WebDB 2010
13th International Workshop on the Web and Databases
June 6, 2010
Indianapolis, Indiana
U.S.A.

Reviews For Paper

Paper ID 15

Title Position-Based Ranking for Entity-Relationship Queries

Masked Reviewer ID: Assigned_Reviewer_1

Review:

Question	
Overall Recommendation	Weak Reject
Reviewer confidence	Medium (I understand this paper pretty well but I don't know the context or background of the problem.)
Novelty of the proposed problem (Does the paper propose new topics or stimulating ideas?)	Novel
Quality of the solution (Does the paper propose a solution that is effective or has high potential?)	Solved in some way
Organization and presentation. (Is the proposed solution clearly developed?)"	Readable with some difficulty
Summary of Main Contributions	This paper addresses positional ranking in order to answer structured queries over the web (so-called entity-relationship queries).
Strong points and weak points of the paper	* The paper addresses a grand challenge problem. We all dream of being able to have a system that can answer the motivating example from the introduction. * The positional ranking scheme seemed to make sense, although I really

	did not understand all the details. Are there other schemes? Why this? How relevant is it really with regard to the grand vision of answering queries like the motivating example.
Detailed Comments	Main comment: After reading the introduction (which got me all excited), I was somewhat disappointed by the rest of the paper. Yes, ranking is important, but somehow there seem to be so many other issues that need to be addressed (how to derive types in a general way, how to index for positional ranking to name just two). It seems that the really heavy weight-lifting is not addressed in the paper; in particular, it seems that the really heavy weight-lifting has not been addressed in the prototype system: I would have been eager to hear how all that has been done. (Ranking comes somewhat later on the list of the things I want to hear about, even though I know how important it is once the other problems have been solved.) Detailed comments * I think "entity-relationship" queries is a misnomer. "entity-relationship" typically refers to the ER model (by Chen) which is something totally different. * Query answers: do the tuples contain strings or URIs? Wouldn't URIs be a better way to represent concepts such as "Yahoo!"?

Masked Reviewer ID: Assigned_Reviewer_2

Review:

Question	
Overall Recommendation	Weak Reject
Reviewer confidence	Expert (I understand deeply what matters in this area.)
Novelty of the proposed problem (Does the paper propose new topics or stimulating ideas?)	Novel
Quality of the solution (Does the paper propose a	Solved in some way

solution that is effective or has high potential?)	
Organization and presentation. (Is the proposed solution clearly developed?)"	Readable with some difficulty
Summary of Main Contributions	Authors propose a query processing algorithm for Select Project Join Queries on top of labeled Wikipedia documents. The corpus is segmented in sentences. Entities in the corpus are labeled (here by Wikipedia users). The proposed algorithm should compute at runtime the likelihood that semantics of relationships expressed in the SQL query match the syntax of word- and entity sequences in labeled sentences. To prove their findings authors present preliminary theoretical work to rank sentences for a query and verify their findings in a user study.
Strong points and weak points of the paper	Strong points: • Executing structured queries across documents is a novel and relevant topic. To achive this goal, authors combine relevant research from the areas of natural language processing and query processing across structured data. • Besides their theoretical and experimental findings, authors propose a web-demo to verify their experimental results from the paper. • The mentioned related work is relevant. For instance, it contains recent work on open information extraction methods. Weak points: • Although authors conducted many experiments and provide multiple examples, it seems that their findings are not general applicable for multiple reasons: o Analysts that write a SQL query are required to know the keywords (aka the language model) that explicitly indicate the desired relation in the text; here in Wikipedia sentences. Given that humans already face problems with formulating keyword queries against a web search engine I doubt if a human will find distinctive and unambiguous keyword-phrases for retrieving documents that contain a complex relation without support by a machine. Since the corpus is known before query time I suggest to learn a language model for each relation and offer the user most relevant phrases from the a-priori learned model. o Besides the examples given in the paper and on the web page I tried out several simple queries but could not verify the high precision numbers given in the paper. Please verify if your system provides good results for the following simple queries: ☐ Company:IBM; City: New York; relation: “headquarter” ☐ Person: “Tom Cruise”; Person “Katie Holmes” ☐ Person: “Tom Cruise”; Person “Nicole Kidman” ☐ (I also tried the these queries with the relation “married” ☐ Company: SAP Company:BOBJ relation:”acquires”

	☐ Company: SAP Company:BOBJ relation:"merges" Please note: These queries are answered by the opponent system textrunner.
Detailed Comments	To improve the precision you should careful think what other context information could be considered to rank results. I suggest syntactic information (as done by the system textrunner) or lexicographic information (as done by the system Avatar). Please also state how you address the problems of o Multiple relations that are stated in multiple sentences (anaphor resolution) o Diathesis of voices, e.g., the order of S-P-O structures in English language is effected by the active or passive voice. o Compound and complex, nested sentences structures. The structure and the spelling of the paper should be improved. o Abstract: Previous works -> work o Related work should be an own Section o [6]learns -> Authors in [6] propose a method to learn o Section 2 Your example could not be verified o Example 2: Should the first entity be "Stanford University" instead of "Stanford"? o Section 2.2.: The order of entities in a sentence is language specific and fact specific. o Page 4: Patterns from more prominent entities are more effective. That is a novel observation. Can you prove it?

Masked Reviewer ID: Assigned_Reviewer_3

Review:

Question	
Overall Recommendation	Weak Accept
Reviewer confidence	Expert (I understand deeply what matters in this area.)
Novelty of the proposed problem (Does the paper propose new topics or stimulating ideas?)	Novel
Quality of the solution (Does the paper	Solved in some way

propose a solution that is effective or has high potential?)	
Organization and presentation. (Is the proposed solution clearly developed?)"	Adequately organized and written
Summary of Main Contributions	The paper deals with the processing of entity-relationship queries, which blend concepts from structured database query languages and processing an text based information retrieval.
Strong points and weak points of the paper	The paper is extremely well written. The addressed problem is very relevant and of increasing importance, as more and more exploratory search over web and linked data is becoming popular. Technically, the approach is simple but also interesting. Queries can be processed on the fly, using quite consolidated relevance measures already used from search engines. The approach is implemented and an online demo permits one to play with ER queries However, the paper (and the associated implementation) has also some problems. Having tried out the system, I had the feeling that precision is still very far away from a minimum acceptable threshold. The sensitivity to the query wording is very high: if very close synonyms are used the results can change dramatically. I assume that some elementary query expansion (e.g., using Wordnet synsets) could improve the precision very much. Another impression is that false positives tend to rank quite high, and that true positives can be found in lower positions in the ranking. The authors claim this as "accuracy beyond top few", but in the end it turns into bad search experience.
Detailed Comments	Overall, I consider the paper interesting for WebDB, even if more research would grant better results. An interesting area of evolution could also be the architecture. I think there are interesting optimization problems in the way entities and relationships are evaluated from the corpus, based on the expected selectivity of the predicates input by the user.