Predicting Missing Provenance using Semantic Associations in Reservoir Engineering

Jing Zhao
Computer Science Dept.
University of Southern California
zhaoj@usc.edu

Karthik Gomadam
Accenture Technology Labs
karthik.gomadam@gmail.com

Viktor Prasanna
Ming Hsieh Dept. of Electrical Engineering
University of Southern California
prasanna@usc.edu

***Abstract*—Provenance is becoming an important issue as a reliable estimator of data quality. However, provenance collection mechanisms in the reservoir engineering domain often result in missing provenance information. In this paper, we address the problem of predicting missing provenance information in reservoir engineering. Based on the observation that data items with specific semantic "connections" may share the same provenance, our approach annotates data items with domain entities defined in a domain ontology, and represent these "connections" as sequences of relationships (also known as semantic associations) in the ontology graph. By analyzing annotated historical datasets with complete provenance information, we capture semantic associations that may imply identical provenance. A statistical analysis is applied to assign confidence values to the discovered associations, which indicate the trust of each association when it is used for future provenance prediction. The semantic associations, along with their confidence measures, are then used by a voting algorithm to predict the missing provenance information. Our evaluation shows that the average precision of our approach is above 85% when one third of the provenance information is missing.**

I. Introduction

Provenance is metadata that pertains to the derivation history of data objects [1], [2]. Information about how, when, and by whom a piece of data is created and modified, coupled with knowledge about domain processes, allows scientists and engineers to estimate the accuracy and the currency of data. Provenance information is useful in tasks such as data auditing, data quality estimation, and data integration [3], [4], [5].

Reservoir engineering is a domain that applies scientific principles to forecast and optimize the production of oilfields [6]. Applications such as optimizing production in oil fields, analyzing properties of reservoirs, and forecasting future oil production [7], all require provenance information for data quality control. In particular, provenance information plays an important role in quality estimation of reservoir models. Reservoir engineers create different reservoir models for different purposes. Each reservoir model is usually a complex dataset that consists of a large amount of fine-grained data items integrated from different data sources. For example, a reservoir model for production forecast is a dataset that contains thousands of data items. The quality of such a model determines the quality of the outcome of the production forecast process, which in turn influences the reservoir development decisions. The data quality of a reservoir model is measured based on the quality of data items it contains. Because these data items can be integrated from data sources across disciplines and organizations, and each category of data can be generated by various processes and approaches, provenance information becomes a significant evidence for data quality estimation.

Unfortunately, reservoir engineers do not always have complete provenance information for data items contained in the reservoir model. Legacy tools are still widely used in the domain, which do not provide provenance functionality thus do not support automatic provenance collection. Engineers have to manually annotate provenance for the output data. Besides, a lot of manual processes are involved in data archiving and integration. For example, engineers often copy data items from existing data volumes, and then paste them into a new reservoir model. Provenance information of the old data items has to be manually copied and linked to the data items that are pasted into the reservoir model during this procedure. These manual provenance annotation operations can be tedious and error-prone, thus produce incomplete provenance information for some data items.

We aim to predict missing provenance information so that we can achieve better data quality estimation. As defined in existing provenance models [8], [9], provenance of a data artifact has multiple attributes, including the process that generated the data artifact, agents who controlled the process, and other context information of the derivation history. As the first step of our exploration, we discuss how to predict the parent process of a data artifact in this paper, i.e., what kind of process was employed to create the data. The process used to create the data can greatly affect the output quality. Therefore accurate prediction about data's parent process will be valuable for the data quality estimation. For convenience, in the remaining parts of the paper, we say two data items have the "same/identical" provenance information if they were generated by the same kind of process.

Multiple data items contained in the same reservoir model often share the same provenance information. This is supported by the observation that reservoir engineers often employ the same process to create a collection or a series of fine-grained data items, and import them into the same reservoir model. For example, an engineer often utilizes the same

process to cleanse production data of all the oil wells [1] in the same block of the reservoir [2]. Cleansed data of each well is wrapped as a data item, and is then integrated into the same reservoir forecasting model. Given this observation, when provenance of a data item is missing, we can identify data items that share the same provenance information, and use their possibly existing provenance for our prediction.

To identify data items with the same provenance information can be a challenge. We find that data items linked by special semantic "connections" often share the same provenance. In the example we mentioned above, the production data is cleansed by the same process if *the corresponding wells are located in the same block*. In another example, the "original oil in place (OOIP) [3]" estimates of two wells are usually computed by the same process when *the wells belong to the same region*. In fact, these two "connections" reveal the hidden data generation patterns, which may be about the granularity of the data generation algorithms, or the coherence of certain types of data items. We can capture these special semantic "connections", and use them to identify data items with the same provenance information.

We detect and represent these special "connections" by using semantic associations [10], [11]. In our approach, we annotate reservoir models with a semantic domain ontology, which defines domain classes (such as "Well", "Block" and "OOIP Estimate"), domain entities (which are instances of domain classes), and domain relationships between classes and entities. Every data item contained in a reservoir model is annotated by a domain entity defined by the ontology. A semantic association is a sequence of relationships that interconnect two domain entities in the domain ontology. Using historical reservoir models that have complete provenance information, we discover semantic associations between the entities whose annotated data items share the same provenance.

For two data items sharing the same provenance, there may exist multiple associations connecting their annotation entities in the ontology graph. Only part of them always imply identical provenance thus can be used for future provenance prediction. To distinguish these associations from other useless ones, we employ a statistical approach to calculate confidence values for discovered associations. The confidence value of an association indicates its trust when it is used for provenance prediction, i.e., the probability that two data items whose annotation entities are connected by the association share the same provenance. Based on the discovered associations and their confidence values, when given a reservoir model containing data items with missing provenance information, we use a voting scheme to predict the missing provenance.

The contributions of our work include:

1) We propose a novel approach to solve a practical problem in reservoir engineering domain: incomplete provenance information of reservoir models. Based on the observation that data items in a reservoir model may be created by the same process, we use the possibly existing provenance to predict the missing provenance.
2) We utilize semantic associations to reveal the implicit semantic "connections" between data items that share the same provenance information.
3) We provide a statistical approach to calculate confidence values of discovered semantic associations. The confidence values allow us to find data items whose provenance is most probable to be identical with the missing one.

[1] http://en.wikipedia.org/wiki/Oil_well

[2] A reservoir is divided into several blocks, which contains a set of wells

[3] http://en.wikipedia.org/wiki/Oil_in_place

A. A Motivating Scenario

We introduce a typical scenario in reservoir engineering as our motivating example. A reservoir engineer, Jim, is creating a simulation model for production forecast of several reservoirs. Such a complex simulation model is a big dataset containing different categories of information, e.g., descriptions of reservoir capability, historical production records, and surface facility constraints. Each category of information further consists of a lot of fine-grained data items. Data contained in the model is integrated from different data sources, across disciplines and organizations. Various approaches and processes may be employed to create the data. Usually even the same kind of data can be generated by different methods: e.g., the OOIP estimates of wells can be calculated by using different algorithms.

Jim integrates OOIP estimates of all the wells from several different data sources. For example, OOIP estimates of wells in several regions are all coming from another reservoir engineer Jane. For each region, Jane selects the most suitable OOIP calculation approach, and uses the approach to compute OOIP estimates for all the wells in the region. When integrating the data into the reservoir model, as important evidence of future quality estimation, provenance information has to be manually annotated for each OOIP estimate about its generating process, i.e., what approach has been employed to calculate the OOIP estimates. Provenance of some OOIP estimates may be missing during the tedious manual annotation procedure. However, if we can detect the fact that Jane usually utilizes the same approach to compute OOIP for all the wells belonging to the same region, we can use the existing provenance to predict the missing ones.

The rest of the paper is organized as follows: Section II formally defines the problem and provides an overview of our approach. Section III introduces our semantic domain ontology and how we use the domain ontology to annotate data items in a reservoir model. In Section IV and V, we introduce semantic associations, and present the approach that captures specific associations and calculates their corresponding confidence values. We evaluate our work and present experimental results in Section VII. Related work is discussed in Section VIII, and we conclude the paper in Section IX.

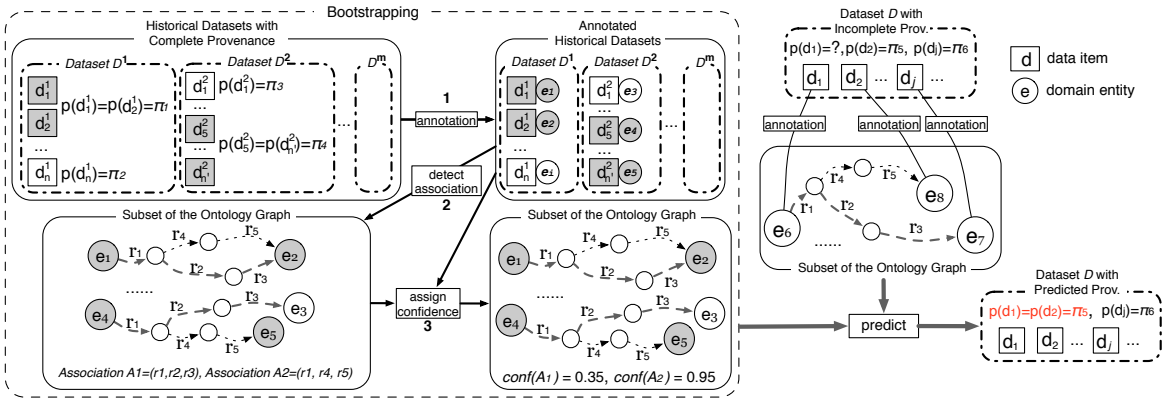


Fig. 1: Overview of bootstrapping and provenance prediction

II. Overview

We define the problem and provide our approach overview in this section. We consider a reservoir model as a dataset D^k=$\{d_1^k, d_2^k, \ldots, d_n^k\}$, which consists of data items $d_i^k, 1 \leq i \leq n$. The provenance of a data item d_i^k is defined as $p(d_i^k)$, which is the information about the process that created d_i^k.

The provenance information for each data item is either complete or missing. We define an indicator function $f_c : d_i^k \rightarrow \{0, 1\}$:

$$f_c(d_i^k) = \begin{cases} 0 & , p(d_i^k) \text{ is missing} \\ 1 & , p(d_i^k) \text{ is complete} \end{cases}$$

This definition allows us to divide the data items in D^k into two sets, namely: $D^k_{complete} = \{d_i^k | f_c(d_i^k) = 1\}$ and $D^k_{missing} = \{d_i^k | f_c(d_i^k) = 0\}$. Data items d_i and d_j have the same provenance if $p(d_i) = p(d_j)$, i.e., the process creating d_i and d_j is the same.

The overview of our approach is illustrated in Figure 1. To bootstrap our system, we analyze historical datasets (D^1, D^2, …, D^m in the figure), which are reservoir models with complete provenance information, so as to discover semantic associations that imply the identical provenance. In particular, the bootstrapping contains 3 steps:

1) Annotation. We annotate historical datasets by using a semantic domain ontology. Every data item is annotated by a domain entity defined in the ontology. For example, in the figure, the data item d_1^1 is annotated by entity e_1, and d_2^1 is annotated by e_2.

2) Association detection. We first identify data items in each dataset that share the same provenance information. In Figure 1, data items d_1^1 and d_2^1 share the same provenance in dataset D^1, and data items d_5^2 and $d_{n'}^2$ share the same provenance in dataset D^2. We use dark colors to illustrate them and their annotation entities. Once these pairs of data items are detected, we identify the semantic associations in the ontology graph between the domain entities that the data items are annotated with. Each association is represented as a sequence of relationships. In our example, two associations A_1 and A_2 are discovered between e_1 and e_2, and A_2 also connects e_4 and e_5.

3) Confidence assignment. Not all the discovered associations can be perfectly used for future provenance predication. For example, A_1 also connects e_3 and e_4, whose annotated data items d_1^2 and d_5^2 do not share the same provenance. For each discovered association, by analyzing all the historical datasets and provenance information, we compute confidence values to reflect the probability that two data items whose annotation entities are connected by the association share the same provenance. In our example, we suppose the confidence of association A_1 is 0.35, and the confidence of A_2 is 0.95.

After the bootstrapping procedure, we utilize the discovered associations and their confidence values to predict the missing provenance for data items in a dataset (data item d_1 in dataset D in our example). Each data item in D is also annotated by a domain entity defined by our domain ontology. As illustrated in the figure, d_1 is annotated by e_6, d_2 is annotated by e_8, and d_j is annotated by e_7. For a data item with missing provenance, we identify domain entities that connect to its annotation entity through semantic associations discovered in the bootstrapping procedure. Data items annotated by these entities are probable to share the same provenance with the one with missing provenance. Among their existing provenance, we process a voting algorithm based on the association confidence values to generate our prediction. In the example, we find that e_6 is connected with e_8 by the association A_2, and connected with e_7 by A_1. Because the confidence value of A_2 is higher than A_1, in the voting procedure d_2 will "beat" d_j, and we take provenance of d_2 as our prediction (note that the example is just a naive case since we ignore other associations and entities).

We discuss details of each step of our approach in the following sections.

III. Annotation

We use a semantic domain ontology, represented by the OWL ontology language, to annotate reservoir models. The ontology defines domain classes, such as "*Reservoir*", "*Well*", and "*OOIP Estimate*", and their relationships, such as "*WellHasOOIPEstimate*" and "*ReservoirContainsWell*". The ontology also defines domain entities which are instances of domain

classes, including physical oilfield entities such as real-world oil wells, and logical entities such as simulation scenarios.

For a reservoir model, we annotate every data item with a domain entity. For example, a data item about the OOIP estimate of a well W_1 is annotated by a domain entity o_1, which indicates an instance of the domain class "*OOIP Estimate*". The ontology also contains a domain entity w_1 to represent the well W_1, and w_1 and o_1 are connected by a relationship "*WellHasOOIPEstimate*". Data items in different reservoir models can be annotated by the same domain entity. For example, if two reservoir simulation models both contain a data item about the OOIP estimate of well W_1 (these two data items may have different values), both data items will be annotated by o_1. The annotation function is implemented as a module in our prior work, the Integrated Asset Management framework (IAM) [4] [12], which was developed to integrate heterogeneous data representation formats used by proprietary software systems in the energy domain. We skip the details about the annotation implementation here since it is out of the scope of this paper.

In our work, we use function f_a to indicate the mapping between a data item and its annotation domain entity. We define $f_a(d_i^k) = e_i$ if the data item d_i^k is annotated by the domain entity e_i.

IV. Semantic Association

An ontology graph can be derived from the domain ontology, where each vertex in the graph is a domain entity, and the edges between vertices are corresponding domain relationships between entities. The annotation procedure maps a reservoir model to a subset of the whole ontology graph: each data item is mapped to a domain entity, and the implicit semantic "connections" between data items can be represented explicitly by the paths between domain entities. This allows us to detect those special semantic "connections" that imply identical provenance information.

We use semantic associations to represent the paths between domain entities in the ontology graph. As defined in [10], [11], a semantic association, noted as $\mathcal{A}$, is a sequence of relationships, $r_1, r_2, \ldots, r_n$, for which there exist a sequence of domain entities, $e_1, e_2, e_3, \ldots, e_n, e_{n+1}$ that makes the sequence $e_1, r_1, e_2, r_2, e_3, \ldots, e_n, r_n, e_{n+1}$ a path [5] in the ontology graph. In this definition, domain entity e_1 is associated with e_{n+1} by association $\mathcal{A}$, which is denoted by $e_i \xrightarrow{\mathcal{A}} e_j$.

As we have stated, certain semantic associations lead to identical provenance information. Figure 2 depicts such a simple scenario. In Figure 2, d_1^j and d_2^j are two data items in a dataset D^j with complete provenance information, and d_1^k and d_2^k are two data items in another dataset D^k, where the provenance of d_1^k is missing and the provenance of d_2^k still exists. Each data item has been annotated by a domain entity:

[4]The IAM Framework, http://pgroup.usc.edu/iam/

[5]Similar with [10], we ignore direction of relationships when we discuss paths in the ontology graph. Thus $\{r_1, r_2, r_2, r_1\}$ is a valid association in Figure 2.

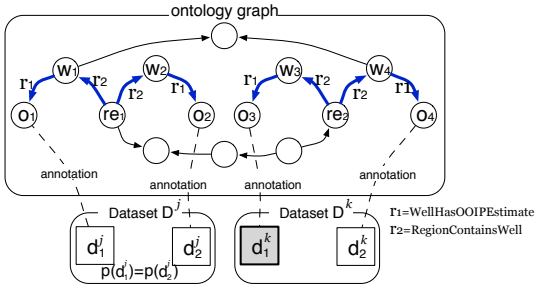


Fig. 2: Using semantic association for provenance prediction

d_1^j and d_2^j are annotated by o_1 and o_2, and d_1^k and d_2^k are annotated by o_3 and o_4, where o_1, o_2, o_3 and o_4 are OOIP estimates for well w_1, w_2, w_3, w_4, respectively. Well w_1 and w_2 belong to region re_1, and w_3 and w_4 belong to region re_2.

Suppose we find that d_1^j and d_2^j have the same provenance information. A semantic association between their annotation domain entities (o_1 and o_2) in the ontology graph, say $\mathcal{A}$={*WellHasOOIPEstimate*, *RegionContainsWell*, *RegionContainsWell*, *WellHasOOIPEstimate*}, is illustrated using bold lines connecting o_1 and o_2 in Figure 2. If we find that most other pairs of data items whose annotation entities are associated by $\mathcal{A}$, also have the same provenance (not depicted in Figure 2), we can speculate that the association $\mathcal{A}$ is a special association that implies identical provenance information. In fact, this association reveals an implicit data generation pattern: OOIP estimates of wells belonging to the same region are usually generated by the same kind of process. Since o_3 and o_4 are also connected by $\mathcal{A}$, we can predict that the provenance of d_1^k is the same as the provenance of d_2^k. Thus we use the provenance of d_2^k as our prediction for d_1^k.

A. Detecting Associations from Historical Datasets

We analyze historical datasets to discover all the possible semantic associations that may imply identical provenance information. Each data item belonging to these datasets has complete provenance information and is semantically annotated with domain entities defined by our domain ontology.

Algorithm 1 depicts our approach. In Algorithm 1, we first group data items by their provenance information. Data items with identical provenance will be grouped together. For each two data items with identical provenance information, we identify their annotated domain entities, and then compute semantic associations between domain entities. To compute semantic associations between two domain entities can be done by using the $\rho - query$ defined in [10], [11]. To keep the number of detected semantic associations small, we count the frequency of occurrence for each association. We eliminate the associations whose frequency is less than a threshold.

V. Confidence of Association

For a semantic association $\mathcal{A}$, its confidence $conf(\mathcal{A})$ is a measure of the probability that the two data items have identical provenance, if their annotation domain entities are

Algorithm 1: Semantic association detection for provenance prediction

Input: a set of historical datasets $H = \{D^k\}$ with complete provenance information
Output: all the semantic associations between domain entities whose annotated data items share identical provenance

$\alpha \leftarrow \{\}$;
foreach *dataset* $D^k = \{d_l^k\}$, $D^k \in H$ **do**
  group data items according to their provenance;
  foreach (d_i^k, d_j^k) *such that* $p(d_i^k) = p(d_j^k)$ **do**
    identify domain entities $e_{i'}$ and $e_{j'}$ such that $f_a(d_i^k) = e_{i'}$, $f_a(d_j^k) = e_{j'}$;
    detect all the semantic associations $\{\mathcal{A}\}$ such that $e_{i'} \xrightarrow{\mathcal{A}} e_{j'}$;
    $\alpha \leftarrow \alpha \cup \{\mathcal{A}\}$;
  end
end
return α

associated by $\mathcal{A}$:

$$conf(\mathcal{A}) = \mathbb{P}\left(p(d_i^k) = p(d_j^k) \mid f_a(d_i^k) \xrightarrow{\mathcal{A}} f_a(d_j^k)\right) \quad (1)$$

In Equation 1, d_i^k and d_j^k are any two data items in a dataset D^k. Their annotation domain entities, $f_a(d_i^k)$ and $f_a(d_j^k)$, are associated by the association $\mathcal{A}^l$ in the ontology graph (recall we define f_a as the annotation function in Section III).

In particular, for a specified ontology item e_i, we are interested in the following conditional confidence measure:

$$conf(\mathcal{A}|e_i) = \mathbb{P}\left(p(d_i^k) = p(d_j^k) \mid f_a(d_i^k) \xrightarrow{\mathcal{A}} f_a(d_j^k), f_a(d_i^k) = e_i\right) \quad (2)$$

Given a domain entity e_i, Equation 2 defines the confidence of an association when it associates e_i with other entities. The same semantic association can have different conditional confidence values for different given entities. This can be illustrated by an example from the reservoir engineering domain. In reservoir R_1, because the cumulative water pressure is measured based on the same sensor mechanism for the whole reservoir, the cumulative water pressure of wells in the same reservoir share the same provenance information, whereas in reservoir R_2, wells in different blocks of the reservoir are instrumented with different sensor mechanisms and thus, the probability for cumulative water pressure of wells in a reservoir sharing the same provenance information is low. Using the same confidence to the *"wells contained in the same reservoir"* association, would lead to poor results when analyzing datasets from reservoir R_2. To address such scenarios, we define the above conditional confidence measure.

In our implementation, we calculate $conf(\mathcal{A}|e_i)$ by analyzing the historical datasets $H = D^1, D^2, ..., D^m$. For each dataset D^k, we count all the data item pairs (d_i^k, d_j^k) such that d_i^k is annotated by e_i, and e_i is associated with the annotation entity of d_j^k by $\mathcal{A}$. We use $C_{total}^k(\mathcal{A}, e_i)$ to specify the number of qualified pairs. Among these pairs, we count all the data item pairs (d_i^k, d_j^k) that also satisfy $p(d_i^k) = p(d_j^k)$. The number of these item pairs is denoted as $C_{identical}^k(\mathcal{A}, e_i)$. The confidence value of $conf(\mathcal{A}|e_i)$ is then estimated as:

$$conf(\mathcal{A}|e_i) = \frac{\sum_k C_{identical}^k(\mathcal{A}, e_i)}{\sum_k C_{total}^k(\mathcal{A}, e_i)} \quad (3)$$

In our bootstrapping process, after we discover a semantic association $\mathcal{A}$, for each domain entity e_i that is associated with another entity by $\mathcal{A}$, we compute the corresponding $conf(\mathcal{A}|e_i)$.

VI. Prediction

After the bootstrapping process, we use discovered semantic associations and their (conditional) confidence values to predict missing provenance. Suppose we are to predict missing provenance for data items in a dataset D. We first annotate every data item in D with a domain entity defined by our domain ontology. Then to predict the missing provenance of a data item d_i annotated by domain entity e_i, we retrieve the conditional confidence values of all the associations that associate e_i with other entities:

$v_i^{conf} = \{conf(\mathcal{A}^1|e_i), conf(\mathcal{A}^2|e_i), \ldots, conf(\mathcal{A}^n|e_i)\}$,

where n is number of associations that connect e_i with other entities. Note that all the conditional confidence values have been calculated in our bootstrapping process.

Next, we spread the conditional confidence values in the ontology graph. For each association A^l $(1 \leq l \leq n)$, we tag the conditional confidence value $conf(\mathcal{A}^l|e_i)$ to all the domain entities that are associated with e_i by A^l. At each domain entity e_j, in case that multiple associations exist between e_i and e_j, we tag e_j with the highest association confidence value. A Breath-First search in the ontology graph does this process in an efficient manner.

In the dataset D, we identify all the data items satisfying two conditions: 1) they still have complete provenance information (i.e., they are contained in $D_{complete}$), and 2) their annotation domain entities are associated with e_i by at least one A^l. Each such data item is then tagged by the same confidence value that is tagged to its annotation domain entity. These data items are likely to share the same provenance with d_i. The confidence values tagged to them indicate the probability of sharing identical provenance.

Finally, we employ a voting process to select the provenance information with the highest confidence value: every data item tagged by a confidence value greater than a threshold votes to its provenance information, using its confidence value as the weight of the vote. The provenance with the highest votes is our prediction.

Algorithm 2 illustrates the pseudo code of the whole prediction algorithm. In the algorithm, we use a set S^i to store data items in D that still have complete provenance information and whose annotation domain entities are semantically associated with e_i. We use ζ to indicate the confidence values tagged to

Algorithm 2: Predict missing provenance of data item d_i in dataset D

Input: $D_{complete}$; d_i, where $f_c(d_i) = 0$ and $f_a(d_i) = e_i$; $v_i^{conf} = \{conf(\mathcal{A}^l|e_i)\}$, $1 \leq l \leq n$; threshold σ
Output: $p(d_i)$

```
S^i ← {};
initialize ζ(·) = 0 for data items in D_complete;
foreach association A^l, 1 ≤ l ≤ n do
    identify S^{i,l} = {d_j}, d_j ∈ D_complete,
    f_a(d_i) --A^l--> f_a(d_j);
    S^i ← S^i ∪ S^{i,l};
    foreach each d_j in S^{i,l} do
        if conf(A^l|e_i) > σ and conf(A^l|e_i) > ζ(d_j) then
            ζ(d_j) = conf(A^l|e_i);
    end
end
initialize V(·) = 0;
foreach each d_j ∈ S^i, ζ(d_j) > 0 do
    V(p(d_j)) ← V(p(d_j)) + ζ(d_j);
end
return p(d_j) with the highest V
```

these data items. Each data item d_j is assigned by a conditional confidence value $conf(\mathcal{A}^l|e_i)$, in which $\mathcal{A}^l$ is the association connecting e_i to $f_a(d_j)$. When multiple associations exist between e_i and $f_a(d_j)$, the highest association confidence value is assigned. In the voting process, data items in S^i with confidence values greater than the threshold σ join the vote. Their ζ values are the weight of the votes. The provenance information with the highest votes $\mathcal{V}$ is our prediction.

VII. Evaluation

We evaluate our approach in this section. We create synthetic datasets based on real data collected from the reservoir engineering domain. We measure the accuracy of our approach under different provenance loss ratio, and compare our approach with another 2 baseline approaches.

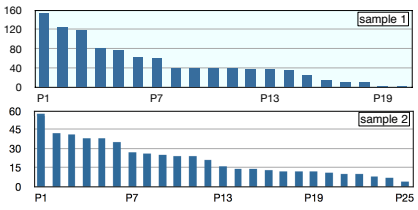


Fig. 3: Number of data items in sample datasets created by each process

A. Experiment Setup

We first collect two groups of reservoir model samples from two practical use cases in the reservoir engineering domain. Each group contains 10 datasets with complete provenance. Datasets in one group contain around 1000 data items each, which are used for reservoir forecasting. Datasets in the other group are for the production optimization problem, containing around 500 data items. From each group we pick one dataset and show the number of data items created by each individual process in Figure 3.

The historical datasets with complete provenance information are synthesized by duplicating the real datasets into a reasonable scale, say, 2,000 copies. In practice, each data item can be created by different kinds of processes. For example, in our first sample group, algorithms described in [13], [14], [15], [16] can all be employed to calculate the same category of data. When we create the historical datasets, data items annotated by entities of the same ontology class are designed to be possibly generated by different kinds of processes. We determine the process according to entity properties and usage frequencies of different processes learnt from the sample groups. 10% of the historical datasets are used as test datasets, and the remaining are used for our bootstrapping process. In each test dataset, we randomly pick data items, drop their provenance information, and regard the dropped provenance as "missing provenance".

We ran our experiments on a machine with 3.06 GHz Intel Core i3 CPU and 4GB memory. While the offline bootstrapping process usually takes several minutes, which is proportional to the size of the historical datasets; the prediction step of our approach responds within seconds.

B. Baseline Approaches

We compare our approach to two approaches:

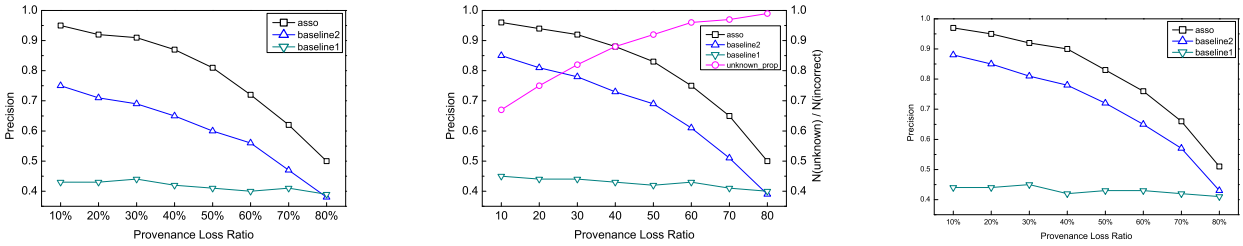
Baseline 1: For a data item annotated by an entity e_i, we predict its missing provenance by simply selecting the generation process which were most frequently used to create data items annotated by e_i in the historical datasets.

Baseline 2: We predict provenance only considering provenance similarity between domain entity pairs, but without introducing the semantic associations. Suppose we are predicting missing provenance for a data item d_i contained in the dataset D, and the domain entity annotated to d_i is e_i. For every other entity e_j in the ontology, this approach counts the total number of times that e_i and e_j's annotated data items share the same provenance when contained in the same historical dataset. We express the counting number as $C_{e_i}(e_j)$. Then in the dataset D, among the data items with complete provenance, we select the provenance of the data item whose annotation entity has the biggest C_{e_i} as our prediction.

C. Results and Discussion

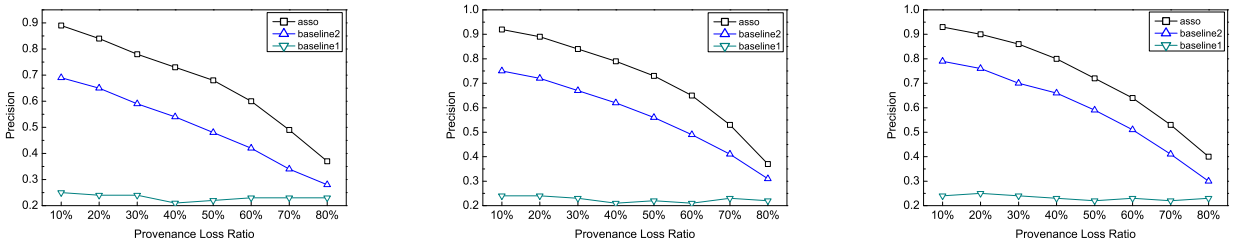
As previously mentioned, we take n datasets as historical data, where n varies from 500 to 2000. We calculate the prediction accuracy of different approaches under different provenance loss ratios by comparing our provenance prediction with the ground truth.

Figure 4 and 5 shows the evaluation results on two use cases. The curve "asso" illustrates the prediction accuracy of our approach, which has the highest precision among all the prediction methods in both cases. In general, our approach can



(a) 500 historical datasets (b) 1000 historical datasets (c) 2000 historical datasets

Fig. 4: Precision of provenance prediction for use case 1



(a) 500 historical datasets (b) 1000 historical datasets (c) 2000 historical datasets

Fig. 5: Precision of provenance prediction for use case 2

achieve an average precision above 85% when one third of the provenance information is missing.

Generally, "baseline1" has a relatively fixed precision and is not affected by changes in the provenance loss ratio. This is because "baseline1" only takes the most-frequently-used process as prediction. Since various processes can be employed to create the same category of data items, this approach cannot achieve a good accuracy.

The accuracy of the other two approaches declines when the percentage of the missing provenance information increases. Intuitively, when we are predicting the missing provenance for a data item, with more provenance information missing, the probability that all the relevant data items lose their provenance information grows. If all the relevant data items that share the same provenance with the given data item lose their provenance, we mark this provenance information as "unknown". The "unknown" provenance causes incorrect predictions since in that case our approach cannot find any existing provenance as prediction. We measure the proportion of "unknown" provenance to incorrect predictions with respect to the "asso" curve in case of 1000 historical datasets. The "unknown_prop" curve in Figure 4(b) illustrates the increasing proportion of "unknown" provenance as the provenance loss ratio increases. Note that we can still use provenance predicted by "baseline1" as a reasonable guess for the "unknown" provenance.

The precision of our approach is better than "baseline2". In "baseline2", we look for entity pairs whose annotated historical data items share the same provenance information. Because each historical dataset only "covers" part of the entities defined by the domain ontology, this approach cannot find all the possible entity pairs with the "same-provenance" connection. Thus we may miss useful data items that share the same provenance with the given data item. Our approach abstracts the "same-provenance" connections as semantic associations. When we find two data items have the same provenance information, specific semantic associations between their annotation domain entities reveal the data generation pattern which derives identical provenance, and we can use this revealed pattern to speculate identical provenance between data items annotated by other entities.

VIII. Related Work

We review the work related to our research in this section. Existing work discusses provenance from different perspectives, such as provenance tracking and management [17], [18], provenance storage [19], [20], and provenance querying [21], [22]. Researchers in the e-science and the scientific workflow community have done extensive work in the area of provenance [23], [24], [25], [26]. Simmhan et al. did a survey about the work of provenance in the e-Science community [2]. While most of the previous research in this area has focused on collection, storage, and querying aspects of provenance, in this paper, we address and provide an approach to solve a practical challenge of missing provenance in the domain of reservoir engineering.

Semantic Web techniques have been used in provenance management. For example, Zhao *et al* discussed the application of semantic Web techniques for managing and querying provenance information as a part of the myGrid project [26], [27]. A semantic provenance model is proposed in their work and domain knowledge is annotated and linked to provenance information. Sahoo et al. discusses semantic provenance management for e-Science in [28]. We also employ semantic Web techniques in our work: a domain ontology is used to annotate data items and specific semantic associations are detected in the ontology graph to infer identical provenance.

Semantic association has been discussed in [11] and [10], in which two entities are semantically associated if they are "semantically connected" or "semantically similar". In [11] Anyanwu et al. define semantic associations and discuss queries of semantic associations. Ranking of semantic associations is further discussed in [10], [29], [30], [31]. Semantic association is used by applications such as documents ranking in [32]. In our work, we use semantic associations to reveal the hidden semantic "connections" between data items that share the same provenance information.

IX. Conclusion

In this paper, we have discussed our work in predicting missing provenance information for reservoir engineering. Our approach utilizes semantic associations to capture hidden semantic "connections" between fine-grained data items sharing identical provenance. By analyzing historical datasets annotated by a domain ontology, we detect specific semantic associations in the ontology graph that may imply identical provenance. A statistical approach has been implemented to measure the confidence of semantic associations. We utilize the discovered associations to identify data items that are likely to share the same provenance with the given data item, and use a voting algorithm based on the confidence values to predict missing provenance.

Provenance information is emerging as a critical aspect of data analysis and integration in many domains. In this paper, we have addressed one aspect of the problem, prediction of missing provenance, with motivating examples from the domain of reservoir engineering. Our continuing research will focus on predicting more complicated provenance information and correcting inconsistent provenance. We will also apply our approach to a provenance integration framework.

Acknowledgment

This work is supported by Chevron Corp. under the joint project, Center for Interactive Smart Oilfield Technologies (CiSoft), at the University of Southern California.

References

[1] P. Buneman, S. Khanna, and W. Tan, "Data provenance: some basic issues," in *Foundations of Software Technology and Theoretical Computer Science*, 2000.

[2] Y. L. Simmhan, B. Plale, and D. Gannon, "A survey of data provenance in e-science," *SIGMOD Record*, vol. 34, 2005.

[3] P. P. da Silva, D. L. McGuinness, and R. McCool, "Knowledge provenance infrastructure," *IEEE Data Engineering Bulletin*, vol. 26, 2003.

[4] H. V. Jagadish and F. Olken, "Database management for life sciences research," *SIGMOD Record*, vol. 33, 2004.

[5] S. Miles, P. Groth, M. Branco, and L. Moreau, "The requirements of recording and using provenance in e-science experiments," University of Southampton, Tech. Rep., 2005.

[6] B. C. Craft, M. Hawkins, and R. E. Terry, *Applied Petroleum Reservoir Engineering*, 2nd ed. Prentice Hall, 1990.

[7] R. Soma and et. al., "Semantic web technologies for smart oil field applications," in *Intelligent Energy Conference and Exhibition*, 2008.

[8] L. Moreau and et. al., "The open provenance model core specification (v1.1)," *Future Generation Computer Systems*, 2009.

[9] S. Sahoo, D. Weatherly, R. Mutharaju, P. Anantharam, A. Sheth, and R. Tarleton, "Ontology-driven provenance management in escience: An application in parasite research," *OTM*, 2009.

[10] B. Aleman-Meza, C. Halaschek, I. B. Arpinar, and A. Sheth, "Context-aware semantic association ranking," in *SWDB*, 2003.

[11] K. Anyanwu and A. Sheth, "p-queries: Enabling querying for semantic associations on the semantic web," in *WWW*, 2003.

[12] R. Soma, A. Bakshi, and V. K. Prasanna, "A semantic framework for integrated asset management," in *CCGrid*, 2007.

[13] U. Demiryurek, F. Banaei-Kashani, and C. Shahabi, "Neural-network based sensitivity analysis for injector-producer relationship identification," in *Intelligent Energy Conference and Exhibition*, 2008.

[14] H. Lee, K. Yao, O. Okpani, A. Nakano, and I. Ershaghi, "Identifying injector-producer relationship in waterflood using hybrid constrained nonlinear optimization," in *SPE Western Regional Meeting*, 2010.

[15] F. Liu, J. Mendel, and A. Nejad, "Forecasting injector/producer relationships from production and injection rates using an extended kalman filter," *SPE Journal*, vol. 14, 2009.

[16] M. Sayarpour, E. Zuluaga, C. Kabir, and L. W. Lake, "The use of capacitance-resistance models for rapid estimation of waterflood performance and optimization," *Journal of Petroleum Science and Engineering*, vol. 69, 2009.

[17] P. Buneman, A. Chapman, and J. Cheney, "Provenance management in curated databases," in *SIGMOD*, 2006.

[18] R. Ikeda and J. Widom, "Panda: A system for provenance and data," in *TaPP*, 2010.

[19] M. K. Anand, S. Bowers, T. McPhillips, and B. Ludascher, "Efficient provenance storage over nested data collections," in *EDBT*, 2009.

[20] A. Chapman, H. V. Jagadish, and P. Ramanan, "Efficient provenance storage," in *SIGMOD*, 2008.

[21] T. Heinis and G. Alonso, "Efficient lineage tracking for scientic workows," in *SIGMOD*, 2008.

[22] M. K. Anand, S. Bowers, and B. Ludscher, "Techniques for efficiently querying scientific workflow provenance graphs," in *EDBT*, 2010.

[23] C. Pancerella and et. al., "Metadata in the collaboratory for multi-scale chemical science," in *Dublin Core Conference*, 2003.

[24] I. Foster, J. S. Vockler, M. Wilde, and Y. Zhao, "Chimera: A virtual data system for representing, querying, and automating data derivation," in *SSDBM*, 2002.

[25] J. Frew and R. Bose, "Earth system science workbench: A data management infrastructure for earth science products," in *SSDBM*, 2001.

[26] J. Zhao, C. Goble, M. Greenwood, C. Wroe, and R. Stevens, "Annotating, linking and browsing provenance logs for e-science," in *ISWC Workshop on Retrieval of Scientific Data*, 2003.

[27] J. Zhao, C. Wroe, C. Goble, R. Stevens, S. Bechhofer, D. Quan, and M. Greenwood, "Using semantic web technologies for representing e-science provenance," in *ISWC*, 2004.

[28] S. Sahoo, A. Sheth, and C. Henson, "Semantic provenance for escience: Managing the deluge of scientific data," *Internet Computing, IEEE*, vol. 12, 2008.

[29] K. Anyanwu, A. Maduku, and A. P. Sheth, "Semrank: ranking complex relationship search results on the semantic web," in *WWW*, 2005.

[30] B. Aleman-Meza, C. Halaschek-Wiener, I. B. Arpinar, C. Ramakrishnan, and A. Sheth, "Ranking complex relationships on the semantic web," *IEEE Internet Computing*, 2005.

[31] G. Kasneci, F. M. Suchanek, G. Ifrim, M. Ramanath, and G. Weikum, "Naga: Searching and ranking knowledge," *Data Engineering, International Conference on*, 2008.

[32] B. Aleman-Meza, I. B. Arpinar, M. V. Nural, and A. P. Sheth, "Ranking documents semantically using ontological relationships," in *Proceedings of the 2010 IEEE Fourth International Conference on Semantic Computing*, ser. ICSC '10, 2010.