Example Based Entity Search in the Web of Data

Marc Bron[1], Krisztian Balog[2], and Maarten de Rijke[1]

[1] ISLA, University of Amsterdam, Scienc Park 904, 1098 XH Amsterdam
[2] University of Stavanger, NO-4036 Stavanger
m.m.bron@uva.nl, krisztian.balog@uis.no, derijke@uva.nl

Abstract. The scale of today's Web of Data motivates the use of keyword search-based approaches to entity-oriented search tasks in addition to traditional structure-based approaches, which require users to have knowledge of the underlying schema. We propose an alternative structure-based approach that makes use of example entities and compare its effectiveness with a text-based approach in the context of an entity list completion task. We find that both the text and structure-based approaches are effective in retrieving relevant entities, but that they find different sets of entities. Additionally, we find that the performance of the structure-based approach is dependent on the quality and number of example entities given. We experiment with a number of hybrid techniques that balance between the two approaches and find that a method that uses the example entities to determine the weights of approaches in the combination on a per query basis is most effective.

1 Introduction

In entity search, entities are returned to the user instead of documents [2]. An increasing number of entity-oriented search tasks have been proposed recently. The INEX Entity Ranking track provided an evaluation platform for entity ranking and entity list completion using semi-structured data (Wikipedia) [15]. The TREC Entity track introduced the task of related entity finding in an unstructured web corpus [3, 9]. At the core of these tasks, systems need to find entities that engage in a certain relation.

The Linking Open Data (LOD) cloud is part of an interconnected Web of Data (WoD) that contains information about relations between objects. The WoD is formed by connections between a multitude of knowledge bases and information repositories [5]. This type of structured data has the potential to be helpful in entity-oriented search tasks since a large part of the WoD revolves around entities and their relations [6].

The scale of today's Web of Data motivates the use of text-based approaches [21, 23, 24] to retrieve information about entities in addition to traditional structure-based approaches. Text-based approaches make limited use of the available structure and instead focus on text associated with objects. From a user's point of view it is easier to specify keyword queries than to issue a structure-based query, e.g., using SPARQL,[1] which requires knowledge of the underlying schema. An alternative to using keyword queries is to allow users to submit examples of entities they are searching for. The structural information associated with example entities then provides input for structure-based

[1] http://www.w3.org/TR/rdf-sparql-query

P. Serdyukov et al. (Eds.): ECIR 2013, LNCS 7814, pp. 392–403, 2013.

methods. Possible scenarios for users to obtain examples are to use keyword queries to retrieve examples from an initial result set or to use a schema browser allowing a user to wander from one entity to the next until one or more examples are found [30].

In this paper we look into the challenge of utilizing examples for retrieving entities that engage in a certain relation with other entities. We look at a text-based approach, a structure-based approach that uses examples, and combinations of these two methods. The context in which we evaluate our methods is modeled after the Entity List Completion (ELC) task as seen at various evaluation platforms: INEX [15], the Semantic Search Challenge[2] and TREC [3]. We define the task as follows: *given a query (Q) consisting of a relation (R) and example entities (X), complete the list of examples by finding URIs of entities that join in the specified relation.*

In this setting we aim to answer the following research questions: (i) is a structure-based method that uses examples competitive when compared against a text-based approach; (ii) does the performance of text- and structure-based methods depend on the quality and the number of examples that are given; and (iii) can a hybrid method automatically balance between the two approaches in a query-dependent manner?

2 Related Work

A traditional way of accessing Linked Data is through structured query languages such as SPARQL, that express queries through constraints on relations (*links*) between URIs. These languages, however, are difficult to use and require knowledge of the underlying ontologies. More recent user-oriented approaches address this issue by automatically mapping keyword queries to structured queries [29, 33]. A number of services provide keyword based interfaces to search in Linked Data for URIs of entities [6]. Other approaches use keyword queries against a free text index of Linked Data [25, 30].

Hybrid approaches to ranking entity URIs exploit the link structure and textual information contained in Linked Data. For example, one approach returns both URIs that contain query terms as well as URIs that link to those URIs [26]. Yet others propose a combination of structured and keyword-based retrieval methods [2, 14]. Common to the text-based and hybrid approaches mentioned here is their focus on retrieving URIs for entities given a name or a description.

A hybrid method able to retrieve entities that engage in a certain relation with another entity is proposed by Elbassuoni et al. [16]. This method uses a language modeling approach to construct exact, relaxed, and keyword augmented graph pattern queries. In order to estimate the language models, RDF triple occurrence counts and co-occurring keywords are extracted from a free text corpus. We do not consider an outside corpus as the size of the Linked Data sample we use would require a very large web corpus to obtain reliable estimates and leave this as future work.

As a first step towards evaluating these semantic search approaches, the Semantic Search Workshop launched the "ad-hoc object retrieval task" [8, 25] focused on retrieving URIs for entities described by free text. Ad-hoc object retrieval differs from entity list completion in its focus on resolving entity names to URIs in the LOD cloud, instead of locating entities that stand in some specified relation. Examples of approaches to this

[2] `http://semsearch.yahoo.com`

task are to use a linear combination of the language model scores for different textual entity representations and a variant of the BM25F model that takes into account various statistics of the attributes in entity representations [7]. A hybrid approach that combines inverted indexes with automatically generated structure-based queries turns out to be the most effective approach, outperforming a BM25 model by up to 25% [28].

We investigate a different task, i.e., Entity List Completion (ELC) previously studied in a semi-structured setting, i.e., list completion using Wikipedia at INEX [15]. Approaches to this task retrieve Wikipedia documents, i.e., articles representing entities, using both text-based methods and methods based on the Wikipedia category structure [1]. One existing method combines the two approaches using a linear combination, where the mixing parameter depends on the difficulty of a topic [31]. A model is trained to predict each topic's difficulty and the combination weight is set accordingly. We also propose a query dependent method that combines text-based retrieval with additional structure, but differ from supervised machine learning based approaches [27] in that our method does not require any training data. Moreover, machine learning based approaches do not necessarily outperform unsupervised approaches in this setting [19].

The TREC Entity track's variation on the ELC task extends the INEX ELC task in that entities are no longer Wikipedia pages but URIs in a sample of the LOD cloud. Approaches to this task where evaluated on a limited (8) number of topics. They include a text-based [17] method using a filtering approach based on WordNet and link-based methods [10, 13] using link overlap and set expansion techniques.

In the entity list completion task of the 2011 Semantic Search Challenge entities are represented by URIs as well, but no example entities are given and only a textual description of the common relation between the target entities is provided. Approaches to this task are predominantly text-based [4]. A notable exception is an approach that re-ranks an initially retrieved list of entities using spread-activation [11].

There are other unsupervised approaches to combining results, also known as late data fusion, that use different ways of weighting the scores from various result lists [18]. These, however, do not exploit features other than those available in the result lists, i.e., they do not consider example entities.

3 Task and Approach

We define the *entity list completion* (ELC) task as follows: given a query (Q) consisting of (i) a textual representation for the relation (R) and (ii) a URI based representation for the example entities (X), complete the list of examples by finding URIs of entities that join in the specified relation; see Table 1 for an example topic. The data we consider for this task consists of a sample of the LOD cloud. Linked Data is typically represented using the RDF format[3] and defines relations between objects in the form of triples. An RDF triple consists of a subject, a predicate, and an object. A subject is always a URI and represents a "thing" (in our case: an entity), such as Michael Schumacher in Table 2. Subject URIs serve as unique identifiers for entities. An object is either a URI referring to another "thing" or a string (attribute), holding a literal value. Predicates are also always URIs and specify the relations between subjects and objects.

[3] http://www.w3.org/RDF/

Table 1. An example ELC test topic description

query Q	R :	Apollo astronauts who walked on the Moon
	X :	dbpedia:Buzz_Aldrin dbpedia:Neil_Armstrong

Text-Based Approach. There are two choices to be considered in designing a text-based approach to entity finding in Linked Data: (i) the representation of entities and (ii) the retrieval model. A popular approach to representing entities is to group all triples that have the same URI as subject together [4, 12, 21]. We follow [7, 23, 24] and use a fielded representation where triples associated with an entity are grouped into a small set of predefined categories. We consider the following three categories: (i) *attributes*, i.e., triples that have a string as object; (ii) *types*, i.e., triples for which the predicate is one of a predefined set of common predicates to indicate type information (`/22-rdf-syntax-ns#type, /core#subject, /subject`); and (iii) *links*, i.e, triples that have another node as object and are not of the *types* category. The objects of the links and types categories are URIs. This results in an entity representation as shown in Table 2. To obtain a meaningful textual representation we expand these URIs with the text associated with an object through the `/rdfs:label` predicate, which is widely used to provide a natural language description for Linked Data objects.

For the retrieval model, we adopt a language modeling approach because of its probabilistic foundations and effectiveness in entity-oriented search tasks [12, 16, 23]. In this framework we rank document representations of entities (e) based on the probability of being relevant to the relation (R) as specified in a query (Q): $P(e|R)$. We apply Bayes' rule to reformulate this to $P(R|e)P(e)/P(R)$ and drop the denominator $P(R)$ as it does not influence the ranking. For the entity prior, $P(e)$, we assume a uniform distribution. We model the entity document representation e as a Dirichlet smoothed multinomial distribution over terms (θ_e) that captures the probability of the entity model generating the terms in R: $P(R|\theta_e)$. By further assuming that terms are generated independently we obtain $P(R|\theta_e)$ as the product over the terms in the relation: $P(R|\theta_e) = \prod_{t \in R} P(t|\theta_e)$. What remains is to estimate the probability of a term t given the Dirichlet smoothed language model. We follow the standard language modeling approach [32] and estimate $P(t|\theta_e)$ as:

$$P(t|\theta_e) = \frac{tf(t,e)+\mu P(t|\theta_c)}{|e|+\mu},$$

Table 2. An example of the three entity representations: attributes, types, and links

	subject dbpedia.org/resource/Michael_Schumacher	
	property	object
attributes	dbpedia.org/property/shortDescription	Formula1 driver, 7 times world champion
	www.w3.org/rdf-schema#label	Michael Schumacher
types	www.w3.org/22-rdf-syntax-ns#type	umbel.org/umbel/rc/Athlete
	purl.org/dc/terms/subject	yago:GermanFormulaOneDrivers
links	dbpedia.org/ontology/fastestDriver_of	dbpedia:1998_British_GP

where $tf(t, e)$ is the term frequency of t in the representation document of e, $|e|$ is the number of terms in the entity representation, and $P(t|\theta_c)$ is the Dirichlet smoothed model of the entire collection of triples. To obtain a ranking for different entity representations, we estimate $P(t|\theta_e^{cs})$ for each category subset (cs), where θ_e^{cs} is a multinomial distribution estimated over the terms occurring in the triples of a category subset cs.

Previous work on ad-hoc entity search has shown that a linear mixture of the representation language models is effective [23]. We follow this approach and re-estimate the probability of a term given the weighted representation language models as follows:

$$P(t|\theta_e^w) = \textstyle\sum_{cs \in \{tp, lk, at\}} P(t|\theta_e^{cs}) P(cs),$$

where $P(cs)$ is the weight given to a specific representation model, i.e., types (tp), links (lk), and attributes (at). The probability of the weighted text-based model then becomes: $P(R|\theta_e) = \prod_{t \in R} P(t|\theta_e^w)$.

Using Examples with a Structure-Based Approach. An alternative to the text-based approach is to represent an entity by the links it has to other entities. Taking an entity URI as starting point we consider all RDF triples that have that URI as subject (i.e., outlinks) or object (i.e., inlinks). Together, these triples form the link based representation of an entity ($e_l = \{tr_1, \ldots, tr_m\}$, where tr_i is an RDF triple).

Under this representation, entities consist of sets of triples. The set of example entities becomes a set of sets of triples ($X = \{x_1, \ldots, x_n\}$ and $x_i = \{tr_1, \ldots, tr_k\}$). We rank entities according to the probability of the entity's link based representation e_l given a set of example entities X: $P(e_l|X)$. To incorporate the intuition that triples with the same predicate-object pair observed with more examples are more important than others, we expand this term to incorporate the triples tr explicitly: $P(e_l, tr|X)$. By assuming independence between the examples and the entity given the triples we can factorize this probability as follows: $P(e_l|tr)P(tr|X)$. Taking X to be a multinomial distribution over relations, θ_X, and marginalizing over the relations observed with the examples we obtain:

$$P(e_l|\theta_X) = \textstyle\sum_{tr \in \bigcup_{x \in X}} P(e_l|tr) P(tr|\theta_X),$$

where $\bigcup_{x \in X}$ is the union of the triples associated with each example. We estimate $P(tr|\theta_X)$ as follows:

$$P(tr|\theta_X) = \frac{\sum_{x \in X} n(tr, x)}{\sum_{tr' \in \bigcup_{x \in X}} \sum_{x \in X} n(tr', x)}.$$

Here, $n(tr, x)$ is 1 if tr occurs in the representation of example x and 0 otherwise. For $P(e_l|tr)$ we use a function which is 1 if tr occurs in the context of e_l and 0 otherwise.

Combining Approaches. Merging and learning to rank methods that combine various ranked lists have gained in popularity. We experiment with two unsupervised versions of such combination methods: (i) we employ a linear combination of the normalized similarity scores of the text and structure-based method; and (ii) we make use of the example entities to choose between the text-based approach, the structure-based approach, or a combination of these two approaches.

In the linear combination approach we use the parameter λ to control the weight assigned to the structure and text-based methods as follows:

$$P_{cmb}(e|Q) = \lambda \cdot P(e|\theta_X) + (1 - \lambda) \cdot P(R|\theta_e),$$

where Q consists of the relation R and the set of examples X.

Our second, alternative method is to predict the effectiveness of the text-based and structure-based techniques by capitalizing on the availability of explicit relevance feedback in the form of example entities. This *switch* method chooses between the text-based and structure-based method depending on which method is better able to retrieve the example entities. If both methods achieve similar performance, the linear combination method is used. We formalize this method as follows: given two ranked lists, one produced by the text-based method for a query ($L_{P(R|\theta_e)}$) and one produced by using the examples with the structure-based ($L_{P(e|\theta_X)}$), we use the example entities as relevance judgements and calculate the average precision (AP) for each of the lists. Based on the difference between the AP scores, λ is set to 0, to 1, or to the same value as in the linear combination method:

$$P_{switch}(e|Q) = \begin{cases} P(e|\theta_X) & \text{if overlap} < \gamma \\ & \text{and AP}(L_{P(e|\theta_X)}) > \text{AP}(L_{P(R|\theta_e)}) \\ P(R|\theta_e) & \text{if overlap} < \gamma \\ & \text{and AP}(L_{P(e|\theta_X)}) < \text{AP}(L_{P(R|\theta_e)}) \\ \lambda \cdot P(e|\theta_X) + & \\ (1 - \lambda) \cdot P(R|\theta_e) & \text{otherwise,} \end{cases} \tag{1}$$

where overlap is defined as:

$$\text{overlap} = \frac{\min(\text{AP}(L_{P(R|\theta_e)}), \text{AP}(L_{P(e|\theta_X)}))}{\max(\text{AP}(L_{P(R|\theta_e)}), \text{AP}(L_{P(e|\theta_X)}))},$$

and γ is a threshold parameter that determines how much the performance of the two methods is allowed to overlap, before one is chosen over the other. In case both methods have similar performance, a combination of both methods is used; otherwise, the best performing method is picked. Note that we focus on establishing a solid baseline for a pure text-based method and do not use examples, e.g., through relevance feedback.

4 Experimental Setup

The dataset in our experiments is the Billion Triple Challenge 2009 (BTC2009) data set.[4] We use three sets of topics for evaluation. The first set consists of the 50 semantic search challenge list completion task topics (SemSearch'11). This task was conducted on the BTC2009 data set and the evaluation data (qrels) with relevant URIs for each topic have been made available. In this specific setting no explicit examples are provided, only the desired relation that the target entities should satisfy is specified. The relevance judgements are graded on a relevance scale of 0 to 2. We consider URIs

[4] http://km.aifb.kit.edu/projects/btc-2009/

Table 3. Results of text-based language modeling (LM) approaches using different subsets of RDF triples as entity representation: only attributes, only triples containing type information, only triples linking to other nodes, all triples, and a weighted combination of the representations

	SemSearch'11				INEX'07				INEX'08			
	map	Rprec	rel_ret	rel	map	Rprec	rel_ret	rel	map	Rprec	rel_ret	rel
LM attributes	.0726	.1096	**193**	650	.0497	.0699	40	432	.0173	.0330	82	849
LM links	.0854	.1028	169	650	**.0746**	.0673	**76**	432	.0670	.0816	186	849
LM types	**.0891**	**.1176**	144	650	.0651	**.0821**	67	432	**.0816**	**.0922**	**197**	849
LM all	.1311	.1488	247	650	.0713	.0942	58	432	.0298	.0537	152	849
LM combine	**.1632**	**.1935**	**270**	650	**.1187**	**.1370**	**93**	432	**.0898**	**.1073**	**217**	849

judged as either relevant (2) or somewhat relevant (1) the same in our experimental setting as 454 of the 650 judgements are considered somewhat relevant.

In addition, we convert the original INEX'07 and INEX'08 topics to conform to the semantic search setting. INEX topics contain a description similar to the semantic search topic relation (R), e.g., *I want a list of the state capitals of the United States of America*. The topic further contains example entities, e.g., *Lincoln, Nebraska*. In the original INEX entity list completion task the goal is to retrieve entities from Wikipedia. The evaluation data also consists only of titles of Wikipedia pages. We combined several approaches to create an initial mapping of Wikipedia entities (pages) to DBpedia URIs [20, 22, 24] and refined this mapping through manual inspection.[5] The examples provided with each topic were added to the evaluation data. This results in a set of 25 and 35 topics with 423 and 849 URIs judged as relevant, respectively. We use the official TREC evaluation measures: R-precision (Rprec), Mean Average Precision (MAP) and number of relevant URIs returned (rel_ret). Results list are evaluated till rank 100.

In order to obtain example entities we randomly sample relevant entities for each topic from the evaluation data. In our experiments we select 10 random samples for each setting of our *number of examples* parameter as we increase the number of examples provided to the structure-based method. In order to make a fair comparison between methods we remove the sampled examples from the evaluation data. This procedure generates a different evaluation data set each time a different set of examples is selected.

5 Results

We first consider the results of our text-based approach. Table 3 shows the results of the language modeling (LM) approach on different subsets of RDF triples as entity representation. We find that of the representations that use a subset of triples associated with an entity the type representation generally outperforms the other representations in terms of MAP and Rprec. This is in line with our expectations as at the INEX Entity Ranking track treating type information as a special field was a popular approach [1, 31]. We observe that when using all triples as entity representation, precision and recall improve over using any subset of triples as representation for the SemSearch'11 data set and that results decrease for both INEX data sets. The best performance is achieved

[5] See `http://ilps.science.uva.nl/ecir2013elc` for topics and ground truth.

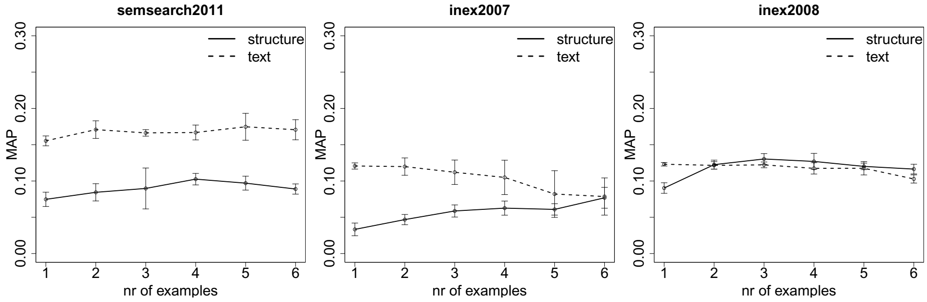

Fig. 1. The average MAP and standard deviation achieved by the text-based method (dotted line) and the structure-based method (solid line)

with a weighted combination of the different representations. The weights for each of the representations are set to the same values across the three data sets, i.e., to 0.4 for the *attributes*, 0.2 for the *links*, and 0.4 for the *types* entity representation.

For the evaluation of the text-based method we use the verbatim evaluation data with all entities included. This allows us to compare our results to those obtained at the 2011 Semantic Search Challenge. We find that our implementation of the text-based approach is able to reproduce these results, e.g., the highest pure text-based approach achieved a MAP of 0.1625.[6] Higher performance is achieved by approaches that re-rank an initial ranked list based on the link structure between top ranked entities. We focus on a pure text-based approach as baseline in order to analyze the individual contributions of text- and structure-based methods.

Results Using Examples with a Structure-Based Approach. We now consider whether the number of examples influences performance, how the structure-based method compares to the text-based method, and how performance varies with the quality of the examples. The solid line in Fig. 1 shows the mean and standard deviation of MAP achieved by the structure-based method over 10 samples for different numbers of examples for the INEX and SemSearch data sets. The dotted line shows the mean and standard deviation of MAP achieved by the text-based method. Note that as the evaluation data changes with every sample and that the results here are not directly comparable to those in Table 3. We observe that on the INEX'07 and SemSearch'11 topics the text-based approach outperforms the structure-based approach, while on the INEX'08 data set comparable performance is achieved. On the INEX'07 data performance of the text-based method decreases as the number of examples increases, but this phenomena is not observed on the other topic sets. Performance of the structure-based method increases on all three topic sets when the number of examples is increased and levels off when more than 4 examples are provided. With more examples the structure-based method is better able to determine the importance of triples in the example set but as more examples are added this results in diminishing returns.

Regarding the standard deviation of MAP scores achieved by the structure-based method we observe no obvious pattern and performance of the structure-based method

[6] `http://semsearch.yahoo.com/results.php#`

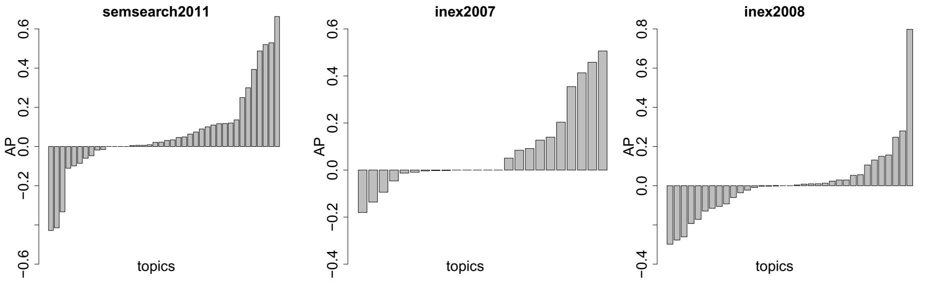

Fig. 2. Barplot of the difference in AP achieved by each topic. A negative value indicates that the structure-based method achieves better AP for that topic than the text-based method. A positive value indicates that the text-based method performs better.

does not become more or less robust as more examples are added. The performance of the text-based method also varies, this as a consequence of sampling entities and removing them from the evaluation data. This variation in performance suggests that the text-based method is dependent on a particular set of entities being relevant.

Next we take a closer look at the per query performance of the text and structure-based methods. Fig. 2 shows the difference in Average Precision (AP) achieved by the two methods per topic. A positive value indicates that the text-based method is more effective and a negative value indicates that the structure-based method achieves higher AP. The run on which these differences are based uses two examples and was further picked at random. We observe that the text-based method achieves a higher AP on more topics than the structure-based method on the INEX'07 and SemSearch'11 topics. On the INEX'08 topics there is no clear winner. We find that a considerable number of topics exists on which the structure-based method outperforms the text-based method. These results suggest that the text-based and structure-based methods work well on different queries and sets of example entities, motivating the use of a hybrid method.

Combined Approaches. A standard approach to combine structured information with a text-based approach is to use a linear combination ($P_{comb}(e|Q)$), where the contribution of each method is governed by a parameter (λ). To investigate the potential of this approach we perform a sweep, i.e., initialize λ form 0 to 1 with steps of 0.1, and find the optimal setting of λ over the number of examples: 0.1. For the switch method ($P_{switch}(e|Q)$) we likewise set γ to the optimal value (0.0 for INEX'07, 0.1 for INEX'08, and 0.0 for SemSearch'11) and we use the same λ as for the linear combination. When γ is set to 0 the switch method decides to mix if there is any overlap in performance between the two methods and otherwise uses the method that was able to return the examples. Note that using optimal settings allows us to investigate how the performance of text- and structure-based methods relate under ideal conditions. We leave an investigation of parameter sensitivity as future work. Fig. 3 shows the average and standard deviation of the MAP achieved by the linear combination method (dashed black line) and the switch method (dotted black line). We observe that on all three topic sets the performance of the switch method increases when the number of examples provided increases. In contrast, the performance of the linear combination method

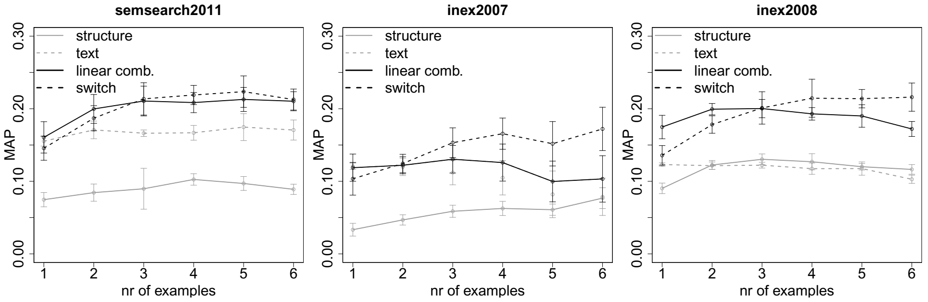

Fig. 3. Average and standard deviation of the MAP achieved by the linear combination method (solid black line) and the switch method (dotted black line). The structure-based method (solid grey line) and text-based method (dotted grey line) are added for comparison.

decreases when more examples are provided. When providing 3 or more examples the switch method outperforms the linear combination on each data set. On the INEX'07 dataset using 3 or more examples results in significantly ($\alpha = .05$) better performance in terms of MAP compared to the other three methods. On the INEX'08 dataset the same holds when using 4 or more examples. On the SemSearch'11 dataset we find no significant difference between the linear combination and switch methods, however, both significantly outperform the individual methods when using more than 1 example.

These results confirm our earlier observation that the text and structure-based methods return different sets of entities and are effective for different topics. The switch method is able to use the examples to determine which of these two methods will be most effective. The linear combination method performs initially better but is not able to utilize the information provided by the structure-based method. This has implications for such methods in a scenario where users may provide any combination of example entities and are no longer interested in re-finding them.

We observe that the variance for the linear combination and switch method increases compared to the structure-based approach. The methods become more sensitive to the specific examples that are available. This adds another challenge to using examples for entity search, i.e., how to asses the quality of the examples provided to our methods.

6 Conclusion

In this paper we have investigated the use of examples within a structure-based approach for entity search in the Web of Data. We found that depending on the number and quality of the examples, a structure-based approach achieves comparable performance to a competitive text-based approach. Through a per topic analysis, however, we find that each method returns different sets of entities, motivating the use of a hybrid approach. We have performed an analysis of the performance of two hybrid methods on repeated samples of example entities and relevance judgements. Results showed that a standard linear combination approach is suboptimal when the set of examples and entities considered relevant changes. This has consequences for the applicability of linear combination approaches in scenarios where a user provides examples, i.e., the particular set of entities the text-based method is effective in finding may overlap with the examples. We found that a hybrid method that uses example entities to determine whether to use a text-based, structure-based, or linear combination approach, outperforms a standard linear combination. We have also found that the variance in the performance achieved by both hybrid methods increases over the text-based and structure-based methods based on the specific set of examples provided. This suggests that a new direction in using examples for entity search lies in assessing the quality of examples provided.

In future work we plan to look into more sophisticated approaches to combining text and structural information for entity search in Linked Data. Specifically, text-based methods that incorporate structure in the form of spread activation and supervised learning to rank methods, and to investigate their sensitivity to varying sets of examples.

Acknowledgements. This research was partially supported by the European Union's ICT Policy Support Programme as part of the Competitiveness and Innovation Framework Programme, CIP ICT-PSP under grant agreement nr 250430, the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreements nr 258191 (PROMISE Network of Excellence) and 288024 (LiMoSINe project), the Netherlands Organisation for Scientific Research (NWO) under project nrs 612.061.-814, 612.061.815, 640.004.802, 727.011.005, 612.001.116, HOR-11-10, the Center for Creation, Content and Technology (CCCT), the WAHSP and BILAND projects funded by the CLARIN-nl program, the Dutch national program COMMIT, by the ESF Research Network Program ELIAS, and Elite Network Shifts project funded by the Royal Dutch Academy of Sciences (KNAW).

References

[1] Balog, K., Bron, M., de Rijke, M., Weerkamp, W.: Combining Term-Based and Category-Based Representations for Entity Search. In: Geva, S., Kamps, J., Trotman, A. (eds.) INEX 2009. LNCS, vol. 6203, pp. 265–272. Springer, Heidelberg (2010)

[2] Balog, K., Meij, E., de Rijke, M.: Entity search: building bridges between two worlds. In: Semantic Search Workshop 2010, pp. 1–5 (2010)

[3] Balog, K., Serdyukov, P., de Vries, A.: Overview of the TREC 2010 Entity Track. In: TREC 2010 (2010)

[4] Balog, K., Ciglan, M., Neumayer, R., Wei, W., Nørvåg, K.: NTNU at SemSearch 2011. In: Semantic Search Workshop 2011 (2011)

[5] Berners-Lee, T., Hendler, J., Lassila, O., et al.: The semantic web. Scientific American 284(5), 28–37 (2001)

[6] Bizer, C., Heath, T., Berners-Lee, T.: Linked data-the story so far. International Journal on Semantic Web and Information Systems 5(3), 1–22 (2009)

[7] Blanco, R., Halpin, H., Herzig, D., Mika, P., Pound, J., Thompson, H.: Entity search evaluation over structured web data. In: Workshop on Entity-Oriented Search 2011 (2011)

[8] Blanco, R., Mika, P., Vigna, S.: Effective and Efficient Entity Search in RDF Data. In: Aroyo, L., Welty, C., Alani, H., Taylor, J., Bernstein, A., Kagal, L., Noy, N., Blomqvist, E. (eds.) ISWC 2011, Part I. LNCS, vol. 7031, pp. 83–97. Springer, Heidelberg (2011)

[9] Bron, M., Balog, K., de Rijke, M.: Ranking related entities: Components and analyses. In: CIKM 2010 (2010)

[10] Bron, M., He, J., Hofmann, K., Meij, E., de Rijke, M., Tsagkias, M., Weerkamp, W.: The University of Amsterdam at TREC 2010: Session, entity and relevance Feedback. In: TREC 2010 (2011)

[11] Ciglan, M., Nørvåg, K., Hluchỳ, L.: The SemSets model for ad-hoc semantic list search. In: WWW 2012, pp. 131–140 (2012)

[12] Dalton, J., Huston, S.: Semantic entity retrieval using web queries over structured RDF data. In: Semantic Search Workshop 2010 (2010)

[13] Dalvi, B., Callan, J., Cohen, W.: Entity list completion using set expansion techniques. In: TREC 2010 (2011)

[14] Davies, J., Weeks, R.: QuizRDF: search technology for the semantic web. In: HICSS 2004 (2004)

[15] Demartini, G., Iofciu, T., de Vries, A.: Overview of the INEX 2009 entity ranking track. Focused Retrieval and Evaluation, 254–264 (2010)

[16] Elbassuoni, S., Ramanath, M., Schenkel, R., Sydow, M., Weikum, G.: Language-model-based ranking for queries on rdf-graphs. In: CIKM 2009, pp. 977–986 (2009)

[17] Fang, Y., Si, L., Somasundaram, N., Al-Ansari, S., Yu, Z., Xian, Y.: Purdue at TREC 2010 Entity Track: a Probabilistic Framework for Matching Types between Candidate and Target Entities. In: TREC 2010 (2011)

[18] Fox, E., Shaw, J.: Combination of multiple searches. In: TREC 1994, p. 243 (1994)

[19] Gao, J., Wu, Q., Burges, C., Svore, K., Su, Y., Khan, N., Shah, S., Zhou, H.: Model adaptation via model interpolation and boosting for web search ranking. In: EMNLP 2009, pp. 505–513 (2009)

[20] He, J., de Rijke, M., Sevenster, M., van Ommering, R., Qian, Y.: Generating links to background knowledge: A case study using narrative radiology reports. In: CIKM 2011 (2011)

[21] Liu, X., Fang, H.: A study of entity search in semantic search workshop. In: Semantic Search Workshop 2010 (2010)

[22] Meij, E., Weerkamp, W., de Rijke, M.: Adding semantics to microblog posts. In: WSDM 2012, pp. 563–572. ACM (2012)

[23] Neumayer, R., Balog, K., Nørvåg, K.: On the modeling of entities for ad-hoc entity search in the web of data. In: Advances in Information Retrieval, pp. 133–145 (2012)

[24] Pérez-Agüera, J., Arroyo, J., Greenberg, J., Iglesias, J., Fresno, V.: Using BM25F for semantic search. In: Semantic Search Workshop 2010 (2010)

[25] Pound, J., Mika, P., Zaragoza, H.: Ad-hoc Object Ranking in the Web of Data. In: WWW 2010 (2010)

[26] Rocha, C., Schwabe, D., Aragao, M.: A hybrid approach for searching in the semantic web. In: WWW 2004, pp. 374–383 (2004)

[27] Sheldon, D., Shokouhi, M., Szummer, M., Craswell, N.: LambdaMerge: merging the results of query reformulations. In: WSDM 2011, pp. 795–804 (2011)

[28] Tonon, A., Demartini, G., Cudré-Mauroux, P.: Combining inverted indices and structured search for ad-hoc object retrieval. In: SIGIR 2012, pp. 125–134 (2012)

[29] Tran, T., Wang, H., Rudolph, S., Cimiano, P.: Top-k exploration of query candidates for efficient keyword search on graph-shaped (RDF) data. In: ICDE 2009, pp. 405–416 (2009)

[30] Tummarello, G., Cyganiak, R., Catasta, M., Danielczyk, S., Delbru, R., Decker, S.: Sig. ma: live views on the web of data. In: Web Semantics: Science, Services and Agents on the World Wide Web (2010)

[31] Vercoustre, A., Pehcevski, J., Naumovski, V.: Topic difficulty prediction in entity ranking. In: Advances in Focused Retrieval, pp. 280–291 (2009)

[32] Zhai, C.: Statistical language models for information retrieval a critical review. Foundations and Trends in Information Retrieval 2(3), 137–213 (2008)

[33] Zhou, Q., Wang, C., Xiong, M., Wang, H., Yu, Y.: SPARK: Adapting Keyword Query to Semantic Search. In: Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L.J.B., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) ISWC/ASWC 2007. LNCS, vol. 4825, pp. 694–707. Springer, Heidelberg (2007)