Associating Structured Records To Text Documents

Rakesh Agrawal Ariel Fuxman Anitha Kannan John Shafer Partha Talukdar*
Search Labs, Microsoft Research
Mountain View, CA, USA

ABSTRACT

Postulate two independently created data sources. The first contains text documents, each discussing one or a small number of objects. The second is a collection of structured records, each containing information about the characteristics of some objects. We present techniques for associating structured records to corresponding text documents and empirical results supporting the proposed techniques.

Categories and Subject Descriptors

H.2.8 [**Information Systems**]: Database Management
Database applications Subjects: Data mining

General Terms

Algorithms, Experimentation

Keywords

Matching, structured data, unstructured data

1. INTRODUCTION

We present a solution for associating records in a structured database with documents in a text corpus, such that records and documents are paired if they refer to the same object (*e.g.,* a person, business, or product). We make no assumptions regarding the organization of the two data sources. Each document is simply a sequence of words, and there is no categorization or structure that serves to identify the objects being discussed. Similarly, each structured record is nothing more than a set of key-value pairs (*e.g.,* <*color,blue*>) that serves to describe various characteristics of the object it represents. We do not require knowledge as to the significance of the characteristics or their role in distinguishing objects. As a result, our approach is general and is not restricted to, or specialized for, a particular domain.

Related Work. The research relevant to composing data from multiple sources can be categorized into three streams: i) identifying similar structured records, ii) linking text documents, and iii) matching structured records to text data. The last stream is the one closest to our work.

This stream of work includes the EROCS system [2]. However, EROCS requires manual input of object templates, one for each type of object. An object template defines how to uniquely determine the best match of the object if part or full context of that object is given. We did not want manual input of the object templates, which would need to be provided for every category of homogenous objects.

Dalvi *et. al.* [3] identify the object that is the topic of a review. They hypothesize a language model underlying the creation of reviews, which leads to a method for finding the object most likely to be the topic of a review. This work has been generalized in [4] to allow for attributes in structured records to have different weights and admit semantic translations of values. However, the success of this proposal again is dependent upon good pre-categorization of documents and structured records.

2. APPROACH

We observe that, in order to associate a document to its relevant objects, it suffices to look at the object *traits* that appear in the document. *Traits* are sets of attribute-value pairs that serve to distinguish objects, and are computed directly from the structured database. For example, we may determine from the structured database that the single attribute-value pair $T = <$*model,SD1300IS*$>$ is discriminative enough to determine that a particular record is about the camera "Canon SD1300IS". We would thus call T a trait. While traits might sound similar to keys in relational databases, they are fundamentally different, because they are instance-based rather than schema-based. For example, the "Fuji FinePix A400" may require the longer trait {<*model,A400*>, <*brand,Fuji*>} because <*model,A400*> is insufficient to distinguish it from the "Canon Powershot A400".

At the core of our implementation framework is a trait generation algorithm that draws upon a connection between our problem and that of finding infrequent itemsets [1]. Additionally, the algorithm uses a graphical model to identify unsafe attribute-value pairs that should not participate in a trait. To appreciate the need for this feature, consider the following example: for the camera "Olympus Stylus 600", the set {<*brand,Olympus*>, <*model,600*>} would appear to be a reasonable trait we might compute from the database. However, the term "600" may appear in many documents for reasons unrelated to a model number (*e.g.,* a price of 600 dollars). Such a trait could therefore lead to erroneously associating irrelevant documents to the Olympus camera. Note that there may still be other traits that could be used for mapping. In our example, one such trait is {<*brand,Olympus*>, <*line,Stylus*>, <*resolution,6mp*>}.

*Currently at Carnegie Mellon University; work done while on contract at Microsoft.

WWW 2012 Companion, April 16–20, 2012, Lyon, France.
ACM 978-1-4503-1230-1/12/04.

The presence of a particular trait in a document is itself a strong indicator that the document refers to the object that the trait represents. However, a document may contain traits belonging to several objects, and we should not assume that all objects are equally important. To handle this situation, our framework also includes a machine-learned scoring function that computes a probabilistic score of agreement between matched documents and objects by utilizing all the observed attribute-value pairs, not just those participating in traits. See [5] for further details.

3. EXPERIMENTAL EVALUATION

Experimental Setup. We experimented with five structured databases from the product search vertical of a search engine. Four of them contain information about TVs, cameras, computing products (computers, printers, hard drives), and household appliances (refrigerators, air conditioners, *etc*). The fifth (henceforth, "comprehensive database") contains the union of these databases plus products from additional electronic categories (*e.g.,* MP3 players, speakers, DVD players), totaling 32 categories.

The text corpus was obtained from a scan of upwards of a billion documents from the web index of the search engine. There were 10,884,689 documents in the index containing some value from one of the traits. From those, we randomly selected 200,000 documents. Note that the mere occurrence of a trait value does not imply that the document is about an object in the database.

We used the standard metrics of precision and recall to measure performance. For computing precision, the ground truth was obtained by labeling 200 documents. Each document t is paired with plausible records r, and the annotator is asked whether t is about the object represented by r. For computing recall, we found it was difficult for the annotator to identify if the text is about one of the objects in the structured database. Therefore, we computed the minimum recall (*i.e.*, actual recall may be higher) by having the annotator simply identify if the document belongs to any of the corresponding product categories.

Results. Figure 1 shows the precision and recall for the comprehensive database. At a recall of 0.1, we get a precision of 0.7 (dotted curve). A drill-down revealed that many of the incorrect associations were due to the "accessory problem": that is, the system was associating a page about an accessory of a product p to the record for p. For example, a page about a Lexmark printer cartridge was associated to the record for the "Lexmark P6250 multifunction printer". In general, this error happens when the structured database does not contain records corresponding to the accessories, but includes records for their main products. In fact, we found that there were far too many pages about product accessories in our text corpora but not many structured records about them.

To isolate the impact of the "accessory problem", we experimented with another truth set that consisted of the same 200 test documents, but the association $< t, r >$ was considered correct if the text t was about r or about an accessory of the product represented by r. Call this truth set "Accessories-Good" and the original truth set "Accessories-Bad". We observe in Figure 1 that with the Accessories-Good ground truth set, we achieve much higher precision for the same level of recall.

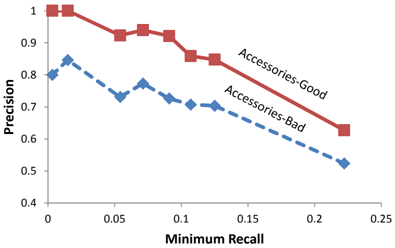

Figure 1: Precision and minimum recall (actual recall may be higher) for comprehensive database

	Precision	
	Accessories-Bad	Accessories-Good
Cameras	0.82	1.00
Computing products	0.68	0.85
Household Appliances	0.93	0.93
Televisions	0.84	0.89
Comprehensive database	0.73	0.92

Table 1: Precision for the five databases

We also measured precision for the other four databases. For each database, the results are from a sample of 100 labeled documents. Table 1 shows precision numbers using the scoring threshold that corresponds to 0.73 and 0.92 precision in the comprehensive database using the Accessories-Bad and Accessories-Good ground truth sets, respectively. The highest precision (0.93) in both the cases was obtained with household appliances, a category where there were hardly any accessories sold separately from the main product. For computing products, the precision jumps from 0.68 to 0.85 if we do not penalize for accessories; for cameras, it jumps from 0.82 to 1. These are the categories where accessories are prevalent (*e.g.,* ink cartridges for printers or lenses for cameras). Televisions have relatively fewer accessories and we accordingly see smaller increase in precision.

4. FUTURE WORK

We plan to develop a deeper understanding of the performance characteristics of the proposed approach using generative models for documents and objects. We are hoping that such a study will also lead to developing techniques for estimating the performance of the composition without having to resort to using human judgments. We also plan to apply our framework to a number of applications including web search by augmenting the web index with structured data to improve relevance and enrich search snippets.

5. REFERENCES

[1] E. Boros, V. Gurvich, L. Khachiyan, and K. Makino. On maximal frequent and minimal infrequent sets in binary matrices. *Annals of Mathematics and Artificial Intelligence*, 39:211–221, 2003.

[2] V. Chakaravarthy, H. Gupta, P. Roy, and M. Mohania. Efficiently linking text documents with relevant structured information. In *VLDB*, pages 667–678, 2006.

[3] N. Dalvi, R. Kumar, B. Pang, and A. Tomkins. Matching reviews to objects using a language model. In *EMNLP*, pages 609–618, 2009.

[4] N. Dalvi, R. Kumar, B. Pang, and A. Tomkins. A translation model for matching reviews to objects. In *CIKM*, pages 167–176, 2009.

[5] R.Agrawal, A.Fuxman, A.Kannan, Q.Lu, and J.Shafer. Composing text and structured databases. Technical Report MSR-TR-2012-22, Microsoft Research, 2012.