Data Conflict Resolution with Markov Logic Networks

Li Qing-zhong, Zhang Yong-xin, Cui Li-zhen

School of Computer Science and Technology, Shandong University, Jinan 250101, China

ABSTRACT

In data integration, data conflict resolution is the key issue which is closely related to the quality of integrated data. Interest in it has grown rapidly in recent years, and many approaches have been proposed. However, because they lack of considering all-round influence factors of data conflict, the resolution results of them are often inaccurate. On the other hand, when a new factor is considered, the existing methods need to redefine the conflict resolution functions or re-modeling and are not adaptive. In this paper, we propose a novel data conflict resolution approach based on Markov Logic Networks (MLNs). Our method synthetically utilizes multi-angle features and knowledge to improve the accuracy of data conflict resolution. Based on MLN, the features and knowledge we proposed can be formulated and seamlessly combined. Experiments on two real world datasets show the effectiveness of our approach.

Keywords: Data Conflict Resolution; Markov Logic Networks; Data integration

1. INTRODUCTION

Data integration is the process of providing users of an integrated information system with a unified view of several data sources. However, due to data quality discrepancy of data sources, different sources can often provide conflicting data; some can reflect real world while some cannot. To provide high-quality data to user, it is essential for data integration system to resolve data conflicts and discover the true values from false ones. This process is called data conflict resolution and has recently received increasing attention in data integration field [1, 2].

The current major works to resolve data conflicts are based on relational algebra and define some conflict resolution strategies and functions [3]. By relational operations expansion or user-defined-functions, user or domain expert can assign conflict resolution functions to different data conflicts according to their requirements or domain knowledge [4]. Though these methods can resolve data conflict to some extent, they fall short in the following aspects.

1) The assignments of conflict resolution functions depend on the knowledge of user or domain expert, it is labor-intensive and time-consuming.
2) When new data and data sources are integrated into system, the previous assignment may be refined. Even a new conflict resolution function will be assigned or defined. So these methods can hardly adapt the situation where data integration is dynamic.

3) Among all conflict resolution strategies, “Trust your friends” and “Cry with wolves” [3] are widely used. Their principles are taking the value of a preferred source and taking the most frequent value. However, it is a challenge for data integration how to choose the most trustworthy data source. And it is arbitrary to only trust a certain source. In addition, especially on Web, with the ease of publishing and spreading information, the false information becomes universal. The voting strategy that prefers the most frequent is not sufficiently reasonable. So the current methods can hardly guarantee the precision of data conflict resolution.

Recently, there has been a few interesting techniques developed that aim to identify the true values from false ones [5, 6, 7]. They can be called truth discovery or others. These approaches treat data conflict resolution as an inferring problem, and integrate more semantic features and sophisticated human knowledge to determine which value is true. In the process of handling data conflicts, any helpful confidences and rules can be considered. However, as the uncertainty of the knowledge, it is a hot potato how to combine these evidences to infer the true values.

In this paper, we propose an approach for data conflict resolution based on Markov Logic Networks (MLNs) [8]. To improve the accuracy of data conflict resolution, we incorporate multi-angle features and knowledge. Moreover, we leverage the power of Markov Logic Networks to combine these evidences for inference. By integrating all these evidences and learning the corresponding weight, MLNs can handle uncertainty and even tolerant imperfect and contradictory knowledge. The experimental results on two real world datasets show an encouraging performance, and also demonstrate the adaptability of our method.

This paper is organized as follows. We briefly review some related research efforts in Section 2, and describe the problem in Section 3. The overview of the proposed approach is introduced in Section 4, and the model details are described in Section 5. Experimental evaluations are reported in Section 6, and in the last section we draw conclusions and point out some future directions.

2. RELATED WORD

The current major works to resolve data conflict on query time are based on relational algebra. The most representative work is conducted by Felix Naumann et al. Naumann et al. summarize current conflict resolution strategies and functions, and propose two research prototypes: HumMer [9] and FeSum [10]. They also extend and implement some relational operators such as minimum union [11].
Besides resolving data conflicts by relation expansion, there are some researches which focus on identifying true value from conflicting data. Minji Wu et al. [5] propose aggregating query results from general search engine by considering importance and similarity of the sources. The importance of the sources can be measured by their ranks and popularity. However, the rank of web pages according to authority based on hyperlinks does not reflect accuracy of information exactly. In addition, the method has certain limitation because it can only focus on queries whose answers are numerical values.

For discovering the true fact from conflict information provided by multiple data sources, Xiaoxin Yin et al. [6] propose an iterative algorithm - TruthFinder, which considers trustworthy of sources, accuracy of facts and interrelationship of two aspects. Nevertheless, this method does not consider dependence between sources in truth discovery. With the ease of publishing and spreading false information on the Web, a false value can be spread through copying and that makes truth discovery extremely tricky.

Xin Dong et al. [7] propose a novel approach that considers dependence between data sources in truth discovery. And they apply Bayesian analysis to decide dependence between sources and design an algorithm that iteratively detects dependence and discovers truth from conflicting information. However, Bayesian model will be re-trained when some new inference rules join. So the approach is not adaptive enough.

Markov logic networks [8] is a simple approach to combining first-order logic and probabilistic graphical models in a single representation. As a general probabilistic model for modeling relational data, MLNs have been applied to joint inference under different domains, such as entity resolution [12] and information extraction [13]. We'll give a more detailed introduction to MLNs in Section 5.

3. PROBLEM DEFINITION

To make a clear presentation and facilitate the following discussions, we first explain some concepts in this paper in this section.

**Entity.** An entity is a real world thing which is recognized as being capable of an independent existence and which can be uniquely identified, such as a book, a movie, etc.

**Entity Attribute.** Obviously, an entity attribute represents a particular aspect of a real world entity, such as an author of a book, a director of a movie.

**Fact.** For an entity attribute, the value provided by a data source can be called fact.

**Data Conflict.** When some data sources provide different facts for the same entity attribute, data conflict will occur.

**True Value.** In the conflicting facts, the fact which describes the real world is the true value.

Different data sources can provide different facts for some entity attributes. Among facts provided for an entity attribute, one correctly describes the real world and is the true value, and the rest are false. Figure 1 depicts the sources, facts, entities, entity attributes and the relationships between them.

4. APPROACH OVERVIEW

The flowchart of our method is illustrated in Fig. 2, which mainly consists of three steps: (a) feature extraction; (b) rules setting; and (c) joint inference for the true values from conflicting facts.

The input of our method is the data set in which the duplicated records are detected. The output is the data set in which the data conflicts are resolved. The first step is in charge of feature extraction. There are four kinds of features: basic features, the features of inter-dependency between sources and facts, the features of mutual implication between facts and the features of mutual dependency between sources. In the second step, we introduce some rules to infer the true values, which are represented as formulas in MLN. Then in the third step, we will train our MLN model by training set to automatically learn the weights of each formula and infer the true values in test set. Finally, we can get accurate and consistent data set according to the inference results.

5. MODULAR DETAIL

5.1 Markov Logic Networks

Markov Logic Networks (MLNs) [8] is a simple approach to combining first-order logic and probabilistic graphical models in a single representation, and is a probabilistic extension of a first-order logic for modeling
relation data. MLNs soften the constraints of a first order logic. That is, when a world violates one formula it is less probable, but not impossible. Thus, for the problem of data conflict resolution, MLNs is a sounder model since the real world is full of uncertainty, noise imperfect and contradictory knowledge.

**DEFINITION 1.** A Markov logic network L is a set of pairs \( \{(F_i, w_i)\}_{i=1}^{n} \), where \( F_i \) is a formula in first logic and the real number \( w_i \) is the weight of the formula. Together with a MLN \( L \) and a finite set of constants \( C = \{C_1, C_2, \ldots, C_{\ell}\} \), it constructs a Markov Random Field \( M_{L,C} \) as follows:

1. \( M_{L,C} \) contains one binary node for each possible grounding of each predicate appearing in \( L \). The value of the node is 1 if the ground atom is true and 0 otherwise.
2. \( M_{L,C} \) contains one feature for each possible grounding of each formula \( F_i \) in \( L \). The value of this feature is 1 if the ground formula is true and 0 otherwise. The weight of the feature is the \( w_i \) associated with \( F_i \) in \( L \).

Thus, MLN can be viewed as a template for constructing Markov Random Fields \[18\]. The probability of a state \( x \) in a MLN can be given by:

\[
P(X=x) = \frac{1}{Z} \exp \left( \sum_{i=1}^{n} w_i n_i(x) \right) = \frac{1}{Z} \prod \phi(x_{[i]})^{1(1)}
\]

where \( Z \) is a normalization factor employed for scaling values of \( P(X=x) \) to [0,1] interval, \( n_i(x) \) is the number of true groundings of \( F_i \) in \( x \), \( x_{[i]} \) is the state of the atoms appearing in \( F_i \), and \( \phi(x_{[i]}) = e^{w_i} \), \( w_i \) is the weight of the \( i^{th} \) formula.

Eq. (1) defines a generative MLN model. It defines the joint probability of all the predicates. In our application of data conflict resolution, we know the evidence predicates and the query predicates a prior. Thus, we turn to the discriminative MLN. Discriminative models have the great advantage of incorporating arbitrary useful features and have shown great promise as compared to generative models \[8,14\]. We partition the predicates into two sets - the evidence predicates \( X \) and the query predicates \( Q \). Given an instance \( x \), the discriminative MLN defines a conditional distribution as follows:

\[
P(q|x) = \frac{1}{Z_q(x)} \exp \left( \sum_{e \in E} \sum_{r \in R} w_{e\rightarrow r}(q,x) \right)
\]

where \( Z_q(x) \) is the normalization factor, \( F_{0} \) is the set of formulas with at least one grounding involving a query predicate, and \( g_{r}(q,x) \) is a binary function and equals to 1 if the \( r^{th} \) ground formula is true and 0 otherwise.

The problem of data conflict resolution introduced in this paper is to examine the correctness of conflicting facts and identify the true value corresponding to the real world. Thus, in our MLN model, we only need to define one query predictor as \( IsAccurate(fact) \), which describe the accuracy of a fact. The confidence predictors can be the feature of conflicting facts. In a discriminative MLN model as defined in Eq. (2), the evidence \( x \) can be arbitrary useful features. With the predefined features, we define some rules or the formulas in MLNs. With these rules, MLN can learn the weight of the roles and infer the accuracy of facts.

### 5.2 Features

We extract features from the following four aspects: basic features, inter-dependency between sources and facts, mutual implication between facts and mutual dependency between sources. In the following, we will represent the above four kinds of evidences respectively.

Here we will give an example for convenient illustration of our features. We tried to find out who wrote the book “Flash CS3: The Missing Manual” (ISBN: 0596510446). We found many different sets of authors from different online bookstores, and we show several of them in Table I. From the image of the book cover we found that ABC Books and AI Books provide the more accurate information, but the expressions are different. Auriga Ltd provides the duplicated information. In comparison, the information from textbooksNow and Powell’s Books is incomplete, and that from Book Lovers USA and Stratford Books is incorrect.

### Table I. Conflicting Information about Authors of a Book

<table>
<thead>
<tr>
<th>Source</th>
<th>Authors</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABC Books</td>
<td>Chris Grover, E. A. Vander Veer</td>
</tr>
<tr>
<td>AI Books</td>
<td>Veer, E. A., Vander, Grover, Chris</td>
</tr>
<tr>
<td>Auriga Ltd</td>
<td>E A Vander Veer, Chris Grover, Vander Veer, Grover Chris</td>
</tr>
<tr>
<td>textbooksNow</td>
<td>Vander Veer</td>
</tr>
<tr>
<td>Powell’s Books</td>
<td>Vander Veer, E A</td>
</tr>
<tr>
<td>Book Lovers USA</td>
<td>Moore, Emily</td>
</tr>
<tr>
<td>Stratford Books</td>
<td>Glover</td>
</tr>
</tbody>
</table>

1) Basic features

The basic features show source, entities, entity attributes, facts and the relationship between them. For example, a data source \( s \) provide a fact \( f \), this evidence can be presented as \( Provide(s, f) \). Also, to present the evidence that fact is a fact \( f \) about an entity attribute \( ea \), we define a predictor \( About(f, ea) \). In addition, for introducing the following voting rule, we introduce another evidence \( MaxFrequency(ea, f) \), which show that \( f \) is the most frequent fact about entity attribute \( ea \).

2) Inter-dependency between sources and facts

Intuitively, there exists the “trustworthy” data source that frequently provides more accurate facts than other sources. This can be validated in the table I, which the data sources ABC Books and AI Books are more trustworthy. And then, a fact is likely to be true if it is provided by trustworthy sources (especially if by many of them). Moreover, a data source is trustworthy if most facts it provides are true. Thus, we represent the trustworthy of a source and the accuracy of a fact as \( IsTrustworthy(s) \), \( IsAccurate(f) \) respectively.
3) Mutual implication between facts
Different facts about the same entity attribute may be conflicting. However, sometimes facts may be supportive to each other although they are slightly different. For example, for the book “Flash CS3: The Missing Manual”, one data source claims the author to be “Chris Grover, E. A. Vander Veer” and another one claims “Veer, E. A. Vander, Grover, Chris”. Though the expressions are different, two facts are equal. For another example, if two sources provide two facts: “E. A. Vander Veer” and “Vander Veer”, then the content of the first fact contain the second one and the last one actually supports the last one. In order to represent such relationships, we represent them as $Equal(f_1, f_2)$ and $Contain(f_1, f_2)$.

4) Mutual dependency between sources
If two data sources provide many same facts for many entity attributes, then the two sources will be dependent each other, so the facts provided by them for others entity attributes may have the same accuracy. To describe the mutual dependency between sources, we define a predictor $InterDepend(s_1, s_2)$ . To describe the relationship more precisely, we give the definition of the mutual dependency between sources.

**DEFINITION 2.** For two data sources $s_1$, $s_2$, if they satisfy the equation $rac{|F_1 \cap F_2|}{|E_A \cap E_A|} \geq \alpha$, then there exists a dependency between the two data sources. Where $F_1$ and $F_2$ represent the set of facts provided by $s_1$, $s_2$ respectively, $E_A$ and $E_A$ represent the set of entity attributes for which $s_1$, $s_2$ provide the facts, and the threshold $\alpha \in [0,1]$. In addition, we regard two facts as equal only if they provide the same value for the same entity attribute.

5.3 Rules
Based on common sense and our observations on real data, we introduce the detail rules in this section. These rules show the heuristic characteristic and are represented as predictor formulas in MLN. Because of the powerful and flexible knowledge representation, when new rules join, we can conveniently define new formulas to describe the rules and learn weights of the formulas to infer. Therefore, it makes our method more adaptive.

1) Rule1: Voting
For the problem of identifying the true value from conflicting facts, voting is a naive rule. Usually, the most frequent fact for an entity attribute is accurate.

$$MaxFrequency(ea, f) \Rightarrow IsAccurate(f)$$ (3)

2) Rule2: Mutual implication between facts
If two facts have the same content for an entity attribute $ea$, then they have the same accuracy. As a rule the detailed information is better than the simple one. Thus, if the content of a fact $f_1$ contains the one of another fact $f_2$ and $f_2$ is accurate, then $f_1$ is also accurate.

$$Equal(f_1, f_2) \Rightarrow \left( IsAccurate(f_1) \Leftrightarrow IsAccurate(f_2) \right)$$ (4)

$$About(f_{ea}) \wedge About(f_{ea}) \Rightarrow \left( IsAccurate(f_1) \Leftrightarrow IsAccurate(f_2) \right)$$ (5)

3) Rule3: Inter-dependency between sources and facts
Base on analysis in the previous section, often the data source which provides accurate facts is trustworthy and the fact provided by trustworthy data sources is accurate. Therefore, we introduce the following formulas:

$$IsAccurate(f) \wedge Provide(s, f) \Rightarrow IsTrustworthy(s)$$ (6)

$$IsTrustworthy(s) \wedge Provide(s, f) \Rightarrow IsAccurate(f)$$ (7)

4) Rule4: Mutual dependency between sources
If two data sources provide many same facts for many entity attributes, then there exists mutual dependency between the two sources. Therefore, the facts provided by them for other entity attributes likely have the same accuracy.

$$InterDepend(s_1, s_2) \wedge About(f_{ea}) \Rightarrow About(f_{ea})$$

$$\Rightarrow IsAccurate(f_1) \Leftrightarrow IsAccurate(f_2)$$ (8)

5.4 MLN Weight Training and Inference
In addition to the features and formulas, a MLN must also include the relative weights of each of these clauses. However, in our case we do not know the relative strength of all of the above formulas beforehand. Therefore, we must train the model to automatically learn the weights of each formula.

The state-of-the-art discriminative weight learning algorithm for MLNs is the voted perceptron algorithm. The voted perceptron is a gradient descent algorithm that will first set all the weights to zero. It will iterate through the training data and update the weights of each of the formulas based on whether the predicted value of the training set matches the true value. Finally, to prevent over-fitting, we will use the average weights of each iteration rather than the final weights. In order to train the data using the voted perceptron algorithm, we must know the expected number of true groundings of each clause. This problem is generally intractable, and therefore, the MC-SAT [15] algorithm is used for approximation.

After learning the weights of the formulas, inference in MLN can be conducted. Traditionally, MCMC [16] algorithms have been used for inference in probabilistic models, and satisfiability algorithms have been used for pure logical systems. Since a MLN contains both probabilistic and deterministic dependencies, neither will give good results. In our experiments, the MC-SAT algorithm will be used to determine the values of query predicates. The MC-SAT is an algorithm that combines MCMC and satisfiability techniques, and therefore performs well in MLN inferences.

Finally, according to the true value of each entity attribute, we merge all record referring to an entity to a single record. So we can get the result set.

6. EXPERIMENT EVALUATION
We perform experiments on two real data sets to examine the precision of our method. Our MLN model will be developed using the Alchemy system, which is a open source software package developed at the University of Washington.
that provides interfaces and algorithms for modeling Markov Logic Networks (alchemy.cs.washington.edu). In order to examine the effectiveness of our model, we perform experiments in the following aspects: (1) The precision of data conflict resolution; (2) The effects of changing the size of the training sample; (3) The effects of rules and their combination.

6.1 Datasets

1) Books
First, we extract book information from O’Reilly web site (http://oreilly.com/). The data set contains 1,258 books and we regard it as ground truth (Our data set does not contain information from O’Reilly). Then, for each book, we use its ISBN to search on www.abebooks.com, which returns the online bookstores that sell the book and the book information from each store. We develop a program to crawl and extract the book information and get 26,891 listings from 881 bookstores. Since the more conflicts appear on the book authors, we perform our method to resolve the data conflicts about the book authors. In addition, we do a pre-cleaning of authors’ names in order to remove some noise information.

2) Movies
In books data set, our method mainly resolves character data conflict, such as the authors. To validate the ability of our method for resolving various type data conflict, we collect data about movies and examine the method for numerical data such as movie runtime. First, we extract top 250 movie information from IMDB.com. Because of the authority of IMDB, we consider the information it provides as the standard facts (Also, information from IMDB.com is excluded from our data set). Then, according to the name of movies, we collect information of each movie using Google as described in [6]. The movies data set contains 7,119 movie listings from 952 data sources.

6.2 Experimental Results

1) Precision of data conflict resolution
We measure the performance of data conflict resolution via precision, which can be defined as the percentage of the entity attributes whose true values are identified correctly over all entity attributes. We compare the precision of our approach against voting and TruthFinder [6] in the above two data sets. Our approach is represented as MLN. Specially, TruthFinder will give the incomplete facts partial scores. However, in our method, the incomplete facts can be considered as false. Moreover, if two facts are equal, then the representation of them can be ignored. For example, for authors of a book, if the number of authors and each author’s information are correct, then fact is correct, without considering the sequence of each author.

We randomly select records referring to 600 entities from the books data set and the others are test set. And in the movies data set, the training set contains records referring to 120 entities. In this experiment, we set the threshold for mutual dependency between sources as $\alpha = 0.8$.

![Figure 3. Precision comparison among Voting, TruthFinder and our approach](image)

Fig.3 shows that our approach gets higher precision over other two approaches across the two data sets. In the books data set, our approach has an obvious advantage (the precision is 92.9%), it is because there exists plenty of incomplete or incorrect information for the book authors. It also validates the ability of our approach for resolving data conflict to some extent. But, our approach only gets a little higher precision than TruthFinder in the second data set. It is because that the movie runtime referring to a movie are not such variable as the book authors. And Voting also can get a high precision (87%). The experiment demonstrates that our approach improve effectively the precision utilizing multi-dimensional features.

2) Effects of changing the training data
To check the effect factors of our approach, we test the effectiveness of the size of training sample. In the books data set, we randomly select records referring to 300, 600, 900, 1200 entities as training samples and resolve data conflict utilizing our approach. Otherwise, in the movies data set, records referring to 60, 120, 180, 240 entities are selected.

![Figure 4. Effects of changing the training size (The books data set)](image)

![Figure 5. Effects of changing training size (The movies data set)](image)
Fig.4 and Fig.5 show the precision with increasing training sizes on the books and movies data set respectively. When increasing the training size, a gradual improvement on precision is obtained. More interesting, the slope of the two curves becomes flatter and flatter as increasing the training size. It shows that the bigger of training size, the more precise of our approach. But with the training size is bigger and bigger, its effectiveness will degrade gradually. In addition, when the training size is too large, labeling the training sample will be time-consuming.

3) Effects of rules and their combination

To validate the rules proposed in this paper, the performance of our approach utilizing various rules and their combination is reported. We test our four rules: Voting (denoted as V), Mutual implication between facts (denoted as I), Inter-dependency between sources and facts (denoted as SF), Mutual dependency between sources (denoted as D). We regard Voting as a basic rule, and then add one of the other three rules to the basic rule; finally we combine all the four rules. Thus, we get five rules and their combination. We test the precision of our method utilizing the five respectively.

This experiment is executed in the books data set, and the other setting is the same as (1). In Fig.6, it shows the precision using various rules and their combination. Obviously, each rule can improve the precision to some degree and it can validate the effectiveness of our rules. Among all rules, I and SF have a more obvious effect than D. On the one hand, it validates the existence of “trustworthy” data sources and the effect for identifying the true values from conflicting facts. On the other hand, conflicting information is often represented as incomplete or inconsistent, and it is one of the main troubles for resolving data conflict. In addition, we do not consider the dependency direction of D; it causes D not show enough significance.

This experiment also shows that our approach can combine various rules conveniently by adding or removing the corresponding formulas. Because data integration is a dynamic process, the appearance of new data conflict types can be predicted. We can extract new features and rules from new data conflict types, and then MLN weight training and inference are conducted. It also demonstrates the adaptability of our method.

7. CONCLUSION

In this paper we have presented an approach for resolving data conflict based on MLN. With multi-angle features and rules, our approach can effectively improve the accuracy of data conflict resolution. Based on the flexibility of knowledge representation as well as the ability to handle uncertainty of MLN, our approach can combine the imperfect and contradictory knowledge and is more adaptive.

8. REFERENCES


