Query Performance Prediction for Entity Retrieval

Hadas Raviv, Oren Kurland
Faculty of Industrial Engineering and Management, Technion, Haifa 32000, Israel
hadasrv@tx.technion.ac.il,kurland@ie.technion.ac.il

David Carmel
Yahoo! Labs, Haifa 31905, Israel
david.carmel@ymail.com

ABSTRACT

We address the query-performance-prediction task for entity retrieval; that is, retrieval effectiveness is estimated with no relevance judgements. First we show how to adapt state-of-the-art query-performance predictors proposed for document retrieval to the entity retrieval domain. We then present a novel predictor that is based on the cluster hypothesis. Evaluation performed with the INEX entity ranking track collections shows that our predictor can often outperform the most effective predictors we experimented with.

Categories and Subject Descriptors: H.3.3 [Information Search and Retrieval]: Retrieval models

Keywords: entity retrieval, query performance prediction

1. INTRODUCTION

Recently, it has been observed that for many user queries, named entities such as people, organizations and locations, better satisfy the user's information need than full documents [10]. Accordingly, there is a growing body of work on entity retrieval which deals with ranking entities by their presumed relevance to a query.

Entities are somewhat complex objects which are characterized by different properties such as name, type and associated document. Several entity retrieval methods that exploit these properties have been proposed [1, 11].

Here we address the *query-performance-prediction* (QPP) task for entity retrieval. The goal is to estimate, without relevance judgements, the effectiveness of retrieval performed in response to a query. While there is a large body of work on QPP for document retrieval [2], there has been very little work on QPP for entity retrieval [9]. Yet, the same motivation that triggered the development of predictors for document retrieval holds for entity retrieval. For example, alerts for ineffective retrieval can direct users to better formulate their queries.

We present a study of adapting state-of-the-art query-performance predictors, proposed for document retrieval, to the entity retrieval domain. In addition, we present a novel query-performance predictor for entity retrieval. The predictor relies on retrieval scores of clustered entities, following a study of the cluster hypothesis for entity retrieval [12]. Evaluation performed with the INEX entity ranking track collections shows that our novel predictor can often outperform the most effective predictors we experimented with.

SIGIR'14, July 6–11, 2014, Gold Coast, Queensland, Australia.

http://dx.doi.org/10.1145/2600428.2609519.

2. RELATED WORK

Query-performance prediction methods proposed for document retrieval can be categorized to two groups [2]. Pre-retrieval predictors analyze the query using corpus-based term statistics [7]. Post-retrieval predictors analyze also the result list of top-retrieved documents [2]. We adapt the most effective of these predictors to the entity retrieval domain.

To the best of our knowledge, there is a single report of work on QPP for entity retrieval [9]. The entity-list completion task was addressed where examples of relevant entities are provided. The most effective predictors used the description and narrative of the (INEX) topic as well as information induced from the example entities. In contrast, we address the entity ranking task, and the predictors we study do not use entity feedback (i.e., examples) nor the topic's narrative and description. We show that post-retrieval predictors outperform pre-retrieval predictors, which was not the case in this work [9] that did not adapt state-of-the-art predictors proposed for document retrieval.

3. QPP FOR ENTITY RETRIEVAL

Our focus is on predicting retrieval performance for queries whose goal is finding entities of a particular type or class [10]. We use the datasets of the INEX *entity ranking track* [4, 5]. Each entity in the corpus is represented as a Wikipedia page associated with a set of categories which serve as the entity's type. The entity ranking task queries are composed of a short keyword-based title and a set of categories representing the query's target type. Entities relevant to the query are expected to be associated with categories in the query's target type, or with categories that are "close" to those in the target type in the Wikipedia category graph.

Most entity retrieval methods utilize several properties of entities [14, 1, 11]. Typical properties are the document associated with the entity (the Wikipedia page in our case), the entity type (the set of categories associated with the entity in our case), the entity name (the Wikipedia page title), etc. Accordingly, we study prediction methods that use information induced from two properties which were found to be highly effective for retrieval [1, 11]; namely, the document associated with the entity and the entity type.

Specifically, the prediction methods that we present use three entity representations. The first is **doc**, under which an entity is represented by its associated document. The second representation, **type**, is the bag of terms that appear in the names of the categories that constitute the entity type. Unless otherwise stated, whenever the *doc* and *type* representations are used, we use the set of terms in the query title and the set of terms in the names of the categories which constitute the query target type, respectively. The third entity representation, **score**, is the retrieval score assigned to the entity. The score can rely on either (or both) properties of the entity (its associated document and its type).

Below we present query-performance prediction approaches, denoted $\mathcal{P}$, which utilize the entity representations. We use $\mathcal{P}_{rep-r}$, where $r \in \{doc, type, score\}$ is the entity representation used by $\mathcal{P}$, to denote the resultant prediction methods.

Some of the predictors we explore utilize inter-entity similarity measures. The first measure, referred to as $sim = doc$, is the language-model-based similarity between the (Wikipedia) documents associated with the entities. The similarity between documents x and y is $\exp(-CE(p_x^{[0]}(\cdot)||p_y^{[1000]}(\cdot)))$; CE is the cross entropy measure; $p_z^{[\mu]}(\cdot)$ is the Dirichlet-smoothed unigram language model induced from z with the smoothing parameter μ. The second inter-entity similarity measure is based on the entity type: $sim = type$. The measure is the cosine similarity between the binary vectors that represent two entities in the category space. An entry in the vector is 1 if the corresponding category is associated with the entity and 0 otherwise.

To integrate a predictor which uses the *doc* entity representation (inter-entity similarity measure) with a predictor which uses the *type* representation (inter-entity similarity measure) we multiply the prediction values and denote the integration as $rep = doc \wedge type$ ($sim = doc \wedge type$).

3.1 Prediction approaches

3.1.1 Pre-retrieval predictors

Pre-retrieval prediction methods analyze the query using corpus-based term statistics prior to retrieval. We adapt two highly effective pre-retrieval methods from document retrieval to the entity retrieval setting.

The first type of predictors is based on analyzing the inverse document frequency (IDF) values of the set of terms in the query title; the *doc* entity representation is used. The resultant predictors are named $AIDF_{rep=doc}$ (cf., [3, 7]), where $A \in \{avg, sum, max\}$ is the aggregation type (average, summation, maximization) of the terms' IDF values.

We also use the IDF values of the set of terms that appear in the names of the categories that constitute the query target type. The *type* entity representation is used yielding the $AIDF_{rep-type}$ predictor.

The predictors just described quantify the discriminative power of the query by analyzing the IDF values of either its title or target type terms. Along the same lines, we study the $AVarTF.IDF_{rep-doc}$ predictor (cf. [16]) which measures for each query title term the variance of its *tf-idf* values across all the entity documents that contain it[1].

[1]Experiments showed that using the $VarTF.IDF$ predictors with the *type* entity representation yields poor prediction quality. Actual numbers are omitted as they convey no additional insight.

3.1.2 Post-retrieval predictors

We now describe post-retrieval predictors that analyze the n most highly ranked entities in a result list retrieved by an entity retrieval method; n is a free parameter.

Clarity. The *Clarity* prediction method [3], proposed for document retrieval, is based on the premise that the more focused the result list with respect to the corpus the more effective the retrieval. Specifically, the KL divergence between a relevance language model [8] induced from the result list and a language model induced from the corpus is used to measure focus. For the entity retrieval task, we use the *doc* entity representation for *Clarity* computation. The resultant $Clarity_{rep-doc}$ predictor is the analogue of the *Clarity* predictor used for document retrieval [3]. Alternatively, the focus of the entity result list can be measured using the *type* entity representation, yielding the $Clarity_{rep=type}$ predictor. Particularly, a relevance language model induced from the bags of terms that represent the entity types is used.

QF. Query feedback (QF) [17] is based on measuring the robustness of the result list. Specifically, a relevance model is constructed from the original result list and is used to retrieve a second list from the corpus. The overlap between the two lists, measured by the number of documents which are at the l_{qf} highest ranks of both lists, is used for prediction; l_{qf} is a free parameter. Higher prediction value presumably attests to improved robustness of the result list, and therefore to increased retrieval effectiveness. For the entity retrieval task, we simply use the *doc* entity representation for QF computation. The resultant $QF_{rep=doc}$ predictor is the analogue of that used for document retrieval.

WIG and NQC. The WIG [17] and NQC [13] methods measure the mean and standard deviation, respectively, of document retrieval scores in the result list. To apply WIG and NQC for the entity retrieval task, we use the *score* entity representation; i.e, the retrieval scores of entities in the result list are utilized. The resultant predictors are $NQC_{rep=score}$ and $WIG_{rep=score}$, respectively.[2]

Cohesion. It was suggested that a cohesive document result list indicates effective retrieval [2]. We measure the cohesion of the entity result list by the average similarity between two entities in the list using the *doc* and *type* inter-entity similarity measures. The resultant predictors are denoted $Cohesion_{sim-doc}$ and $Cohesion_{sim=type}$, respectively.

AutoCorrelation (AC). The auto-correlation predictor [6] (AC), which was proposed for document retrieval, measures the extent to which similar documents in the result list are assigned with similar retrieval scores. We use AC for the entity retrieval task as follows. First, the retrieval scores of the entities in the result list are normalized to have a zero mean and unit variance. Then, all entities in the list are assigned with a second score. This new ("regularized") score is the weighted average of the original (normalized)

[2]We do not use the corpus-based retrieval score normalization as in the original implementations of WIG [17] and NQC [13]. Rather, we sum-normalize the entity retrieval score with respect to the scores of all entities in the result list following previous recommendations [13].

scores of the entity's k nearest neighbors in the list; k is a free parameter. Nearest neighbors are determined using the inter-entity similarity measures (*doc* or *type*) which also serve for weighting. The prediction value is the Pearson correlation between the original (normalized) scores in the list and the new scores. The resultant predictors, which differ by the inter-entity similarity measure employed, are denoted $AC_{rep=score;sim=doc}$ and $AC_{rep=score;sim=type}$. These predictors are based on the premise that "similar" entities should be assigned with similar retrieval scores. This prediction principle is a manifestation of the cluster hypothesis which was recently explored for entity retrieval [12].

Max Cluster Score (MCS). The AC predictor can assign high prediction values to result lists with very low (yet similar) retrieval scores. The *WIG* predictor assigns a high prediction value if the entities' scores at the top ranks of the list are high. However, *WIG* does not account for the extent to which similar entities are assigned with similar scores. Hence, to conceptually leverage the strengths of the two approaches, we present a novel prediction method (MCS).

The predictor uses nearest-neighbor clustering of the entity result list. Each entity and its k nearest neighbors in the list form a cluster. The score of a cluster is the geometric mean of the normalized retrieval scores of its constituent entities [12].[3] The maximal cluster score is the prediction value. The resultant predictors, $MCS_{rep=score;sim=doc}$ and $MCS_{rep=score;sim=type}$, use the *doc* and *type* inter-entity similarity measures, respectively, to create clusters. The prediction principle is that a result list which contains entities that are (i) similar to each other, and (ii) assigned with high retrieval scores, is likely to be effective.

4. EVALUATION

4.1 Experimental setup

We performed experiments with the datasets of the INEX *entity ranking track* of 2007 and 2008 [4, 5]. These tracks used the English Wikipedia dataset from 2006. The tracks provide a total of 109 topics for the entity ranking task, which were originally used for training and testing. We use all of these queries in our experiments.[4] The data is preprocessed using Lucene[5], including tokenization, stopword removal, and Porter stemming.

To measure prediction quality, we follow common practice in work on QPP for document retrieval [2]. We use the Pearson correlation between the prediction values assigned to a set of queries by a predictor and the ground-truth average precision (AP@1000) which is determined based on relevance judgements.[6]

To set the values of free parameters of predictors, we applied 100 tests of 2-fold cross validation performed over all queries. The resultant average prediction quality is reported. Statistically significant differences of prediction quality are determined using the two-tailed paired t-test computed over the folds using a 95% confidence level. Prediction quality (measured using Pearson correlation) serves as the optimization criterion in the learning phase. The 2-fold procedure enables to have enough queries (~55) in both the train and test sets so as to compute Pearson correlation in a robust manner. The free-parameter values of each predictor's version (*doc*, *type* and $doc \wedge type$) were learned separately.

Clarity and *QF* use the RM1 relevance model [8] which is constructed from maximum likelihood estimates of the entities' representations (*doc* or *type*). The exponent of the entities' retrieval scores (described below) serve for entity weighting. The number of terms used by RM1, and the number of top-retrieved entities used to construct it, are set to values in $\{10, 50, 100\}$ and $\{25, 50, 100\}$, respectively. QF's l_{qf} parameter is selected from $\{5, 10, 20, 30, 40, 50\}$.

The number of most highly ranked entities considered by *WIG* and *NQC*, n, is selected from $\{5, 10, 20, 30, 40, 50, 100\}$ and $\{10, 20, 30, 40, 50, 100, 500\}$, respectively. For *Cohesion*, AC and MCS, n is set to values in $\{10, 50, 100\}$. The number of nearest neighbors, k, used in the AC and MCS predictors, is selected from $\{4, 9\}$.

We predict the effectiveness of two lists, each contains 1000 entities, that are retrieved using effective methods [11]. The first, L_D, is created by applying a standard language-model-based approach upon the *doc* representation of entities. The score of entity e, represented by document e_x, with respect to query q is based on the cross entropy measure: $S_D(e) \stackrel{def}{=} -CE(p_q^{[0]}(\cdot)||p_{e_x}^{[100]}(\cdot))$. (Refer to the description of the inter-entity similarity measures in Section 3 for details regarding the language model notation used.) The second list, $L_{D;T}$, is created by re-ranking L_D using a linear interpolation of two entity retrieval scores. The first is that used to create L_D (i.e., $S_D(e)$). The second is an entity-type-based score, $S_T(e)$. Specifically, it is the minus of the minimum (normalized) distance, over Wikipedia's category graph, between a category in the query target type and a category among those associated with e. The interpolated score assigned to e is: $\lambda \log \frac{\exp(S_D(e))}{\sum_{e' \in L_D} \exp(S_D(e'))} + (1 - \lambda) \log \frac{\exp(S_T(e))}{\sum_{e' \in L_D} \exp(S_T(e'))}$; λ is a free parameter set to 0.5. The rest of the technical details regarding the implementation of the retrieval methods follow those in [11].

4.2 Experimental results

Table 1 presents the prediction quality numbers. Our first observation is that the most effective pre-retrieval predictors are outperformed by the most effective post-retrieval predictors, as reported for document retrieval [2]. Also, the *Clarity* predictors are less effective than most other post-retrieval predictors. *QF*, which is a state-of-the-art predictor for document retrieval, is outperformed (often substantially) by quite a few other post-retrieval predictors. *WIG* and *NQC*, which analyze retrieval scores, are highly effective, similarly to the case for document retrieval [2].

The *Cohesion* predictor posts poor prediction quality when using the *doc* inter-entity similarity measure. This finding is in accordance with those reported for document retrieval [2]. However, the prediction quality is relatively high when using the *type* inter-entity similarity measure. Thus, an entity result list which is cohesive in terms of the categories of

[3]Normalized retrieval scores are attained by a sum-normalization of the exponents of the original scores.

[4]We did not use the 2009 dataset since there are too few queries for learning free-parameter values of predictors.

[5]http://lucene.apache.org/core/

[6]The performance for queries of the 2008 track was originally evaluated using extended inferred average precision (xinfAP) [15]. We found that the standard AP measure is 99.99% correlated with xinfAP for the retrieval methods we use. Hence, for consistency with the queries used in 2007, AP was used in all experiments.

the entities it contains is somewhat likely to be effective. In contrast to the case for *Cohesion*, for the *AC* predictor the *doc* inter-entity similarity measure is more effective than the *type* measure. This finding could potentially be attributed to the sparseness of the *type* measure. That is, in some cases an entity might not share categories with other entities in the list and hence the inter-entity similarity is 0. We use entity IDs to break similarity ties.

Predictors employed with both the $rep = doc$ and $rep = type$ representations are in most cases more effective when using the former than the latter. Yet, in quite a few cases (e.g., for *maxIDF* and *Clarity*), using both representations ($rep = doc \wedge type$) is superior to using either.

The prediction quality for almost all predictors is higher for the L_D list than it is for the $L_{D;T}$ list. Recall that $L_{D;T}$ is a re-ranked version of L_D created by interpolation of two entity scores. The first is based on the entity's document and the second is based on the entity's categories. However, the category-based information (distance in the Wikipedia category graph) is different than that used by the prediction methods (terms in categories' names), and therefore the prediction quality for $L_{D;T}$ might be lower. We hasten to point out, however, that some of the prediction quality numbers for $L_{D;T}$ are quite high and competitive with those for L_D; e.g., for $WIG_{rep=score}$ and $NQC_{rep=score}$ that use retrieval scores.

Our novel *MCS* predictor is the most effective for the L_D list when using the *doc* inter-entity similarity measure ($MCS_{rep=score;sim=doc}$); this predictor outperforms to a statistically significant degree all other predictors. Furthermore, $MCS_{rep=score;sim=type}$ outperforms to a statistically significant degree all predictors except for *WIG*. For the $L_{D;T}$ list, $MCS_{rep=score;sim=type}$ and $MCS_{rep=score;sim=doc}$ are the second and fourth best, respectively. While the former outperforms all predictors, except for *WIG*, to a statistically significant degree, it is outperformed by *WIG* in a statistically significant manner. All in all, these findings attest to the merits of our *MCS* predictor that relies on the cluster hypothesis.

Predictor	L_D	$L_{D;T}$
$avgIDF_{rep=doc}$	$0.555^{d,t}_{\wedge}$	$0.441^{d,t}_{\wedge}$
$avgIDF_{rep=type}$	$0.297^{d,t}_{\wedge}$	$0.248^{d,t}_{\wedge}$
$avgIDF_{rep=doc\wedge type}$	$0.523^{d,t}_{\wedge}$	$0.414^{d,t}_{\wedge}$
$sumIDF_{rep=doc}$	$0.070^{d,t}_{\wedge}$	$-0.002^{d,t}_{\wedge}$
$sumIDF_{rep=type}$	$0.042^{d,t}_{\wedge}$	$0.106^{d,t}_{\wedge}$
$sumIDF_{rep=doc\wedge type}$	$0.100^{d,t}_{\wedge}$	$0.105^{d,t}_{\wedge}$
$maxIDF_{rep=doc}$	$0.475^{d,t}_{\wedge}$	$0.280^{d,t}_{\wedge}$
$maxIDF_{rep-type}$	$0.254^{d,t}_{\wedge}$	$0.191^{d,t}_{\wedge}$
$maxIDF_{rep=doc\wedge type}$	$0.489^{d,t}_{\wedge}$	$0.301^{d,t}_{\wedge}$
$avgVarTF.IDF_{rep=doc}$	$0.547^{d,t}_{\wedge}$	$0.444^{d,t}_{\wedge}$
$sumVarTf.IDF_{rep=doc}$	$0.395^{d,t}_{\wedge}$	$0.294^{d,t}_{\wedge}$
$maxVarTf.IDF_{rep=doc}$	$0.532^{d,t}_{\wedge}$	$0.377^{d,t}_{\wedge}$
$Clarity_{rep=doc}$	$0.370^{d,t}_{\wedge}$	$0.295^{d,t}_{\wedge}$
$Clarity_{rep=type}$	$0.303^{d,t}_{\wedge}$	$0.279^{d,t}_{\wedge}$
$Clarity_{rep=doc\wedge type}$	$0.369^{d,t}_{\wedge}$	$0.312^{d,t}_{\wedge}$
$WIG_{rep=score}$	$0.651^{d}_{\wedge}$	$\mathbf{0.623}^{d,t}_{\wedge}$
$NQC_{rep=score}$	$0.600^{d,t}_{\wedge}$	$0.578^{d,t}_{\wedge}$
$QF_{rep=doc}$	$0.437^{d,t}_{\wedge}$	$0.410^{d,t}_{\wedge}$
$Cohesion_{sim=doc}$	$-0.026^{d,t}_{\wedge}$	$-0.106^{d,t}_{\wedge}$
$Cohesion_{sim-type}$	$0.508^{d,t}_{\wedge}$	$0.403^{d,t}_{\wedge}$
$Cohesion_{sim=doc\wedge type}$	$0.360^{d,t}_{\wedge}$	$0.257^{d,t}_{\wedge}$
$AC_{rep=score;sim=doc}$	$0.475^{d,t}_{\wedge}$	$0.378^{d,t}_{\wedge}$
$AC_{rep=score;sim=type}$	$0.418^{d,t}_{\wedge}$	$0.319^{d,t}_{\wedge}$
$AC_{rep=score;sim=doc\wedge type}$	$0.468^{d,t}_{\wedge}$	$0.360^{d,t}_{\wedge}$
$MCS_{rep=score;sim=doc}$	**0.665**	0.563
$MCS_{rep=score;sim=type}$	0.650	0.596
$MCS_{rep=score;sim=doc\wedge type}$	0.591	0.502

Table 1: Prediction quality. The best result in a column per a result list is boldfaced. 'd', 't' and '$\wedge$' mark statistically significant differences with $MCS_{rep-score;sim-doc}$, $MCS_{rep-score;sim-type}$ and $MCS_{rep=score;sim=doc\wedge type}$, respectively.

5. CONCLUSIONS

We presented a study of adapting query-performance predictors proposed for document retrieval to the entity retrieval domain. We also presented a novel predictor that relies on the cluster hypothesis and showed that it can often outperform the most effective predictors we studied.

6. ACKNOWLEDGEMENTS

We thank the reviewers for their comments. This paper is based on work supported in part by the Israel Science Foundation under grant no. 433/12 and by a Google faculty research award.

7. REFERENCES

[1] K. Balog, M. Bron, and M. De Rijke. Query modeling for entity search based on terms, categories, and examples. *ACM Trans. Inf. Syst.*, 29(4), 2011.

[2] D. Carmel and E. Yom-Tov. Estimating the query difficulty for information retrieval. *Synthesis Lectures on Information Concepts, Retrieval, and Services*, pages 1–89, 2010.

[3] S. Cronen-Townsend, Y. Zhou, and W. B. Croft. Predicting query performance. In *Proc. of SIGIR*, pages 299–306, 2002.

[4] A. P. de Vries, A.-M. Vercoustre, J. A. Thom, N. Craswell, and M. Lalmas. Overview of the INEX 2007 entity ranking track. In *Proc. of INEX*, pages 245–251, 2007.

[5] G. Demartini, A. P. de Vries, T. Iofciu, and J. Zhu. Overview of the INEX 2008 entity ranking track. In *Proc. of INEX*, pages 243–252, 2008.

[6] F. Diaz. Performance prediction using spatial autocorrelation. In *Proc. of SIGIR*, pages 583–590, 2007.

[7] C. Hauff, D. Hiemstra, and F. de Jong. A survey of pre-retrieval query performance predictors. In *Proc. of CIKM*, pages 1419–1420, 2008.

[8] V. Lavrenko and W. B. Croft. Relevance based language models. In *Proc. of SIGIR*, pages 120–127, 2001.

[9] A. marie Vercoustre, J. Pehcevski, and V. Naumovski. Topic difficulty prediction in entity ranking. In *Proc. of INEX*, pages 280–291, 2009.

[10] J. Pound, P. Mika, and H. Zaragoza. Ad-hoc object retrieval in the web of data. In *Proc. of WWW*, pages 771–780, 2010.

[11] H. Raviv, D. Carmel, and O. Kurland. A ranking framework for entity oriented search using markov random fields. In *Proc. of JIWES*, 2012.

[12] H. Raviv, O. Kurland, and D. Carmel. The cluster hypothesis for entity oriented search. In *Proc. of SIGIR*, pages 841–844, 2013.

[13] A. Shtok, O. Kurland, D. Carmel, F. Raiber, and G. Markovits. Predicting query performance by query-drift estimation. *ACM Transactions on Information Systems*, 30(2):11, 2012.

[14] J. A. Thom, J. Pehcevski, and A.-M. Vercoustre. Use of wikipedia categories in entity ranking. *CoRR*, abs/0711.2917, 2007.

[15] E. Yilmaz, E. Kanoulas, and J. A. Aslam. A simple and efficient sampling method for estimating ap and ndcg. In *Proc. of SIGIR*, pages 603–610, 2008.

[16] Y. Zhao, F. Scholer, and Y. Tsegay. Effective pre-retrieval query performance prediction using similarity and variability evidence. In *Proc. of ECIR*, pages 52–64, 2008.

[17] Y. Zhou and W. B. Croft. Query performance prediction in web search environments. In *Proc. of SIGIR*, pages 543–550, 2007.