Predicting the Performance of Passage Retrieval for Question Answering

Eyal Krikon
krikon@gmail.com

David Carmel
carmel@il.ibm.com
IBM Research Lab
Haifa 31905, Israel

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

ABSTRACT

We present a novel approach to predicting the performance of passage retrieval for question answering. That is, estimating the effectiveness, for answer extraction, of a list of passages retrieved in response to a question when relevance judgments are not available. Our prediction model integrates two types of estimates. The first estimates the probability that the information need expressed by the question is satisfied by the passages. This estimate is devised by adapting query-performance predictors developed for the document retrieval task. The second type estimates the probability that the passages contain the answers. This estimate relies on the occurrences of named entities that are likely to answer the question. Empirical evaluation demonstrates the merits of our prediction approach. For example, the prediction quality is much better than that of the only previous prediction method devised for the task at hand.

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

General Terms: Algorithms, Experimentation

Keywords: Question answering, Passage Retrieval, Query Performance Prediction

1. INTRODUCTION

The retrieval process of typical QA systems is based on two main phases [10, 12, 9, 8]. The first is retrieving passages from the collection that presumably contain the right answers. The second phase is extracting the answers from these passages. The performance of the second phase, and that of the entire system, is known to significantly depend on the performance of the first [5, 12, 8].

Thus, it is important for a QA system to be able to automatically determine if the first phase failed and the retrieved passages cannot be effectively used for the subsequent answer extraction phase. Accordingly, the challenge we focus on is *predicting* the effectiveness of the first phase.

CIKM'12, October 29–November 2, 2012, Maui, HI, USA.

We present a novel performance prediction method for passage retrieval for QA that predicts the "quality" of a retrieved passage list; i.e., the presumed inclusion of many relevant passages that contain the correct answers. Our model integrates two types of estimates. The first is for the probability that the information need expressed by the question is satisfied by the passage list; i.e., that the list provides question-related context. For this estimate, we adapt state-of-the-art query-performance predictors that were devised for the document retrieval task. The second type of estimate is for the probability that the passage list contains pertaining answers. Since the answers to many questions are named entities, we base the estimate on the presence of named entities, which are of the same type as that of the question's answer, within the question's context (i.e., in proximity to the question terms) in the passages.

Evaluation performed with TREC benchmarks shows that our approach substantially outperforms the only prediction method previously employed for the task we address here [7]. Furthermore, we show that applying our entity-based estimate benefits retrieval and prediction both independently and when applied simultaneously for these tasks.

2. RELATED WORK

The Clarity measure [6], which was originally designed for predicting document-retrieval effectiveness, was applied on passages to predict their effectiveness for the answer extraction phase [7] — the task we address here. Although there is abundance of work on predicting query performance for document retrieval [4], to the best of our knowledge, there was no further work on devising prediction methods for passage retrieval for QA. We show that our approach substantially outperforms Clarity [7].

Our *prediction* method uses occurrence statistics of named entities in the retrieved passages. There is much work on using named entities for improving the passage *retrieval* performance for QA [10, 3, 8, 1]. As noted above, our entity-based estimate is effective for both prediction and retrieval.

3. PERFORMANCE PREDICTION

The challenge we address is performance prediction for the passage retrieval phase of the question answering (QA) task. Specifically, the goal is to predict the effectiveness (for answer extraction purposes) of a ranked list of passages, G, that was retrieved in response to question q by *some* retrieval method; $S(g; q)$ denotes the retrieval score assigned

to passage g (in G). In Section 4.1 we discuss the passage retrieval method used for evaluation.

3.1 Prediction framework

The prediction task can be formally stated as estimating the probability $p(A_q|G)$ that the answer A_q to the question q can be found in (or extracted from) G.

A key observation that we make is that a requirement for the passage list G to be effective for answer extraction is that the answers appear within a context that is relevant to the information need, I_q, expressed by q. For example, for the question $q \stackrel{def}{=}$ "What is the most densely populated city in France?", the correct answer, A_q, is "Paris"; the information need, I_q, is "densely populated cities in France". Now, if the retrieved passages contain the correct answer, "Paris", but do not discuss aspects related to population density but rather topics such as geography, history, or even celebrities (e.g., Paris Hilton), then the answer extraction phase is likely to fail. Indeed, the guidelines for TREC's QA tracks [14] specify that a correct answer appearing in a document that does not allow for identification or verification of the answer's correctness is not considered correct.

Accordingly, the prediction task is devising an estimate

$$\hat{p}(A_q|G) \stackrel{def}{=} \hat{p}(A_q|I_q, G)\hat{p}(I_q|G), \quad (1)$$

where $\hat{p}(I_q|G)$ is an estimate for the probability that I_q is satisfied by G; and, $\hat{p}(A_q|I_q, G)$ is an estimate for the probability that the correct answer A_q can be extracted from G (i.e., is found within G in the right context) given that I_q is satisfied. In what follows we present estimates that can be used to instantiate specific predictors from Eq. 1.

3.1.1 Information need satisfaction

We first devise an estimate $\hat{p}(I_q|G)$ for the probability that the information need is satisfied by passages in G. To that end, we adapt previously proposed predictors, denoted $P(I_q|G)$. These predictors were devised and shown to be effective for estimating document retrieval effectiveness with respect to information need expressed by a query. Similarly to the document retrieval case [4], here we employ the predictors on $G^{[k]}$, the k highest ranked passages in G. The focus on highly ranked passages serves to better align the prediction with the passage-retrieval performance metric of concern. Specifically, this is the average precision (AP) measure that will be used in Section 4 as the goal of prediction, and which was used in previous work for evaluating passage retrieval effectiveness for QA [9]. We now turn to describe the post-retrieval predictors adapted for passage retrieval.

Clarity. The Clarity predictor [6] estimates the focus of G with respect to a corpus of documents $\mathcal{D}$, by measuring the KL divergence between their induced language models; passages in G are parts of documents in $\mathcal{D}$. (See Section 4.1 for details regarding the passage retrieval methods employed.) The higher the divergence, the more focused G is considered to be, and consequently, more likely to satisfy the underlying information need. Specifically,

$$P_{Clarity}(I_q|G) \stackrel{def}{=} \sum_w p(w|G^{[k]}) log \frac{p(w|G^{[k]})}{p(w|\mathcal{D})}; \quad (2)$$

$p(w|x)$ is the probability assigned to term w by a language model induced from x. (We describe the language-model induction techniques in Section 4.1.)

WIG. The WIG predictor [15] is based on the premise that high retrieval scores in the list imply to its effectiveness in satisfying the information need. Indeed, WIG was shown to be a highly effective query-performance predictor for document retrieval, and is formally defined here by:

$$P_{WIG}(I_q|G) \stackrel{def}{=} \frac{1}{k} \sum_{g \in G^{[k]}} S(g; q). \quad (3)$$

NQC. While WIG is based on the average retrieval score of top-retrieved passages, the NQC predictor measures the standard deviation [11]. High deviation of *document* retrieval scores was argued to imply reduced "query drift", and thereby improved effectiveness for document retrieval. Accordingly, we define

$$P_{NQC}(I_q|G) \stackrel{def}{=} \sqrt{\frac{1}{k} \sum_{g \in G^{[k]}} (S(g; q) - \mu)^2}, \quad (4)$$

where $\mu = \frac{1}{k} \sum_{g \in G^{[k]}} S(g; q)$ is the average score in $G^{[k]}$. Originally, both WIG [15] and NQC [11] were normalized using the corpus retrieval score. We do not employ this normalization as the passage retrieval scores are already normalized (see Section 4.1) and further corpus-based normalization yielded degraded prediction quality.

3.1.2 Answer extraction

We next devise an estimate $\hat{p}(A_q|I_q, G)$ (see Eq. 1) for the probability that a correct answer can be found in (extracted from) G given that the information need is satisfied. We refer to this estimate as a "judge" [8].

For many questions the answer is a named entity [3]. This fact can be exploited for devising a judge. In what follows, we focus on four types of entities that were found to be effective for the question answering task [3]; namely, Person, Organization, Location and Date. We use *Misc* to refer to the type of an answer that is not an entity or that is an entity of a type not among the four just specified (e.g., "What is the distance of the moon from earth?"). We assume that for each question the answer type t ($\in \{Person, Organization, Location, Date, Misc\}$) can be identified [8]; q_t denotes a question q with answer type t.

3.1.3 Entity-based judges

We first consider a judge for questions with answers that are entities with types in {Person, Organization, Location, Date}. Then, we present a generalized judge that also accounts for questions with the Misc answer type.

The **NEQ** judge that we consider accounts only for entities that occur in a question-related context. Specifically, an entity E_t of type t is considered highly related to q_t if E_t and *all* q_t's terms appear together in a short window of text within the passage. The statement $(E_t, q_t) \in win^g_\Delta$ is used below to specify that such co-occurrence holds in a window, win^g_Δ, that is composed of Δ consecutive terms in passage g; Δ is a free parameter.

$$P_{NEQ}(A_{q_t}|I_{q_t}, G) \stackrel{def}{=} \quad (5)$$
$$\log\Big(\sum_{g \in G^{[k]}} \#\{E_t : E_t \in g \bigcap (E_t, q_t) \in win^g_\Delta\} + 2\Big).$$

We apply log transformation to moderate the effect of large counts and employ add-2 smoothing to avoid zero multiplication in Eq. 1. Thus, NEQ considers G as likely to contain

the answer if G contains many entities of the same type as that of the question's answer type and that co-occur together with the question terms.

If the answer to q is of type Misc, then NEQ cannot be applied. More generally, it could be that NEQ should be more "trusted" for certain types of answers than for others. For example, if entities of a certain type are more likely, in general, to have high occurrence in texts than entities of another type, then the resultant NEQ estimate can be unjustifiably biased across answer types. Since one of our goals is to compare prediction quality across questions of varying answer types, we study the **NEQT** judge:

$$P_{NEQT}(A_{q_t}|I_{q_t}, G) \stackrel{def}{=} \lambda_t P_{NEQ}(A_{q_t}|I_{q_t}, G) + (1 - \lambda_t). \quad (6)$$

NEQT controls, on a per answer-type basis, the reliance on the NEQ judge; λ_t is a free parameter that depends on t. That is, for a small λ_t value, NEQT backs off from using NEQ to a constant-based prediction value, which amounts to more heavily relying on the information-need satisfaction aspect of prediction. (See Eq. 1.). As NEQ cannot be applied to Misc questions, we set λ_{Misc} to 0.

4. EVALUATION

4.1 Experimental setup

As is common in work on QA [3, 12, 9], the first step of our passage retrieval approach is retrieving from the corpus a document list, $\mathcal{D}_q$, based on document-question similarities. The *sentences* in these documents, capped at 64 terms, serve for passages; $\mathcal{D}_q$ contains 500 documents as this results in highly effective passage retrieval performance.

To measure the similarity between the question q and text x, a passage or a document, we use a language-model-based estimate: $Sim(q, x) \stackrel{def}{=} \prod_{q_i \in q} p(q_i|x)$, where $p(w|x)$ is the probability assigned to term w by a Dirichlet-smoothed unigram language model induced from x with the smoothing parameter set to 2000.

Passage g is scored with respect to q by a common passage retrieval approach [2], henceforth referred to as **PDQ**:

$$S_{PDQ}(g; q) \stackrel{def}{=} \frac{\alpha Sim(q, g)}{\sum_{g \in d \in \mathcal{D}_q} Sim(q, g)} + \frac{(1 - \alpha) Sim(q, d_g)}{\sum_{d \in \mathcal{D}_q} Sim(q, d)}; \quad (7)$$

d_g is the document to which g belongs; α is a free parameter set to 0.9 as this yields (near)-optimal passage retrieval performance (with respect to values in $\{0, 0.1, \ldots, 1\}$) for all experimental settings. The 1000 most highly ranked passages serve for the passage list; estimating the effectiveness of this list is the prediction goal.

As discussed in Section 3, we use the average precision measure (AP at cutoff 1000) to evaluate the effectiveness of the passage list for QA. We measure the AP of a list of passages using TREC's relevance judgments and the answer patterns associated with the questions [13]. A passage is considered relevant if it (i) contains at least one of the answer patterns, and (ii) is extracted from a relevant document [13].

Pearson's correlation between predicted (AP) performance and actual (AP) performance is the prediction quality measure as in work on query-performance prediction for document retrieval [4]. Statistical significance of prediction values is determined using Pearson's correlation significance test at a confidence level of 95%.

The prediction methods we study incorporate free parameters. The values of these parameters, listed below, were selected based on the prediction quality attained for the various experimental settings after extensive experimentation with wide ranges. For Clarity (Eq. 2), k, the number of top-ranked passages used for inducing the (relevance) language model, which uses 100 terms, was set to 100. For WIG (Eq. 3) and NQC (Eq. 4), k was set to 5 and 25, respectively. For the NEQ judge (Eq. 5), k was set to 5. For λ_t (Eq. 6), which controls the reliance on NEQ for the varying question types, 0 was used for the Misc type, and a fixed positive value was effective for the other question types; specifically, the values 0.2, 0.7, and 0.8 were used when NEQ was integrated with Clarity, WIG, and NQC respectively. The window size, Δ, in Eq. 5 was set to 50.

We used Stanford's NER[1], a named entity extraction application, for annotating Person, Organization and Location entities within the passages. For Date entities, we used a simple date pattern detection method. Although some classification schemes were successfully employed for categorizing questions [3, 7, 1], we manually categorized the test questions to achieve maximal classification accuracy; the classification problem is out of the scope of this paper.

Experiments were conducted using TREC QA datasets:

Corpus	# docs	Questions	TREC collection
TREC99	528,155	1-200	Disks 4-5 - CR
TREC00	978,952	201-893	Disks 1-5
TREC01	978,952	894-1393	Disks 1-5
TREC02	1,033,461	1394-1893	ACQUAINT

We applied tokenization and Porter stemming to all data using the Lemur/Indri toolkit. We did not remove stopwords from the documents nor from the questions. However, for the NEQ judge, which counts only entities occurring together with *all* question terms in a small text window, we removed stopwords from the questions (using the INQUERY stopword list plus single letter terms) as it significantly improved its performance.

4.2 Experimental results

Main result. Table 1 presents the prediction quality numbers. The second row presents the prediction quality of NEQT when used alone. The following rows present the prediction quality of the post-retrieval predictors when used alone and when integrated with NEQT based on Eq. 1.

NEQT posts low prediction quality when used alone. A possible explanation is that the passage list may contain many entities having the question's answer type and hence highly estimated by NEQT. However, whereas the list does not pertain to the information need, we deem it ineffective. In contrast, once NEQT is integrated with any of the post-retrieval predictors, all prediction quality numbers are statistically significant. Furthermore, we get an increased prediction quality in nearly all cases with respect to that of not using NEQT. These findings support the merits of our prediction approach that integrates a prediction for information need satisfaction with an estimate, based on named entities, for the probability that an answer can be extracted.

A comparison of the predictors, with or without NEQT, reveals that WIG·NEQT and NQC·NEQT yield the highest

[1]http://www-nlp.stanford.edu/software/CRF-NER.shtml

	TREC99	TREC00	TREC01	TREC02
NEQT	−0.048	0.028	0.065	0.025
Clarity	*0.361*	*0.372*	*0.227*	*0.168*
Clarity·NEQT	*0.349*	*0.373*	*0.248*	*0.180*
WIG	*0.430*	*0.408*	*0.349*	*0.244*
WIG·NEQT	*0.447*	***0.437***	***0.380***	*0.285*
NQC	*0.457*	*0.410*	*0.301*	*0.275*
NQC·NEQT	***0.471***	*0.433*	*0.353*	***0.312***

Table 1: Main result. (Prediction quality measured using Pearson's correlation.) Boldface: the best result in a column. Statistically significant correlations are italicized.

	List	TREC99	TREC00	TREC01	TREC02
NQC	G_{PDQ}	0.457	0.410	0.301	0.275
	$G_{PDQ \wedge NEQ}$	**0.479**	**0.445**	0.341	0.313
NQC· NEQT	G_{PDQ}	0.471	0.433	0.353	0.312
	$G_{PDQ \wedge NEQ}$	0.462	0.427	**0.357**	**0.327**

Table 2: Prediction quality for different passage lists. Boldface: the best result in a column. All correlations are statistically significant.

prediction quality over all collections. In fact, the Clarity predictor, the only baseline for predicting the effectiveness of passage retrieval for QA in previous work [7], is inferior in all cases to all other predictors when used alone (as in [7]), and when integrated with NEQT.

Using entity-based estimates for retrieval. We demonstrated above the effectiveness of our approach, which uses an entity-based judge, for predicting the performance of the PDQ retrieval method (Eq. 7) that does not use entity-based information. We now explore whether our prediction approach is also effective for predicting the performance of a passage retrieval method that uses entity-based information. To that end, we use the **PDQ∧NEQ** method, which scores passage g by: $S_{PDQ}(g;q)P_{NEQ}(A_{q_t}|I_{q_t},g)$ (refer back to Eq. 5 and 7). To alleviate the computation effort, PDQ∧NEQ re-ranks the 50 initially highest ranked passages; lower ranked passages retain their original ranks. Furthermore, passages retrieved for Misc type questions are not re-ranked as they have no target entity, and hence, re-ranking (based on NE occurrences) has no impact.

As it turns out, PDQ∧NEQ outperforms PDQ in a substantial and statistically significant manner (two tailed paired t-test, $p = 0.05$). Specifically, PDQ's MAP performance is .31, .24, .21, and .18 for TREC'99, TREC'00, TREC'01, and TREC'02, respectively, while that of PDQ∧NEQ is .33, .26, .23, and .21. This finding, which echoes those from work on using named entities for improving passage retrieval performance for QA [3, 5, 1], attests to the effectiveness, in terms of retrieval performance, of our entity-based judge.

In Table 2 we present the prediction quality of NQC and NQC·NEQT over all the question types studied (including Misc) when applied over G_{PDQ} (the passage list used insofar) and $G_{PDQ \wedge NEQ}$, the passage list created using PDQ∧NEQ. (Using WIG yields similar prediction quality patterns.)

We can see in Table 2 that the prediction quality for $G_{PDQ \wedge NEQ}$ is often superior to the prediction quality for G_{PDQ}. Thus, our prediction approach is effective whether the passage retrieval method uses entity-based information or not. Furthermore, these findings allow for maximal flexibility in terms of resource allocation between the retrieval and prediction tasks for QA, because the entity-based judge can be used simultaneously for improving retrieval and for predicting retrieval performance.

5. CONCLUSION

We presented a novel approach to predicting the performance of passage retrieval for question answering and empirically demonstrated its merits. The approach integrates predictors adapted from work on performance prediction for document retrieval with named-entity-based estimates.

Acknowledgments We thank the reviewers for their comments. This paper is based upon work supported in part by the Israel Science Foundation under grant no. 557/09, by IBM's SUR award, and by Google's and Yahoo!'s faculty research awards. Any opinions, findings and conclusions or recommendations expressed in this material are the authors' and do not necessarily reflect those of the sponsors.

6. REFERENCES

[1] E. Aktolga, J. Allan, and D. A. Smith. Passage reranking for question answering using syntactic structures and answer types. In *Proceedings of ECIR*, pages 617–628, 2011.

[2] J. P. Callan. Passage-level evidence in document retrieval. In *Proceedings of SIGIR*, pages 302–310, 1994.

[3] C. Cardie, V. Ng, D. Pierce, and C. Buckley. Examining the role of statistical and linguistic knowledge sources in a general-knowledge question-answering system. In *Proceedings of ANLC*, pages 180–187, 2000.

[4] D. Carmel and E. Yom-Tov. *Estimating the Query Difficulty for Information Retrieval*. Synthesis Lectures on Information Concepts, Retrieval, and Services. Morgan & Claypool Publishers, 2010.

[5] A. Corrada-Emmanuel, W. B. Croft, and V. Murdock. Answer passage retrieval for question answering. Technical Report IR-283, Center for Intelligent Information Retrieval, University of Massachusetts, 2003.

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

[7] S. Cronen-Townsend, Y. Zhou, and W. B. Croft. Precision prediction based on ranked list coherence. *Information Retrieval*, 9:723–755, 2006.

[8] D. Ferrucci, E. Brown, J. Chu-Carroll, J. Fan, D. Gondek, A. A. Kalyanpur, A. Lally, J. W. Murdock, E. Nyberg, J. Prager, N. Schlaefer, and C. Welty. Building Watson: An Overview of the DeepQA Project. *AI magazine*, pages 59–79, 2010.

[9] V. Murdock and W. B. Croft. Simple translation models for sentence retrieval in factoid question answering. In *Proceeding of the SIGIR Workshop on Information Retrieval for Question Answering*, pages 31–35, 2004.

[10] J. Prager, E. Brown, A. Coden, and D. Radev. Question-answering by predictive annotation. In *Proceedings of SIGIR*, pages 184–191, 2000.

[11] A. Shtok, O. Kurland, and D. Carmel. Predicting query performance by query-drift estimation. In *Proceedings of ICTIR*, pages 305–312, 2009.

[12] S. Tellex, B. Katz, J. Lin, A. Fernandes, and G. Marton. Quantitative evaluation of passage retrieval algorithms for question answering. In *Proceedings of SIGIR*, pages 41–47. ACM Press, 2003.

[13] E. M. Voorhees. Overview of the trec-9 question answering track. In *Proceedings of TREC-9*, pages 71–80, 2001.

[14] E. M. Voorhees and D. M. Tice. The TREC-8 question answering track evaluation. In *Proceedings of TREC-8*, pages 83–105, 1999.

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