Entity-Aware Query Processing for Heterogeneous Data with Uncertainty and Correlations

Ekaterini Ioannou
L3S Research Center
Leibniz Universität Hannover
ioannou@L3S.de

ABSTRACT

Many modern systems rely on rich heterogeneous data that has been integrated from a variety of different applications and sources. To successfully perform their tasks, these systems require to know which data refer to the same real-world entities, such as locations, people, or movies. My work focuses on addressing this requirement through a new approach for entity-aware query processing over heterogeneous data. Data provided for integration is processed to generate the possible entities and linkages between these entities. This information is never merged with the original data, but used during query processing to provide entity-aware results that reflect the real-world entities existing in the current data. Special emphasis is given to the effective management of uncertainty and correlations that either exist in the original data, or are generated by data matching techniques.

Advisor: Prof. Dr. Wolfgang Nejdl

1. INTRODUCTION

Retrieving the complete set of information that corresponds to the same real-world entity (e.g., a specific person, an event, or a movie) is an important task, especially when integrating heterogeneous data from a variety of different applications and sources. Many existing approaches tackle this problem by identifying data which describe the same real-world entity[1], and then merging the results with the original data using thresholds or human intervention. This paper describes a new approach which is more suitable for addressing the challenges of this problem.

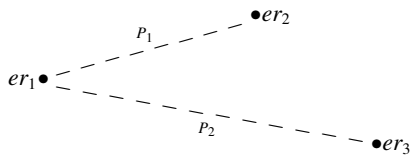
One big group of applications that contain data with uncertainty is social applications, for example *MySpace*, *Facebook*, *Flickr*, and *You-Tube*. These applications allow users to publish data while enabling data sharing and interactions between users. Uncertainty in such data appears for various reasons. One example is data uncertainty that comes directly from the extraction process, due to the very low quality typically accompanies the unstructured data of such applications [22]. Another example is the uncertainty introduced when building structures for processing the data, e.g., social network analysis [1]. A second big group of data with uncertainty arises from the use of compression and structure summarization. Recent approaches for information exchange use this to minimize the network cost as much as possible (e.g., Google BigTable [9]). These approaches typically affect the quality of data. For instance, data retrieved from a Bloom filter summarizing structure come with a probability that indicates the belief that the structure has that the specific data is correct.

[1]A technique referred to as data linkage, matching, de-duplication, reconciliation, or alignment.

EDBT 2009, March 24–26, 2009, Saint Petersburg, Russia.

Film from Wikipedia:
er_1 = { (-; Daniel Radcliffe; p),
(-; Emma Watson; p),
(-; J.K. Rowling; p),
(-; Fantasy), . . . }
DVD from Amazon:
er_2 = { (title; Harry Potter and the Chamber . . . , 2002; p),
(starring; Radcliffe, Daniel; p),
(starring; Watson, Emma (II); p), . . . }
Book from Amazon:
er_3 = { (title; Harry Potter and the Chamber of Secrets; p),
(gender; Fantasy; p),
(author; J.K. Rowling; p), . . . }

(a)



(b)

Figure 1: (a) Heterogeneous data describing movies, illustrated as a set of triples with attribute, value, and probability, and (b) the possible matches between them generated by entity linkage algorithms.

Let us now illustrate the problem of identifying entities in heterogeneous data through an example. Consider a system that integrates data coming from a number of different sources. Figure 1 shows data describing a film from Wikipedia, a DVD from Amazon, and a book also from Amazon. Possible queries for this data would be “list movies with "Emma Watson"”, or “retrieve information for movie "Harry Potter and the Chamber of Secrets"”. To answer these queries we need to know which data refer to the same real-world entity, for example if “Emma Watson” and “Watson, Emma (II)” are the same person. It is easy to see that such data may contain various real-world entities, such as people (i.e., starring, and writer), and movies. To simplify the example, for the

rest of the paper we consider only movie entities. Thus, we need to deal with three possible entities, one per movie, denoted with entity representations er_1, er_2, and er_3.

Information in our example is given as a set of triples with: (i) attribute (e.g., "author" of the book), (ii) value corresponding to this attribute (e.g., "J.K. Rowling" is the "author"), and (iii) probability reflecting the confidence for the specific attribute value pair. For example, the probability given by the used compression structure or the extraction tool. For the compression structure this probability would show the confident of the particular compression algorithm that this particular attribute value pair really exists in the specific entity representation. For extraction tools, the probability typically shows the relevance of this attribute value pair to the specific representation[2].

Even from the simple data of our example, we can see that identifying the real-world entities that this data describes is an essential task for the integration. A successful integration in of course is reflected in the results returned when querying the integrated data. Consider for example a query that requests all movies containing "fantasy" as "genre" and "Radcliffe, Daniel". Obviously, the result for this query should be the specific real-world entity (or entities) that correspond to the provided information. The system can therefore correctly answer this query only if it has a uniform view over the data, which basically shows the relationships between the data and the real world.

Identifying and linking the data that refer to the same real-world entity in not a straightforward process, since, many different aspects need to be considered. Among these aspects are the various attribute sets used in heterogeneous data (e.g. "starring", "actor"), and of course the use of name variants (e.g., "Emma Watson", "Watson, Emma (II)"). Another main aspect is that the entity linkage solution can never be finalized, because the data will be constantly changing and evolving through the addition of even more data.

One possible entity linkage solution for our example data states that er_1 should be linked to er_2 with probability P_1 and er_1 to er_3 with probability P_2. A simple merge of these probabilistic linkages with the original data is not possible, because it requires selection of the correct threshold for deciding whether a linkage with a specific probability is accepted, or even linkage verification through manual processing. To use this information we need to consider both the uncertainty and correlations in linkages. For instance, uncertainty in linkages would mean that we can have more than one possibilities for solutions, e.g., both linkages exist (i.e., $er_1 = er_2$ and $er_1 = er_3$), only one of the two exists (i.e., $er_1 = er_2$ or $er_1 = er_3$), or none of them exist. Of course, the possible solution can also be affected by correlations, e.g., existence of linkages $er_1 = er_2$ and $er_1 = er_3$ implies $er_2 = er_3$ and if this is not supported by our data, then the specific solution should not be accepted.

My Ph.D. thesis proposes to uniformly maintain entity related information describing the real-world entities and their possible linkages as detected by matching algorithms. This information is never merged with the original data, but it is used during query processing to provide results that reflect the real-world entities existing in the current data. As this document explains, this leads to a more effective management of the uncertainty and correlations that either exist in the original data, or are generated by the matching algorithms.

The rest of the paper is organized as follows. Section 2 explains the problem addressed in this thesis and its main challenges. Then, in Section 3 we discuss related work. Section 4 presents the topics which will be addressed by the PhD thesis, and Section 5 describes the planned experimental methodology. Finally, Section 6 presents the next steps in this work.

[2]OpenCalais is one of the extraction tool that generate such probabilities. More information and details are available at: http://www.opencalais.com/APIresponses#relevance

2. PROBLEM STATEMENT

Heterogeneous data, especially when integrated from different sources and applications, describe/refer the same real-world entities in a different manner (i.e., people, movies). Querying such data is expected to return results that reflect the corresponding real-world entities. The work in this Ph.D. thesis focuses on the efficient and effective entity-aware query processing in heterogeneous data with uncertainty and correlations.

In the following paragraphs we present the challenges for entity-aware query processing, followed by the main contributions of this Ph.D. thesis.

2.1 Challenges

There are three main issues related to entities in heterogeneous data. The first issue is that heterogeneous data typically contain various entity types. In our example (Section 1) we had data describing entities of type film, book, DVD, and also of people. The second issue is that the information about entities is typically a collection of data, such as a film which is described through its actors and title. This information can also have different types, e.g., with or without schema. The third issue is that the information about the entities will be constantly changing and evolving through the addition of more data. This means that the entity related information can never be finalized, and thus the data coming from the various sources can not be changed. As explained in the previous sections, a successful solution will need to process the original data and relate them to the real-world entities. The general challenge arising from the above issues is:

Challenge 1. ***(Uniform View over Entities)*** *Develop methodologies for managing the data that refer or describe the same real-world entity. This includes: (i) a generic data model that captures the generated entity related information, (ii) techniques for identifying the possible linkages between data that describe the same real-world entity in heterogeneous data, and (iii) effective handling of the uncertainty and correlations existing in the data.*

A major task when merging heterogeneous data is answering queries in a manner that reflect the data contributed by all sources. As shown by our example (Section 1), query processing is such heterogeneous data is not a straightforward process. We need to take into consideration the uncertainty and correlations found in both the original and the generated data (i.e., entity linkages). Given these issues, we arrive at the following challenge:

Challenge 2. ***(Query Processing)*** *Develop an efficient and effective methodology for processing queries over integrated heterogeneous data. Returned results should reflect the real-world entities represented by this data.*

Until now, we explained that we need to integrate heterogeneous data from various sources and also generate a uniform view over the real-world entities. For a number of external systems —such as the contributing sources and the query posting— provenance is quite important. In our case, provenance would provide information of the data origin, but also explanations for any generated information and returned results. The main challenge regarding this issue can be summarized as:

Challenge 3. ***(Reasoning: Uncertainty & Provenance)*** *Develop methodologies to fully support uncertainty and provenance, which are strongly encapsulated in both the original data and the unified view over entities.*

2.2 Thesis Contributions

This thesis proposes a new approach for enabling entity-aware query answering for heterogeneous data. In contrast to the existing approaches that aim at addressing this problem through matching algorithms, we follow an alternative and more suitable methodology. We consider as entity representations, all information coming from a source that possibly describes the same real-world entity, including possible differences in the data models such as the existence or not of schema information. Data given for integration are processed and the entity related information is generated. This also includes entity linkage which identifies which representations possibly referring to the same real-world entity. A uniform model is used to represent all generated entity related information. The information encoded in this uniform model is then used on query processing for providing entity-aware answers. Both entity linkage and query processing take into consideration the uncertainty and correlations found in the data. Provenance information is provided for explaining the origin the justification of the provided results.

3. STATE OF THE ART

This section presents and discusses related work. The first part of this section (Section 3.1) focuses on existing matching algorithms and the second part (Section 3.2) on related work for management data with uncertainty.

3.1 Data Matching/Linkage

Data matching is the process of identifying whether the entity representations describe/refer to the same real-world entities (e.g., same person, or location). Since acquisition of entity representations involves various tools (as explained in Section 1), representations typically have various types (e.g., only data values, data values with schema information) and come without any guarantees that they include all necessary information needed for a correct matching.

In the following paragraphs we discuss various existing matching algorithms grouped according to the information used during the matching. We begin with the basic similarity measures, then algorithms that use only the entity representations, and finally algorithms that also utilized available inner-relationships. A complete overview of the existing work in this domain can be found in surveys [15, 20, 19] and workshops/tutorials [29, 31].

Basic Similarity Measures

The first category includes methods that view entity representations as single text values, or bag of words, and thus perform a matching when these measures detect high resemblance between the corresponding text values of bag of words. This simplistic idea was found partly correct, since the real-world entities to which they refer might not be reflected in the given text values. For example consider two people with the exact same name. Using a similarity measure from this category would result in the incorrect matching of these people. For this reason, these measures are currently typically used only as part of the initial steps of more sophisticated matching approaches, to identify potential matches which are then further processed.

A large group of methods that belong to this category are the *Edit distance* methods, such as the Levenshtein distance, the Jaro [26], and the Jaro-Winkler [35] metrics, and the TF/IDF similarity [32]. These methods compute similarity based on number of the operations (e.g., delete character, insert character) for converting the first text value/bag of words to the second one. Other algorithms are the fuzzy matching similarity [10] which combines transformation operations with edit distance methods, and Soundex which phonetic encodings of words. [12, 8] describe and provide an experimental comparison of various basic similarity measures used for matching names.

Methods using entity representations

The methods of this category consider entity representations as collections of data (e.g., text values with/without schema) and try to use all available information during matching. Various examples of algorithms for this category have been suggested by the database community. One of the most known methods, is the merge-purge [24], aiming in identifying whether two relational records refer to the entity. Similarly, [34, 16] perform matching by discovering possible mappings from one entity representation to another.

Cohen et al. [11] uses methods from the basic similarity measures (previous category) to create techniques to adaptively modify the document similarity metrics. Li et al. [30] also focus in handling multiple types of entities, addressing the problem as this appears in the context of the text documents.

Methods using inner-relationships

The last group of methods is the ones that perform the matching when "links" are identified between entity representations. These links can be seen as inner-relationships, or associations between the representations (e.g., co-authorship of publications). These methods are more successful that the methods from previous categories, since the meaning encoded in the text values is not ignored.

To capture the links found inside the dataset, data are modeled into supportive structures. Ananthakrishna et al. [3] exploit dimensional hierarchies to detect fuzzy duplicates in dimensional tables. Hierarchies are built by following the links between the data from one table to data other tables. Entities are matched when the information along these generated hierarchies is found similar. Getoor et al. [6, 7] model the metadata as a graph structure, with nodes being information describing the entities and edges the relationships between the entities. Edges are used to cluster the nodes, and found clusters help to identify the common entities. In [28, 27], the dataset is also modeled as a graph following a similar methodology as the previous method. This method also generates other possible relationships (modeled as edges in the graph) to represent the candidate matches between entities. Then, graph theoretic techniques are used to analyze the relationships in the graph and decide the correct entity matches.

The TAP system [21] uses a process named *Semantic Negotiation* to identify common descriptions (if any) between the different resources. These common descriptions are used to create a unified view of the data. Benjelloun et al. [5] identify the different properties on which the efficiency of such algorithm depends on, and introduce different algorithms to address the possible combinations of the found properties. Another well-know algorithm is the *Reference Reconciliation* [17]. Here, the authors begin their computation by identifying possible associations between entities by comparing the corresponding entity descriptions. The information encoded in the found associations is propagated to the rest of the entities in order to enrich their information and improve the quality of final results.

3.2 Management of Uncertain Data

Another important aspect of our approach is management of un-

certainty in data; a topic that has received a lot of attention recently. [14] used the notion of possible worlds to introduce query semantics for independent probabilistic data and presented how to efficiently evaluate queries. The approach by Sen et. all [33] moved away from simplistic relations between data (e.g., independent data such as [14]) towards defining and using different correlations, for example existence of one tuple implies or disallows the existence of another tuple.

Few existing proposals focused on managing uncertainty in data integration. Dong et.all [18] used a semi-automatic schema mapping tool to generate the possible mappings between the attributes of the contributing sources with a mediated schema. The resulting mappings were probabilistic, and were used only upon query evaluation. The approach presented in [4] is more similar to ours, since the focus is not on duplication over the schema information but rather the actual data. The main differences of this approach with our suggestion are that they treat each tuple as an entity, they know which tuples correspond to the same entity, and they make simplistic assumptions about correlations between tuples. More specifically, the authors assume tuple independence which —as we explain and show through examples— is rarely the case when dealing with data matching.

Other related approaches are Dataspace [23] and Trio [2]. The main focus of these approaches is to create database systems that support uncertainty along with inconsistency and lineage. The part of these systems that is responsible for providing correct answers over uncertain data which represent duplicated tuples is closely related to one of the issues we are addressing. However, our approach goes beyond by addressing more challenges of heterogeneous data, but mainly because we consider matching on the data (not only on schema information), and also correlations between entities.

4. PROPOSED APPROACH

This thesis aims in overcoming the challenges discussed in the previous section. My approach suggests that data integration is accompanied with information to represent in a uniform manner the entities contained in the integrated data (called entity representations) and the possible matches between these entities (called entity linkages). Entity linkage information will be generated by new matching algorithms that focus on addressing the characteristics of heterogeneous data (Section 4.1). When integrating new data we also process it to generate and update this entity related information. During query processing this information is used in order to return results that confirm to the real-world entities (Section 4.2). Additionally, provenance information (Section 4.3) will be generated so that query processing can provide explanations on why the data appear as such in the generated information (i.e., entity representations and linkages). The following paragraphs present in more details the topics that I will concentrate on during this thesis.

4.1 Probabilistic Entity Linkage

We already worked on a new probabilistic entity linkage algorithm [25]. Our goal was addressing the matching problem as this appears when integrating heterogeneous data from Personal Information Management (PIM). PIM systems, such as NEPOMUK[3] and Haystack[4], integrate data coming from various extractors for desktop resources (e.g., publications, images) and internal application data (e.g., email clients). The problem in PIMs appears because each extractor describes entities in a way most adequate for its purpose. For example, a publication extractor will describe a person using name and affiliation, whereas an email extractor will use the email address.

One of the main problems for linking entities in such heterogeneous data is defining the conditions under which two entity representations should be linked. This is especially true in PIM data, since entity representations come from more than one data source and therefore they have different forms, e.g., no schema, different schema attributes, or various text formats. In addition, representations from PIM data have strong interconnections/correlations between them (e.g., people are related since they are co-authors in publications). Moreover, information provided about the real-world entities is constantly changing and evolving by the addition of data (e.g., new publication, a new email). This changes the information available for the matching algorithms, and imposes the support of incremental computation and adaptation of entity linkage information.

To address the above, our probabilistic entity linkage algorithm: (i) allows the incorporation of various evidences for matches, (ii) clearly separates the original data with data representing decisions for matches, (iii) enables incremental update of matches when new data become available, and (iv) associates each match with a probability indicating the belief (confidence) we have for the existence of the specific match according to the evidences currently in the data.

Entity Representation ($\mathcal{R}$)

id	attr.	value	p
er_1	-	Daniel Radcliffe	0.7
er_1	-	Emma Watson	0.4
er_1	-	J.K. Rowling	0.6
er_1	-	Fantasy	0.6
er_2	starring	Radcliffe, Daniel	
er_2	starring	Watson, Emma (II)	
er_3	author	J.K. Rowling	
er_3	genre	Fantasy	
...			

Entity Linkage ($\mathcal{L}$)

id	e	e	p
$l_{1,2}$	er_1	er_2	P_1
$l_{1,3}$	er_1	er_3	P_2
	...		
$l_{3,4}$	er_3	er_4	P_3
$l_{4,1}$	er_4	er_1	P_4
	...		
$l_{20,30}$	er_{20}	er_{30}	P_5
...			

Figure 2: Entity representation and linkage for that data from the example of Section 2.

Our algorithm uses a Bayesian network to model the possible matches, the evidences that prove the existence of these entities, and relationships them. Probabilistic inference is then used to compute the probability with which each match exists, according the relation of each entity with the relating information. This allows us to compute the probability for each possible match between data according to the evidences currently available in the system, and thus easily modify the network and the entity linkage solution when new information arrives. Figure 2 show the entity representation and linkage for our example from Section 1. As shown the linkages come with a probability and thus we can easily update it when new data arrive.

[3]http://nepomuk.semanticdesktop.org/
[4]http://haystack.lcs.mit.edu/

4.2 Query Processing

Query processing is an essential part of any integration system. In our approach, queries result in a list with the entities regardless of the format of the data that have been integrated in the system. A query for entities is a set of attribute value pairs, for example:

- **Q1:** author="J.K. Rowling"
- **Q2:** starring="Emma Watson", starring="Radcliffe, Daniel"

As explained earlier, query results must reflect the real-world entities found in the heterogeneous data. Providing these results is not a straightforward procedure. A simple merge of the generated information with the original data is quite problematic, since it requires verification of the linkages through manual processing, or the selection of the correct threshold for deciding whether a linkage with a specific probability must be accepted. Besides, it is possible that we have more than one linkage linking the same entity representations with different probabilities (generated by different matching algorithms), or even have conflicting linkages.

We are currently working on an approach that follows the possible world's semantics (from probabilistic databases) for using the entity representation along with the linkage information on query processing. The main goal of our approach is that query processing takes into account the different possible worlds that can be generated for the entity linkages, and thus returns each result with the probability that it truly corresponds to the real answer. As an example, consider only $l_{1,2}$ and $l_{1,3}$ entity linkages from Figure 2. Accepting or not the existence of $l_{1,2}$ and $l_{1,3}$ gives us the four possible worlds shown in Figure 3. Each world has its own probability to exist based on the probabilities of the entity linkage information that it contains.

$l_{1,2}$ ($er_1 = er_2$)	$l_{1,3}$ ($er_1 = er_3$)	Entities		
true	true	(1) $er_1 = er_2 = er_3$		
true	false	(1) $er_1 = er_2$	(2) er_3	
false	true	(1) $er_1 = er_3$	(2) er_2	
false	false	(1) er_1	(2) er_2	(3) er_3

Figure 3: Possible interpretations when having two entity linkages: $l_{1,2}$ stating that er_1 should be merged with er_2, and $l_{1,3}$ showing that er_1 should be merged with er_3.

Our data also have correlations, and thus we can only accept the possible worlds that do not violate the data correlations. For example, when both linkages exist we know that $er_1 = er_2$ and $er_1 = er_3$ exist. But this implies also that $er_2 = er_3$ exists and therefore we need to reject all worlds which specify that $er_2 = er_3$ does not exist. Also, upon "accepting" an entity linkage (e.g., $er_1 = er_2$) we need to merge the information from the corresponding representations. For this, we again need to take into consideration the possible correlations among the final information that we will include in the entities. Among others, we could have: (i) independent attributes which require only one value per attribute, or (ii) exclusive attributes which allow more than one value for the same attribute.

As illustrated in the above example, evaluation of queries should conform to the uncertainty and the correlations of the data. The main challenges we are now investigating are how uncertainty and correlations affect the possible worlds, and how to process the queries efficiently over all the possible worlds.

4.3 Provenance

From the time that the data are given for integration until the time they are returned as the result of a query, a number of "functions/transformations" is applied. This includes the processing of the data for integration purposes, the generation of the entity related information (from the matching algorithms), and the usage of the data during query processing. In addition to the successful final results, many tasks require also to know the origin of the data, namely the provenance information.

Provenance for our case should provide information on two levels. The first is provenance for the original heterogeneous data, such as the source that provided the data. The second level is for the resulted entity-aware data. For the latter, we consider information such as the data used for a specific entity representation, or the considerations that lead to a specific entity linkage.

Considering the above levels, we are planning to work on extending the uniform model for representing the entity related information for including provenance. This is not a trivial task since we need to uniformly model provenance for the information we generate for the entities, and for the original data. Then, we will extend the query language to allow querying for provenance and to adjust query processing.

5. EXPERIMENTAL EVALUATION

This section presents the planned experimental evaluation for the matching algorithms, entity-aware query processing, and provenance answering. It describes the selected datasets, and then the evaluation methodology.

5.1 Datasets & Methodology

Our goal is to use datasets collected from real applications, which truly capture the problems as these appear in real-world heterogeneous data. To achieve this, we are planning to use more than one datasets from different applications, with each dataset having different individual challenges. The following paragraphs present these datasets and the results of the performed evaluation.

Integrating Movie Data. In this dataset we assume a scenario in which we need to integrate information about movies coming from different applications. We collected 23.182 movies from *IMDb*[5] and 28.039 movies from *DBpedia*[6], with a total of 23.361 matches between them. From these sources we generate a number of small repositories containing information about movies. Due to the compression the tuples describing the entities arrive in the integration system with probabilities, which show the belief that the specific tuple belongs to the entity. **Characteristics:** This dataset allows us to evaluate our algorithms in a controlled environment, since (i) the real matches between all movie entities of the two systems are given, and (ii) realistic queries for movies can easily be generated from existing query logs. In addition, we can easily investigate the behavior of the algorithms under difference data uncertainty, since this is something we can control through modifying the configuration parameters of Bloom filters.

Entities from News Articles. The second dataset aims in integrating events from news articles that have been published in the web. The dataset contains a total of 300.000 news articles with ranking information from [13]. We will use an entity extractor system to retrieve the entities from the news articles and combine them with the rank of each article to generate the total uncertainty of each tuple. **Characteristics:** This dataset will be composed from data coming from a number of sources, and thus we will need to deal with higher data heterogeneity compared to the previous dataset.

[5]The Internet Movie Database, http://www.imdb.com
[6]The DBpedia Knowledge Base, http://www.dbpedia.org

Also, the data will contain various entity types (e.g., people, locations, events) and of course various relationships between them (e.g., an event will point to people or/and locations).

For the experiments we will evaluate the effectiveness and efficiency of suggested algorithms and —wherever possible— compare their results with similar approaches with the traditional/ordinary manner of handling these problems.

5.2 Entity Linkage Results

The next paragraphs present the evaluation we performed for our probabilistic entity linkage algorithm. The goal of this evaluation was to measure the quality of the entity linkages we generate. For this, we integrated information about the entities (i.e., publications) incrementally, and computed the entity linkages precision and recall under different number of publications.

For the evaluation we used the Cora dataset[7]. This is a collection of CiteSeer publications, in which information about the real-world entities have different formats while also having various inner-relationships. As explained in Section 4.1, our algorithm generates entity linkages with a probability that indicates the corresponding belief according to the current data (evidences). We used the ground truth provided by Cora dataset, and computed precision and recall when the belief (i.e., expressed through the probability) of the entity linkages was higher than 0.5, 0.6, 0.7, and 0.9.

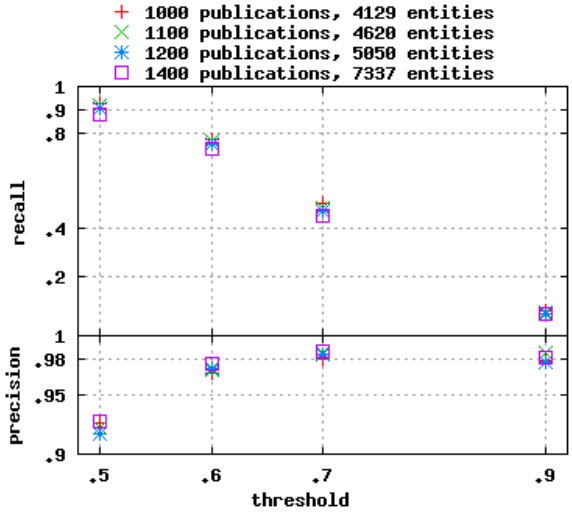


Figure 4: Precision and recall of our probabilistic entity linkage algorithm with linkage probability higher than .5, .6, .7, and .9.

Figure 4 shows the precision and recall of the entity linkages generated by our algorithm. As the plot shows, for a lower probability (i.e., 0.5) recall is very high and precision is already quite satisfactory (around 0.9). Higher probability thresholds (i.e., 0.6, 0.7) increase precision, and as expected, decrease recall. The same stable values of precision and recall we see for the different probability thresholds, show that the linkage probabilities do not just increase when adding more data. Our algorithm is able to adjust (re-evaluate) the linkage probabilities for reflecting the belief we have for each linkage given the current data.

6. NEXT STEPS

As described in the above sections, my Ph.D. thesis will propose a new approach for entity-aware query processing focusing on heterogeneous data with uncertainty and correlations. The following paragraphs present my next steps for completing the thesis.

Entity Linkage. Until now my work focused on providing an adequate entity linkage solution for heterogeneous data coming from Personal Information Management applications [25]. My plan is to improve this algorithm and performing experiments in more general scenarios, for example with unknown data schemes. In addition, I plan to investigate the creation of an algorithm for matching data those text values are not similar but rather alternatives (e.g., attribute "author" with "writer", or "Germany" with "Deutschland"). Being able to efficiently identify such linkages is still an open issue.

Query Processing. The main focus of my current work is using the entity representation and linkage information when processing queries for providing entity-aware answers. As explained, the main issues that should be addressed for a successful solution are (i) the handling of uncertainty and correlations that exist in heterogeneous data, and (ii) the efficient processing of queries.

Provenance. Once the algorithm for entity-aware query processing is completed, the approach will be extended to cover provenance information related to the possible linkage decisions and answers returned by querying.

Future Directions. More future tasks include the investigation of conflicting information in both the entity representation coming from the original data and the entity linkage information. This will allow us to apply our entity-aware query processing on far more complex datasets (e.g., data coming from Web pages).

7. WORKSHOP FEEDBACK

The PhD workshop provided me the opportunity to receive valuable feedback regarding the research directions as well as the approach I follow for addressing the identified problems.

In addition, the participation in the workshop and conference allowed me to get additional feedback for my PhD proposal. The discussion focused mainly on the aspects of uncertainty and heterogeneity as these appear in the data of various applications. This helped me to understand which other domains and systems could benefit of the approach proposed in this PhD.

8. ACKNOWLEDGMENTS

This work is partially supported by the FP7 EU Large-scale Integrating Project "OKKAM – Enabling the Web of Entities" (contract no. ICT-215032).

9. REFERENCES

[1] E. Adar and C. Re. Managing uncertainty in social networks. *IEEE Data Eng. Bull.*, 2007.

[2] P. Agrawal, O. Benjelloun, A. D. Sarma, C. Hayworth, S. U. Nabar, T. Sugihara, and J. Widom. Trio: A system for data, uncertainty, and lineage. In *VLDB*, 2006.

[3] R. Ananthakrishna, S. Chaudhuri, and V. Ganti. Eliminating fuzzy duplicates in data warehouses. In *VLDB*, 2002.

[4] P. Andritsos, A. Fuxman, and R. J. Miller. Clean answers over dirty databases: A probabilistic approach. In *ICDE*, 2006.

[5] O. Benjelloun, H. Garcia-Molina, D. Menestrina, Q. Su, S. E. Whang, J. Widomr, and J. Jonas. Swoosh: A generic approach to entity resolution. Technical report, Stanford InfoLab, 2006.

[6] I. Bhattacharya and L. Getoor. Deduplication and group detection using links. In *LinkKDD*, 2004.

[7] I. Bhattacharya and L. Getoor. Iterative record linkage for cleaning and integration. In *DMKD*, 2004.

[7] http://www.cs.umd.edu/ indrajit/ER/index.html

[8] M. Bilenko, R. J. Mooney, W. W. Cohen, P. Ravikumar, and S. E. Fienberg. Adaptive name matching in information integration. *IEEE Intelligent Systems*, 2003.

[9] F. Chang, J. Dean, S. Ghemawat, W. C. Hsieh, D. A. Wallach, M. Burrows, T. Chandra, A. Fikes, and R. Gruber. Bigtable: A distributed storage system for structured data. In *OSDI*, 2006.

[10] S. Chaudhuri, K. Ganjam, V. Ganti, and R. Motwani. Robust and efficient fuzzy match for online data cleaning. In *SIGMOD*, 2003.

[11] W. Cohen and J. Richman. Learning to match and cluster entity names. In *MF/IR*, 2001.

[12] W. W. Cohen, P. Ravikumar, and S. E. Fienberg. A comparison of string distance metrics for name-matching tasks. In *IIWeb*, 2003.

[13] G. M. D. Corso, A. Gulli, and F. Romani. Ranking a stream of news. In *WWW*, 2005.

[14] N. N. Dalvi and D. Suciu. Efficient query evaluation on probabilistic databases. 2004.

[15] A. Doan and A. Y. Halevy. Semantic integration research in the database community: A brief survey. *AI Magazine*, 2005.

[16] A. Doan, Y. Lu, Y. Lee, and J. Han. Object matching for information integration: A profiler-based approach. In *IIWeb*, 2003.

[17] X. Dong, A. Halevy, and J. Madhavan. Reference reconciliation in complex information spaces. In *SIGMOD*, 2005.

[18] X. L. Dong, A. Y. Halevy, and C. Yu. Data integration with uncertainty. In *VLDB*, 2007.

[19] A. K. Elmagarmid, P. G. Ipeirotis, and V. S. Verykios. Duplicate record detection: A survey. *IEEE Trans. Knowl. Data Eng.*, 2007.

[20] L. Getoor and C. P. Diehl. Link mining: a survey. *SIGKDD Explorations*, 2005.

[21] R. V. Guha and R. McCool. Tap: a semantic web platform. *Computer Networks*, 2003.

[22] R. Gupta and S. Sarawagi. Creating probabilistic databases from information extraction models. In *VLDB*, 2006.

[23] A. Y. Halevy, M. J. Franklin, and D. Maier. Principles of dataspace systems. In *PODS*, 2006.

[24] M. A. Hernández and S. J. Stolfo. Real-world data is dirty: Data cleansing and the merge/purge problem. *Data Min. Knowl. Discov.*, 1998.

[25] E. Ioannou, C. Niederée, and W. Nejdl. Probabilistic entity linkage for heterogeneous information spaces. In *CAiSE*, 2008.

[26] M. A. Jaro. Advances in record-linkage methodology as applied to matching the 1985 census of tampa, florida. *ASA*, 1989.

[27] D. V. Kalashnikov and S. Mehrotra. Domain-independent data cleaning via analysis of entity-relationship graph. *TODS*, 2006.

[28] D. V. Kalashnikov, S. Mehrotra, and Z. Chen. Exploiting relationships for domain-independent data cleaning. In *SIAM SDM*, 2005.

[29] N. Koudas, S. Sarawagi, and D. Srivastava. Record linkage: similarity measures and algorithms. In *SIGMOD*, 2006.

[30] X. Li, P. Morie, and D. Roth. Semantic integration in text: From ambiguous names to identifiable entities. *AI Magazine*, 2005.

[31] A. M. Ouksel and A. P. Sheth. Semantic interoperability in global information systems: A brief introduction to the research area and the special section. *SIGMOD*, 1999.

[32] G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA, 1986.

[33] P. Sen and A. Deshpande. Representing and querying correlated tuples in probabilistic databases. In *ICDE*, 2007.

[34] S. Tejada, C. A. Knoblock, and S. Minton. Learning domain-independent string transformation weights for high accuracy object identification. In *KDD*, 2002.

[35] W. Winkler. The state of record linkage and current research problems, 1999.