here, we set* $\beta_1 = \beta_2 = 1$. *We start with* $\mathbf{L}^+$, *as it contains the lowest score. Initially, the next suffix sublist in* $\mathbf{L}^-$ *is* $\{5\}$, *and its key evidence in* $\mathbf{L}^+$ *has score 60; thus, we pick scores 1 and 2 first and then switch to list* $\mathbf{L}^-$. *We continue till reaching the first element of* $\mathbf{L}^-$. *The result of Round 5 is optimal, even though its difference is not the smallest.* □

In practice, we apply both CUTTAIL and KEEPDIFF and choose the solution with shorter lists (*i.e.*, minimal $s+t$). Our experiments show that these strategies can further shorten the lists by half.

EXAMPLE 3.12. *Continue with explaining no-copying between* S_1 *and* S_2 *for the running example. For evidence in Table 3,* CUTTAIL *would remove the last two pieces of positive evidence, while* KEEPDIFF *would not remove any evidence. We thus choose the results of* CUTTAIL. *The final explanation can go like this.* "There are 3 pieces of positive evidence for no-copying, where the strongest is that S_1 provides 2 different values from S_2 (with score 3.2). There is 1 piece of negative evidence for no-copying: S_1 provides the same *true* value on 3 data items as S_2 (with score .06). The positive evidence is stronger so no-copying is likely." □

4. EXPLAINING THE EXPLANATION

We next consider generating comprehensive explanations, where we take only provided data as evidence. Again, we start with full explanation (Sec.4.1), and then describe how we shorten the explanation efficiently (Sec.4.2). The techniques we present in this section can apply to any type of iterative decisions.

4.1 DAG explanation

A comprehensive explanation needs to in addition explain every "evidence" inferred over the data. We can adopt the DAG structure, where each node explains a decision, and the children are the evidence (note that the final explanation may be in a different presentation, which is outside the scope of the paper).

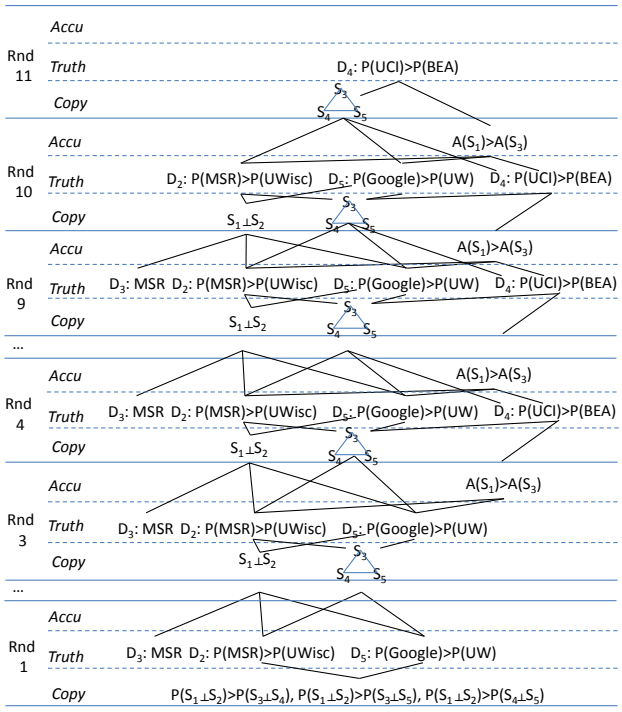


Figure 1: Full explanation DAG for the decision "UCI *is more likely than* BEA *to be the affiliation of* Carey" (represented by $D_4 : P(UCI) > P(BEA)$). The triangle between S_3, S_4 and S_5 represents copying between them; $A(S_1) > A(S_3)$ represents that S_1 has a higher accuracy than S_3; $D_3 : MSR$ represents that MSR is the correct affiliation for Bernstein.

DEFINITION 4.1 (DAG EXPLANATION). *The* DAG explanation *for an iterative MAP decision W is a DAG* $(\mathbf{N}, \mathbf{E}, R)$, *where (1) each node in* $\mathbf{N}$ *represents a decision and its list explanations, (2) each edge in* $\mathbf{E}$ *indicates that the decision of the child node is evidence for that of the parent node, and (3) there is a single node R that has no parent and represents the decision W.* □

Similar to snapshot explanations, an ideal DAG explanation should also be *correct* and *complete*. It is correct if (1) the explanation represented by each node is correct, and (2) each child supports its parents as positive evidence. It is complete if for every node, each positive evidence inferred from the data corresponds to a child node. Note that we do not expand the DAG for negative evidence, since their opposites will only further strengthen our decision.

Consider explaining "UCI *is more likely than* BEA *to be the correct affiliation of* Carey" in the motivating example. As we have shown in Ex.1.2, careless generation of the DAG can result in loops. Similar to explaining "WHY" according to provenance [3], we explain by tracing the decisions from the last round of iterations back to the first round. In particular, we start with generating the root node for the decision at the convergence round and its children for the supporting evidence at the same or the previous round. We repeat until all leaf nodes can be inferred directly from the data. We call the result a *full explanation DAG*. Obviously, a full explanation DAG is both correct and complete.

EXAMPLE 4.2. *Fig.1 shows the full explanation DAG for our example. The root node has two children, showing that we make this decision at the convergence round, the 11th round, because we detect copying between* $S_3 - S_5$ *at the 11th round, and compute a higher accuracy for* S_1 *than* S_3 *at the 10th round. We make both*

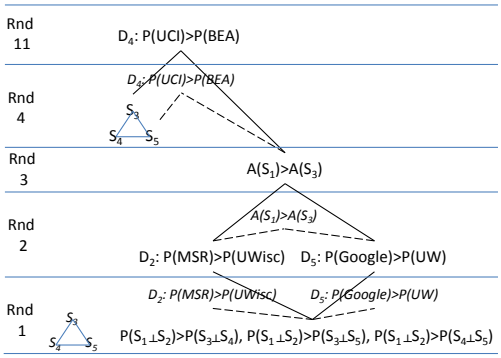


Figure 2: Critical-round DAG. Not being part of the DAG, italic-font nodes indicate the critical round for a decision and dashed lines show the reasons at the critical round.

of these two decisions based on our decisions at the 10th round that UWisc, UW *and* BEA *are wrong. Among them, the decision on* BEA *at Round 10 is made for the same two reasons as at Round 11; the decisions on* UWisc *and* UW*, on the other hand, are made because of copying between $S_3 - S_5$ and no-copying between $S_1 - S_2$ (the reasoning is that these two values are provided by three sources with copying and the correct values are provided by two independent sources), both decided at the 10th round. While copying between $S_3 - S_5$ is detected for the same reasons as at Round 11, no-copying between $S_1 - S_2$ is decided based on the decisions at the 9th round that the shared values* MSR, MSR, *and* Google *(for D_2, D_3, D_5 respectively) are all correct. We decide that* MSR *is correct for D_3 purely from the data, because no other value is provided on D_3, so the node is a leaf node. We further expand the DAG for other decisions.*

At the 4th round, we decide copying between $S_3 - S_5$ only because we decided at Round 3 that UWisc *and* UW *are wrong, which again are decided because of copying between $S_3 - S_5$ and no-copying between $S_1 - S_2$. When we trace back to the 1st round, we show that we made decisions on* UWisc *and* UW *because the no-copying probability between $S_1 - S_2$ is higher than that between $S_3 - S_5$, which in turn is inferred from the raw data because S_1 and S_2 share fewer values (initially we assume the same probability for each value to be true). We can thus terminate.* □

4.2 Shortening DAG explanations

A full explanation DAG is often huge because some parts can be repeated many times; for example, in Fig.1 the subgraphs for Round 4 to 9 are exactly the same. We wish to reduce the size of the DAG by removing the repeated subgraphs. We observe that if the same decision is made at two consecutive rounds, their supporting positive evidence are typically the same. The only difference is the exact scores, which may change slightly between rounds, but such small changes are not significant in understanding the decision. We thus shorten the explanation by explaining a decision only at its *critical round*, the last round when we change our decision; in other words, we explain how we initially make this decision. Such a DAG is called a *critical-round DAG*.

EXAMPLE 4.3. *Continue with Ex.4.2. Fig.2 shows the critical-round DAG for our example. It shows that the decision is first made at Round 4 (before that we wrongly consider* BEA *as correct) based on (1) copying between $S_3 - S_5$, decided at Round 4, and (2) that the accuracy of S_1 is higher than S_3, decided at Round 3. Copying between $S_3 - S_5$ is originally decided at Round 1 according to the high overlap between these sources; it does not have any child because it is purely inferred from the data. The decision that S_1 has a higher accuracy is originally made at Round 2 (although in that round the difference is not significant for believing that* UCI *is correct), based on the decisions at Round 2 that* UWisc *and* UW *are false. These two decisions are originally made at Round 1, again based on the decisions at Round 1 that the no-copying probability between $S_1 - S_2$ is higher than that between $S_3 - S_5$, inferred from the raw data. Critical-round DAGs can be significantly smaller; the example DAG includes only 6 nodes.* □

Critical-round DAGs are both correct and complete if we consider only the critical rounds. Our experiments show that there are typically very few extra reasons appearing after the critical round and even fewer reasons disappearing after the critical round.

We next formally define the critical-round DAG, which we propose to use as the comprehensive explanation.

DEFINITION 4.4 (CRITICAL-ROUND EXPLANATION DAG). *Let W be a decision at Round n. The* critical round *of W, denoted by $r(W, n)$, satisfies the following conditions: (1) W is made in Round $1 \leq r(W, n) \leq n$, (2) $r(W, n) = 1$, or $\neg W$ is made in Round $r(W, n) - 1$.*

An explanation DAG is called a critical-round DAG *if for each node N and its decision W at Round n, N's children represent positive evidence for W at round $r(W, n)$.* □

Obviously, constructing a critical-round DAG would require recording the decisions we make in each round in a *log file*. There can be many rounds before convergence, so constructing a DAG would very often require importing the decisions and restoring the status for different rounds back and forth from the logs. By pre-generating the explanation and the evidence list for each decision offline, and storing them in a *database* (details in [10]), we can speed up DAG construction significantly.

5. EXPERIMENTAL RESULTS

We now describe experimental results on real-world data showing that (1) the list of evidence we generate for the explanations are correct (2) our techniques can significantly reduce the amount of evidence; and (3) we can generate the explanations efficiently.

5.1 Experimental setting

We experimented on the AbeBooks data set, extracted in 2007 from *AbeBooks.com* by searching computer-science books[4]. There are 894 bookstores (data sources), 1265 books, and 24364 listings, each containing attributes ISBN, name, and often authors[5].

We generate explanations for four types of decisions: I. *truth discovery:* true value for the name and author list of each book; II. *copy detection*: copying or no-copying between sources whose Jaccard similarity on data items (intersection over union) is at least .1 (there are 3210 such pairs); III. *copy direction:* direction of copying between sources with detected copying (there are 1552 pairs where the direction can be decided); IV. *copy pattern:* copying by object or copying by attribute [6] between sources with detected copying (there are 1340 detected patterns). We consider all for snapshot explanations, and I and II (only copying) for comprehensive explanations.

For snapshot explanation, we compare six list-shortening strategies: (1) TOPK: showing evidence with the top-k scores; (2) LARGE:

[4]We thank authors of [24] for providing us the data, which can be found at *http://lunadong.com/fusionDataSet.htm*.

[5]Previous study shows that a naive voting for deciding the correct list of authors on this data set obtains an accuracy of only .71, while the advanced fusion technique in [8] obtains an accuracy of .89.

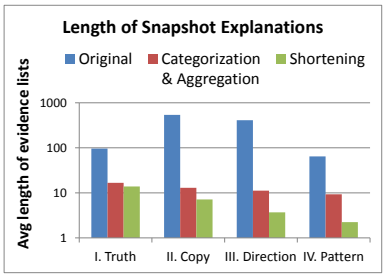


Figure 3: Length of explanations.

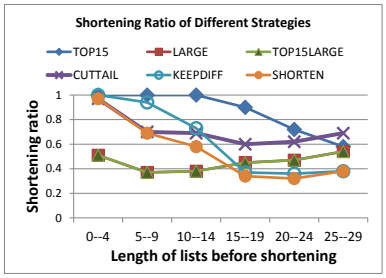


Figure 4: Shortening ratio.

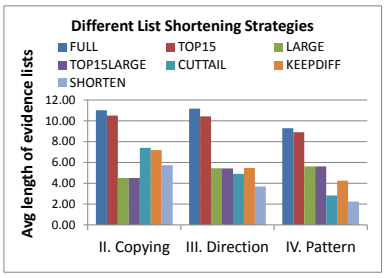


Figure 5: Shortening strategies.

showing evidence whose score is larger than 5; (3) TOPKLARGE: showing the top-k evidence whose score is larger than 5; (4) CUTTAIL: applying Algorithm CUTTAIL; (5) KEEPDIFF: applying Algorithm KEEPDIFF with $\beta_1 = \beta_2 = .01$; and (6) SHORTEN: trying both CUTTAIL and KEEPDIFF, and selecting the results with shorter lists. By default, we apply SHORTEN.

For comprehensive explanation, we compare full explanation DAG and critical-round DAG. We also generated *enriched critical-round DAG*, where appearing and disappearing evidence after the critical round is also expanded. By default, we used critical-round DAGs.

We used Java and experimented on a WindowsXP machine with 2.66GHz Intel CPU and 3.48GB of RAM. We hosted the database using MySQL.

5.2 Snapshot explanations

Shortening strategies: Fig.3 shows results of generated snapshot explanations for the four types of decisions. We have five observations. (1) Evidence categorization and aggregation shortens the evidence list by an order of magnitude on average. (2) List shortening further shortens the evidence list by 51% on average. (3) Evidence categorization and aggregation can reduce the size of explanations more for copy detection than for truth discovery, because the amount of raw evidence for the former, decided by the number of values provided by the sources, is much larger than that for the latter, decided by the number of sources providing the data item. (4) The final amount of evidence is the largest for decisions of Type I as each explanation involves multiple list explanations, then for those of Type II as each explanation involves two list explanations, and last for Type III and IV as each involves a single list explanation. (5) All evidence lists are correct.

As a case study, we observed that the largest explanation without shortening is for a Type II decision. The original explanation contains two lists, in total containing 4927 pieces of evidence. One can imagine how verbose the explanation could be if we give a detailed description of the Bayesian analysis. After categorization and aggregation, there are still 29 pieces of evidence in total. After list shortening the number further drops to 15.

We next compare different list shortening strategies. We first consider decisions of copy detection (Type II). Table 7 shows the average length of the result lists and Fig. 4 shows the *shortening ratio* (percentage of the size of the shortened lists over that of the full lists) for each method. We have four observations. (1) LARGE and TOP15LARGE obtain the shortest evidence lists; however, this is at the price of introducing errors (the sum of scores for positive evidence is no larger than that for negative evidence) in the explanations as they remove evidence without checking. As shown in Table 8, TOPKLARGE introduces errors for 47-48 (2.2%) pairs of sources; TOPK in itself introduces only a few errors, but on the other hand, in reduces the list length only slightly. (2) SHORTEN obtains slightly longer evidence lists than LARGE and TOP15LARGE, but does not introduce any error. (3) CUTTAIL and KEEPDIFF obtain similar results in terms of the average length of the result lists;

Table 7: Average number of evidence in the explanations generated by each shortening strategy.

FULL	TOP15	LARGE	TOP15LARGE	CUTTAIL	KEEPDIFF	SHORTEN
11.0	10.5	4.5	4.5	7.4	7.2	5.7

Table 8: Errors in snapshot explanations.

	ALL	TOP20	TOP15	TOP10
TOPK	0	0	0	1
TOPKLARGE	47	47	47	48

however, the former is better at shortening short lists (5-14 evidence) and the latter is better at shortening long lists (15-29 evidence). SHORTEN combines them and obtains shorter lists. (4) Finally, most methods have a lower shortening ratio for longer lists, whereas LARGE and TOPKLARGE have consistent ratio for lists of various length, and are able to significantly shorten very short lists (0-4 evidence), but this again is at the price of making errors.

We next consider decisions of Type III and IV; Fig.5 shows the length of evidence lists generated by different shortening strategies. The results are in general consistent with our observations for decisions of Type II and we have the following additional observations. First, before shortening, the list explanations for decisions of Type IV are short; since CUTTAIL is better at shortening short lists (see Fig.4), the results of SHORTEN are affected more by CUTTAIL. Second, for decisions of Type III and IV, each evidence typically has a high score, so LARGE and TOP15LARGE under-shorten and generate longer lists than SHORTEN.

Efficiency: Table 9 shows efficiency of generating explanations for each type of decisions. We observe that (1) explanations can be generated very quickly online, and (2) the list shortening strategies introduce a very small overhead. Note that collecting evidence for decisions of Type II-IV all requires scanning provided data and took 62.3 ms on average. Note also that collecting evidence for truth discovery decisions requires computing copying probability for each shared value and thus took longer time.

5.3 Comprehensive explanations

Shortening strategies: The iterative Bayesian analysis on the experimental data set took 9 rounds. Fig.6 plots the size of the critical-round DAGs versus the critical round. We observe that for truth discovery decisions, those that do not change since the first round typically have a small DAG (with less than 15 nodes), whereas those changed at later rounds can have much larger DAGs (the largest DAG has 1035 nodes). In contrast, for copy detection decisions, the DAGs for decisions not changed since the first round have only 1 node (copy detection in the first round is based purely on provided data). Despite the fact that copying decisions typically require more inferred evidence than truth discovery decisions, the former typically have smaller DAGs than the latter; this is because a DAG for a copying decision often has only one node (the root) representing a copying decision, but a DAG for a truth discovery decision can often have several such nodes.

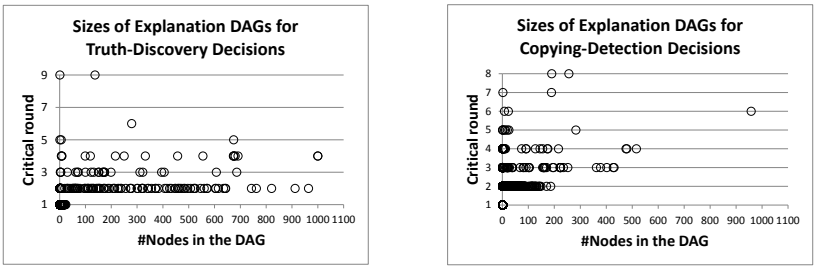


Figure 6: Size distribution of critical-round DAGs.

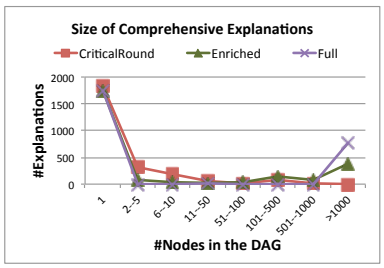


Figure 7: Size of comprehensive explanations.

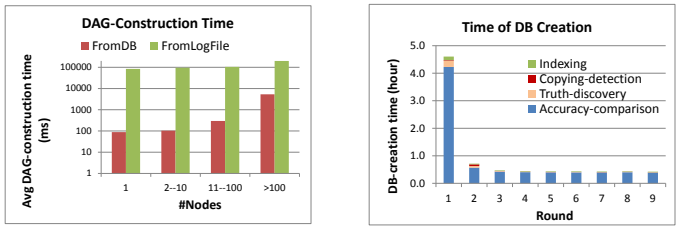
Table 9: Runtime of explanation generation.

(In ms)	I. Truth	II. Cpy.	III. Dir.	IV. Pat.
Evid Collection	350.3	62.3		
Categorization	.08	31.2	8.8	.03
Shortening	.12	.01	.02	.01

Fig.7(a) compares the sizes of different types of DAGs for truth discovery decisions. We have three observations. (1) Most full DAGs either have only 1 node (69%), or have over 1000 nodes (30%), meaning that once a decision is not purely supported by provided data, the full explanation DAG is typically huge. (2) Most critical-round DAGs are small as they show only evidence at the critical rounds: 72.4% of the DAGs have 1 node, 92.6% have less than 10 nodes, and only 1 has more than 1000 nodes. (3) Finally, enriched critical-round DAGs can be much larger than critical-round DAGs. We observe that on average there are .75 appearing evidence for decisions not changed since the first round, and nearly 0 appearing evidence for other decisions, and nearly 0 disappearing evidence for all decisions. However, explaining such additional evidence at a late round can significantly increase the size of the DAGs: 25.9% of the DAGs are of size larger than 10 and 15% have more than 1000 nodes.

Efficiency: Fig.8(a) compares the efficiency of generating comprehensive explanations from the database and directly from the log files. Constructing explanation DAGs from a database was very efficient: on average it took only 0.3 second and in the worst case it took 22 seconds. DAG construction from files on average took 283.5 times as long as that from a database. For DAGs with up to 10 nodes, using the database reduced runtime by 3 orders of magnitude; even for DAGs of size over 100, using the database reduced runtime by more than 1 order of magnitude.

Fig.8(b) reports database creation time. We finished creating the database in 8.4 hours and the size of the database is 766MB. It is acceptable given that it is an offline process. We observe that populating tables for the first round took the longest time (4.6 hours), because most decisions are made at that round; starting from the third round, each round took less than half an hour. We also observe that generating explanation and evidence for accuracy comparison decisions took much longer time than other types of decisions, because there are many more such decisions in each round. Finally, it took 55 hours to create a database for constructing full explanation DAGs (13.7GB), as we need to generate explanation and evidence for each decision at each round; this further shows the huge overhead for generating full explanation DAGs.



(a) DAG construction (b) Database creation

Figure 8: Efficiency of generating comprehensive explanations.

6. RELATED WORK

Generating *provenance* (or *lineage*) information to facilitate understanding of data management and data integration results has received recent interest in the database community. Techniques have been proposed for explaining results for queries [3, 4, 14, 19], workflows [5], schema mappings [13], and information extraction [15, 22]. We propose explaining data fusion results [1, 6, 8, 12, 20, 21, 24, 25, 26, 27]; the core ideas, including how to explain iterative MAP analysis and how to efficiently shorten such explanations, are not discussed in any existing work on data fusion. The following characteristics of our techniques distinguish our work from previous explanation works.

First, we need to explain results from MAP analysis which considers alternate decisions and reasons about the probability of each of them. Causality reasoning [19] does not easily apply for such analysis; we proposed list explanation according to the nature of MAP analysis. One of our key contributions is evidence-list shortening. [14, 15] discussed reducing the number of returned reasons by applying constraints and declaring trust on certain data. These techniques do not apply in our context; we instead consider evidence categorization, aggregation and list shortening.

Second, we need to explain results from iterative reasoning. Among existing work, only [22] explains iterative reasoning: it proposed querying all extraction patterns that contribute to an extracted tuple and all tuples that are influenced by an extraction pattern over all iterations. By creating an explanation database, we can support such queries in the context of data fusion as well, and we in addition generate the evidence DAG for comprehensive explanation. Finally, answers to provenance queries are also in the DAG structure and indexing techniques have been proposed for accelerating query evaluation [16]. Our techniques differ in that we leverage the repetition in the iterations to reduce the size of the explanation DAG and use a database to accelerate DAG construction.

7. CONCLUSIONS

In this paper we study explaining data fusion results obtained by iterative MAP analysis. We proposed snapshot explanations and comprehensive explanations, and showed how we efficiently generate such explanations and significantly reduce the size of the explanations. Future work includes applying our ideas in pinpointing important decisions, and improving data fusion results by seeking user feedback.

8. REFERENCES

[1] L. Blanco, V. Crescenzi, P. Merialdo, and P. Papotti. Probabilistic models to reconcile complex data from inaccurate data sources. In *CAiSE*, 2010.

[2] J. Bleiholder and F. Naumann. Data fusion. *ACM Computing Surveys*, 41(1):1–41, 2008.

[3] P. Buneman, J. Cheney, W.-C. Tan, and S. Vansummeren. Curated databases. In *Proc. of PODS*, 2008.

[4] A. Chapman and H. Jagadish. Why not? In *Sigmod*, 2009.

[5] S. Davidson and J. Freire. Provenance and scientific workflows: Challenges and opportunites. In *Proc. of SIGMOD*, 2008.

[6] X. L. Dong, L. Berti-Equille, Y. Hu, and D. Srivastava. Global detection of complex copying relationships between sources. *PVLDB*, 2010.

[7] X. L. Dong, L. Berti-Equille, Y. Hu, and D. Srivastava. Solomon: Seeking the truth via copying detection. *PVLDB*, 2010.

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

[9] X. L. Dong, L. Berti-Equille, and D. Srivastava. Truth discovery and copying detection in a dynamic world. *PVLDB*, 2(1), 2009.

[10] X. L. Dong and D. Srivastava. Compact explanation of data fusion decisions. http://lunadong.com/publication/explanation_report.pdf.

[11] M. J. Druzdzel. Qualitative verbal explanations in bayesian belief networks. *Artificial Intelligence and Simulation of Behaviour Quarterly*, 94:43–54, 1996.

[12] A. Galland, S. Abiteboul, A. Marian, and P. Senellart. Corroborating information from disagreeing views. In *WSDM*, 2010.

[13] B. Glavic, G. Alonso, R. J. Miller, and L. M. Haas. TRAMP: Understanding the behavior of schema mappings through provenance. *PVLDB*, 3(1), 2010.

[14] M. Herschel and M. A. Hernandez. Explaining missing answers to SPJUA queries. *PVLDB*, 3(1), 2010.

[15] J. Huang, T. Chen, A. Doan, and J. F. Naughton. On the provenance of non-answers to queries over extracted data. *PVLDB*, 1(1), 2008.

[16] A. Kementsietsidis and M. Wang. Provenance query evaluation: what's so special about it? In *CIKM*, 2009.

[17] C. Lacave, R. Atienza, and F. J. Diez. Graphical explanation in bayesian networks. *Lecture Notes in Computer Science*, 1933:122–129, 2000.

[18] X. Li, X. L. Dong, K. B. Lyons, W. Meng, and D. Srivastava. Truth finding on the deep web: Is the problem solved? *PVLDB*, 6(2), 2013.

[19] A. Meliou, W. Gatterbauer, K. F. Moore, and D. Suciu. The complexity of causality and responsibility for query answers and nonanswers. *PVLDB*, 4(1), 2010.

[20] J. Pasternack and D. Roth. Knowing what to believe (when you already know something). In *COLING*, pages 877–885, 2010.

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

[22] A. D. Sarma, A. Jain, and D. Srivastava. I4e: interactive investigation of iterative information extraction. In *Sigmod*, 2010.

[23] M. Wu and A. Marian. A framework for corroborating answers from multiple web sources. *Inf. Syst.*, 36(2):431–449, 2011.

[24] X. Yin, J. Han, and P. S. Yu. Truth discovery with multiple conflicting information providers on the web. *IEEE Trans. Knowl. Data Eng.*, 20:796–808, 2008.

[25] X. Yin and W. Tan. Semi-supervised truth discovery. In *WWW*, pages 217–226, 2011.

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

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