Finding Relevant Relations in Relevant Documents

Michael Schuhmacher[1], Benjamin Roth[2], Simone Paolo Ponzetto[1], and Laura Dietz[1]

[1] Data and Web Science Group, University of Mannheim, Germany
firstname@informatik.uni-mannheim.de
[2] College of Information and Computer Science, University of Massachusetts, Amherst, USA
beroth@cs.umass.edu

Abstract. This work studies the combination of a document retrieval and a relation extraction system for the purpose of identifying query-relevant relational facts. On the TREC Web collection, we assess extracted facts separately for correctness and relevance. Despite some TREC topics not being covered by the relation schema, we find that this approach reveals relevant facts, and in particular those not yet known in the knowledge base DBpedia. The study confirms that mention frequency, document relevance, and entity relevance are useful indicators for fact relevance. Still, the task remains an open research problem.

1 Introduction

Constructing knowledge bases from text documents is a well-studied task in the field of Natural Language Processing [3,5, *inter alia*]. In this work, we view task of *constructing query-specific knowledge bases* from an IR perspective, where a knowledge base of relational facts is to be extracted in response to a user information need. The goal is to extract, select, and present the relevant information directly in a structured and machine readable format for deeper analysis of the topic. We focus on the following task:

Task: Given a query Q, use the documents from a large collection of Web documents to extract binary facts, i.e., subject–predicate–object triples (S,P,O) between entities S and O with relation type P that are both correctly extracted from the documents' text and relevant for the query Q.

For example, a user who wants to know about the Raspberry Pi computer should be provided with a knowledge base that includes the fact that its inventor Eben Upton founded the Raspberry Pi Foundation, that he went to Cambridge University, which is located in the United Kingdom, and so on. This knowledge base should include all relational facts about entities that are of interest when understanding the topic according to a given relation schema, e.g., `Raspberry_Pi_Foundation`–*founded_by*–`Eben_Upton`. Figure 1 gives an example of such a query-specific resource, and shows how relations from text and those from a knowledge base (DBpedia, [1]) complement each other.

In addition to a benchmark dataset,[3] we present first experiments on building query-specific knowledge bases from a large-scale Web corpus by combining state-of-the-art retrieval models with a state-of-the-art relation extraction system [7]. This way we

[3] Dataset and additional information is available at http://relrels.dwslab.de.

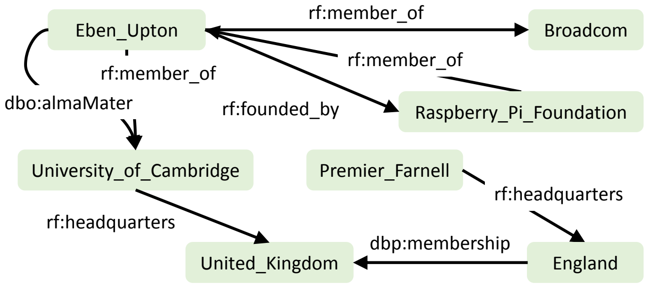

Fig. 1. Example of a knowledge base for the query “raspberry pi”. `rf:` denotes relations extracted from documents, whereas `dbp:` and `dbo:` are predicates from DBpedia.

go beyond previous work on identifying relevant entities for Web queries [8] (where relations between entities were not considered), and query-agnostic knowledge base population (where determining fact relevance is not part of the task).

We aim at quantifying how well the direct application of a relation extraction system to a set of retrieved documents solves the task of extracting query-specific facts. This is different from the task of explaining relationships between entities in a knowledge base [9], since we include also yet unknown facts from documents. It is also different from explaining the relationship between entities and ad-hoc queries [2], since we look at relations between entities in documents. To isolate different kinds of errors, we evaluate the *correctness* of each fact extraction separately from the *relevance* of the fact for the query. We study the following research questions:

RQ1 Can the approach extract relevant facts for the queries?
RQ2 What are useful document- or KB-based indicators for fact relevance?
RQ3 Is relevance of entities and relevance of facts related?

2 Method

Document retrieval. We use the Galago[4] search engine to retrieve documents D from the given corpus that are relevant for the query Q. We build upon the work of Dalton et al. [4] and rely on the same document pool and state-of-the-art content-based retrieval and expansion models, namely the sequential dependence model (SDM), the SDM model with query expansion through RM3 (SDM-RM3), and the SDM model with query expansion through the top-ranked Wikipedia article (WikiRM1).

Relation extraction. A prerequisite for running the relation extraction system is to identify candidate sentences that mention two entities S and O. We use the FACC1 collection of entity links [6]. We identify all sentences in retrieved documents that contain at least two canonical entities in Freebase with types `/people`, `/organization`, or `/location` as candidates for relation extraction. Finally, we use RelationFactory,[5] the top-ranked system in the TAC KBP 2013 Slot filling task, to extract facts (S,P,O) from candidate sentences of the retrieved documents.

[4] `http://lemurproject.org/galago.php`
[5] `https://github.com/beroth/relationfactory`

3 Data Set and Assessments

To our knowledge, there exists no test collection for evaluating relational facts with respect to query-relevance. We augment existing test collections for document-relevance and entity-relevance with assessments on correctness and query-relevance of facts; and make the dataset publicly available.We base our analysis on the collection of test queries from the TREC Web track and documents from the ClueWeb12 corpus, which includes relevance assessments for documents, and the REWQ gold standard of query-relevant entities [8].[6]

The TREC Web track studies queries which fall into one of two categories: the query either constitutes an entity which is neither a person, organization, nor location i.e., "Raspberry Pi", or the query is about a concept or entity in a particular context, such as "Nicolas Cage movies". The closed relation extraction system only extracts relation types involving persons, organizations and locations. Due to this restriction, not all TREC Web queries can be addressed by relations in this schema, this is the case for TREC query 223 "Cannelini beans". We focus this study on the subset of 40% of TREC Web queries such as "Raspberry Pi" for which anticipated relevant facts are covered by the relation schema.

For randomly selected 17 TREC queries, we assess the 40 most frequently mentioned facts and, in addition, all facts of which at least one of the entities was marked as relevant in the REWQ dataset. Due to the high number of annotations needed—914 facts and 2,658 provenance sentences were assessed in total—each item was inspected by only one annotator. We ask annotators to assess for each fact, a) the correctness of the extraction from provenance sentences and b) the relevance of the fact for the query. To assess relevance, assessors are asked to imagine writing an encyclopedic (i.e., Wikipedia-like) article about the query and mark the facts as relevant if they would mention them in the article, and non-relevant otherwise.

The number of provenance sentences per fact ranges from 1 to 82 with an average of 2.9. We define facts as correct when at least one extraction is correct. This leads to 453 out of 914 facts that are correctly extracted. The fact extraction correctness is thus at 49.6%, which is higher than the precision obtained in the TAC KBP shared task, where about 42.5% of extractions are correct. The assessment of relevance is performed on these 453 correctly extracted facts, leading to a dataset with 207 relevant facts and 246 non-relevant facts across all 17 queries, an average of 26.6 relevant facts per query. In this study we only consider queries with at least five correctly extracted facts (yielding 17 queries).

4 Evaluation

We evaluate here how well the pipeline of document retrieval and relation extraction performs for finding query-relevant facts. The *relevance* is separately evaluated from *extraction correctness*, as described in Section 3. In the following, we focus only on the 453 correctly extracted facts. For comparing different settings, we test statistical

[6] `http://rewq.dwslab.de`

Table 1. Experimental results for relation relevance (correctly extracted relations only) comparing different fact retrieval features: All facts (All), facts also included in DBpedia (DBp), fact mentioned three or more times (Frq$_{\geq 3}$), facts extracted from a relevant document (Doc). Significant accuracy improvements over "All" marked with †.

		All	Frq$_{\geq 3}$	DBp	Doc
Per Query (macro-avg)	#Queries	17	10	17	10
	Precision	0.470	0.553	0.455	0.704
	Std Error	0.070	0.100	0.087	0.112
All Facts (micro-avg)	#Retrieved Facts	453	106	145	46
	TP	207	58	64	30
	FP	246	48	81	16
	TN	-	198	165	230
	FN	-	149	143	177
	Precision	0.457	0.547	0.441	0.652
	Recall	1.000	0.280	0.309	0.145
	F_1	0.627	0.371	0.364	0.237
	Accuracy	0.457	†0.565	0.506	†0.574

significant improvements on the accuracy measure through a two-sided exact binomial test on label agreements ($\alpha = 5\%$).

Applicability (RQ1). We report the results on fact relevance as micro-average across all facts (Table 1 bottom) and aggregated macro-averages per query (Table 1 top) to account for differences across queries. Among all correct facts, only every other fact is relevant for the query (0.45 micro-average precision, 0.47 macro-average precision). Factoring in the extraction precision of 0.51 we obtain one relevant out of four extracted facts on average. This strongly suggest that the problem of relevant relation finding (beyond correctness) is indeed an *open research problem*.

In about 60% of TREC queries, such as "Cannelini beans", we found the relation schema of TAC KBP to not be applicable. Nevertheless, even with the schema limitations, the system found relevant facts for the (randomly) assessed 17 queries out of the remaining 40 queries.

Indicators for fact relevance (RQ2). We study several indicators that may improve the prediction of fact relevance. First, we confirm that the frequency of fact mentions indicates fact relevance. If we classify a correctly extracted fact as 'relevant' only when it is mentioned at least three times[7] then relevance accuracy is improved by 23.6% from 0.457 to 0.565 (statistically significant). This also reduces the number of predicted facts to a fourth (see Table 1, column *Frq*$_{\geq 3}$).

Next, we compare the extracted facts with facts known to the large general-purpose knowledge base DBpedia. When classifying only extracted facts as relevant when they are confirmed—that is, both entities are related in DBpedia (independent of the relation type)—we do not obtain any significant improvements in accuracy or precision. Therefore, confirmation of a known fact in an external knowledge base does not indicate relevance. However, we notice that only 64 of the relevant facts are included in

[7] We chose ≥ 3 in order to be above the median of the number of sentences per fact, which is 2.

Table 2. Fact relevance when at least one entity ($S \vee O$) or both entities ($S \wedge O$) are relevant compared to all facts (All). Significant accuracy improvements over "All" marked with †.

	All	S ∨ O	S ∧ O
#Retrieved Facts	108	94	49
TP	78	76	45
FP	30	18	4
TN	-	12	26
FN	-	2	33
Precision	0.722	0.809	0.918
Recall	1.000	0.974	0.577
F_1	0.839	0.884	0.709
Accuracy	0.722	†0.815	0.657

DBpedia, whereas another 143 new and relevant facts are extracted from the document-centric approach (cf. Table 1, column *DBp*). This indicates that extracting yet unknown relations (i.e., those not found in the knowledge base) from query-relevant text has the potential to provide the majority of relevant facts to the query-specific knowledge base.

Our study relies on a document retrieval system, leading to some non-relevant documents in the result list. We confirm that the accuracy of relation relevance improves significantly when we only consider documents assessed as relevant. However, it comes at the cost of retaining only a tenth of the facts (cf. Table 1, column *Doc*).

Fact relevance vs. entity relevance (RQ3). We finally explore whether query-relevance of entities implies relevance of facts. In order to study this implication, we make use of the REWQ test collection on entity relevance [8] by studying the subset of the 108 correct facts where relevance assessments exist for both entities. Due to pooling strategies, this subset has a higher precision of 0.722. In Table 2 we consider the case where entity relevance is true for both entities (S ∧ O) as well as at least one entity (S ∨ O).

For only 12 correct facts, both entities are assessed as non-relevant – these facts were also assessed as non-relevant by our (different) annotators. In contrast, for 45 facts both entities and the fact itself are assessed as relevant (we take this agreement also as a confirmation of the quality of our fact assessments). Using the entity assessments as an oracle for simulating a classifier, we obtain improvements in precision from 0.722 to 0.809 for either entity and 0.918 for both entities. While also accuracy improves for the case of either entity, it is actually much lower in the case of both entities. We conclude that the restriction to both entities being relevant misses 33 out of 78 relevant facts. In this set of 33 relevant facts with one relevant and one non-relevant entity, we find that the non-relevant entity is often too unspecific to be directly relevant for the query such as a country or city. For example, in Figure 1 the `University_of_Cambridge` is relevant mostly because of the fact that `Eben_Upton` is a member.

5 Conclusion

We investigate the idea of extracting query relevant facts from text documents to create query-specific knowledge bases. Our study combines publicly available data sets and state-of-the-art systems for document retrieval and relation extraction to answer research questions on the interplay between relevant documents and relational facts for this task. We can summarize our key findings as follows:

(a) Query-specific documents contain relevant facts, but even with perfect extractions, only around half of the facts are actually relevant with respect to the query.
(b) Many relevant facts are not contained in a wide-coverage knowledge base like DBpedia, suggesting importance of extraction for query-specific knowledge bases.
(c) Improving retrieval precision of documents increases the ratio of relevant facts significantly, but sufficient recall is required for appropriate coverage.
(d) Facts that are relevant can contain entities (typically in object position) that are—by themselves—not directly relevant.

From a practical perspective, we conclude that the combination of document retrieval and relation extraction is a suitable approach to query-driven knowledge base construction, but it remains an open research problem. For further advances, we recommend to explore the potential of integrating document retrieval and relation extraction—as opposed to simply applying them sequentially in the pipeline architecture.

Acknowledgements

This work was in part funded by the Deutsche Forschungsgemeinschaft within the JOIN-T project (research grant PO 1900/1-1), in part by DARPA under agreement number FA8750-13-2-0020, through the Elitepostdoc program of the BW-Stiftung, an Amazon AWS grant in education, and by the Center for Intelligent Information Retrieval. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor. We are also thankful for the support of Amina Kadry and the helpful comments of the anonymous reviewers.

References

1. Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: DBpedia – A Crystallization Point for the Web of Data. Journal of Web Semantics 7(3) (2009)
2. Blanco, R., Zaragoza, H.: Finding support sentences for entities. In: Proc. of SIGIR-10. pp. 339–346 (2010)
3. Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka, E.R., Mitchell, T.M.: Toward an architecture for never-ending language learning. In: Proc. of AAAI-10. pp. 1306–1313 (2010)
4. Dalton, J., Dietz, L., Allan, J.: Entity query feature expansion using knowledge base links. In: Proc. of SIGIR-14. pp. 365–374 (2014)
5. Fader, A., Soderland, S., Etzioni, O.: Identifying relations for open information extraction. In: Proc. of EMNLP-11. pp. 1535–1545 (2011)
6. Gabrilovich, E., Ringgaard, M., Subramanya, A.: FACC1: Freebase annotation of ClueWeb corpora, Version 1 (2013)
7. Roth, B., Barth, T., Chrupała, G., Gropp, M., Klakow, D.: Relationfactory: A fast, modular and effective system for knowledge base population. In: Proc. of EACL-14. p. 89 (2014)
8. Schuhmacher, M., Dietz, L., Paolo Ponzetto, S.: Ranking Entities for Web Queries through Text and Knowledge. In: Proc. of CIKM'15 (2015)
9. Voskarides, N., Meij, E., Tsagkias, M., de Rijke, M., Weerkamp, W.: Learning to Explain Entity Relationships in Knowledge Graphs. In: Proc. of ACL-15. pp. 564–574 (2015)