Cluster-Based Focused Retrieval

Eilon Sheetrit
seilon@campus.technion.ac.il
Technion — Israel Institute of Technology

Oren Kurland
kurland@ie.technion.ac.il
Technion — Israel Institute of Technology

ABSTRACT

The focused retrieval task is to rank documents' passages by their presumed relevance to a query. Inspired by work on cluster-based document retrieval, we present a novel cluster-based focused retrieval method. The method is based on ranking clusters of similar passages using a learning-to-rank approach and transforming the cluster ranking to passage ranking. Empirical evaluation demonstrates the clear merits of the method.

ACM Reference Format:
Eilon Sheetrit and Oren Kurland. 2019. Cluster-Based Focused Retrieval. In *The 28th ACM International Conference on Information and Knowledge Management (CIKM '19), November 3–7, 2019, Beijing, China.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3357384.3358087

1 INTRODUCTION

The potential merits of retrieving for a query short passages rather than long and/or heterogeneous documents have long been acknowledged [1, 7, 20]; e.g., users do not have to browse long retrieved documents to find the sought information.

Inspired by work on cluster-based document retrieval (e.g., [8, 11–14, 16, 18, 26]), and the fact that relevant passages tend to be more similar to each other than to non-relevant passages [22], we present the first, to the best of our knowledge, passage retrieval approach that utilizes information induced from clusters of similar passages. The approach is based on learning-to-rank passage clusters; cluster ranking is then transformed to passage ranking.

Empirical evaluation shows that (i) there are extremely effective passage clusters, thereby providing motivation for engaging in cluster-based passage retrieval; and (ii) our approach substantially outperforms highly effective passage retrieval methods.

2 RELATED WORK

There is much work on cluster-based document retrieval; e.g., [8, 11–14, 16, 18, 26]. Our approach was inspired by the state-of-the-art ClustMRF method which ranks document clusters using learning-to-rank and transforms the cluster ranking to document ranking [18]. We adapt some of ClustMRF's features to rank passage clusters in addition to presenting novel features.

Feature-based passage ranking approaches were proposed, but they do not utilize inter-passage similarities [3, 28]. Our approach is shown to substantially outperform an effective representative [28].

There is a growing body of work on neuro passage retrieval [5, 6, 21, 27]. Neuro methods were shown to substantially underperform a feature based method, used here as a baseline, for retrieving answer sentences to non-factoid queries — a passage retrieval task [28]. Incorporating features in a neural approach did show merit for this task [5]. Integrating our approach with neural architectures is a future direction.

3 RETRIEVAL FRAMEWORK

We start by describing notational conventions. Let q and d denote a query and a document, respectively. We use g to denote a passage and d_g to denote g's ambient document. The maximum likelihood estimate of term w with respect to text (or text collection) x is denoted $p_x^{MLE}(w)$. The probability assigned to w by a Dirichlet smoothed unigram language model induced from x is denoted $p_x^{Dir}(w)$. We measure the similarity between two texts x and y using cross entropy (CE) [29]: $Sim(x, y) \stackrel{def}{=} \exp(-CE(p_x^{MLE}(\cdot) \,||\, p_y^{Dir}(\cdot))$.

As is common in work on passage retrieval, we first rank the documents in the corpus in response to the given query q: $\mathcal{L}_{doc}$ is the list of 1000 documents most highly ranked by $Sim(q, d)$ [29]. We then rank the passages of documents in $\mathcal{L}_{doc}$ using *some* retrieval method; $\mathcal{L}_{psg}$ is the list of top-ranked passages, henceforth referred to as "initial list". In Section 4.1 we use three different retrieval methods to produce $\mathcal{L}_{psg}$.

3.1 Cluster-based passage ranking

Our goal is to re-rank the initial passage list, $\mathcal{L}_{psg}$, using information induced from clusters of similar passages, so as to improve retrieval effectiveness. To this end, we first cluster the passages in $\mathcal{L}_{psg}$ yielding the cluster set $C(\mathcal{L}_{psg})$. Then, we rank the clusters in $C(\mathcal{L}_{psg})$ by their presumed relevance to the query. Operationally, here we quantify the degree of cluster relevance in terms of the fraction of relevant text in its constituent passages, or the fraction of relevant passages it contains. Finally, the induced cluster ranking is transformed to a ranking over passages by replacing each cluster with its passages while omitting repeats, as the clusters can overlap.

3.1.1 Learning to rank clusters. We apply a learning-to-rank (LTR) approach to rank the passage clusters; a cluster-query pair is represented by a feature vector. All features are aggregates of estimates of the cluster's constituent passages. Unless otherwise stated, the geometric mean is used for aggregation, following the findings about its merits in work on cluster-based document retrieval [12, 16, 18].

Query dependent features. The features **PsgInitRank** and **DocInitRank** are the geometric means of the reciprocal rank position of the clusters' constituent passages in $\mathcal{L}_{psg}$ and the set of their ambient documents in $\mathcal{L}_{doc}$, respectively. Past work demonstrated the merits of using relevance estimates of passages' ambient documents to rank the passages [17].

CIKM '19, November 3–7, 2019, Beijing, China

ACM ISBN 978-1-4503-6976-3/19/11…$15.00
https://doi.org/10.1145/3357384.3358087

Table 1: Datasets used for experiments.

Corpus	# of docs	Data	Avg doc. length	Queries
INEX	2,666,190	2009&2010	552	2009001-2009115, 2010001-2010107
AQUAINT	1,033,461	AQUAINT	436	N1-N100

Cluster priors. Inspired by work on using document relevance priors for document retrieval [2, 18], we use aggregates of query-independent passage-relevance priors as cluster features. **Ent** is the geometric mean of the entropy of the term distributions of the cluster's constituent passages; $-\sum_{w\in g} p_g^{MLE}(w)\log p_g^{MLE}(w)$ is the entropy of passage g. The **SW** feature is the geometric mean of the fraction of stopwords in the INQUERY list that appear in the cluster's passages. These two features are estimates, at the passage level, for content breadth [2]; hence, they presumably attest to passage relevance.

It was recently shown that relevant passages tend to be more similar to each other than to non-relevant passages [22]. Accordingly, the **InterPsgSim** feature is a measure of inter-passages similarities in cluster c; specifically, this is the geometric mean, over c's passages g, of $\frac{1}{k}\sum_{g_i\in c} Sim(g, g_i)$; k is the number of passages in c.

The **NumDoc** feature is the number of documents whose passages are members of the given cluster. We found (actual numbers are omitted due to space considerations) that a high number often indicates an increased amount of relevant text in the cluster.

We note that document-level features conceptually similar to PsgInitRank, Ent, SW and InterPsgSim were used in ClustMRF [18].

4 EXPERIMENTAL SETUP

Table 1 presents the datasets used for experiments. The INEX dataset is composed of English Wikipedia articles converted to flat representation (all XML markups were removed). The dataset includes *focused relevance judgments*: the relevant parts (character level) of relevant documents were marked. Passage retrieval was the subject of exploration of the focused retrieval tracks in 2009 and 2010 [1, 7].

The task in the Novelty tracks in TREC 2003 and TREC 2004 was to rank relevant sentences in documents [23, 24]. For consistency with the TREC 2003 track, we rank only sentences in relevant documents. There are, on average, 24 relevant documents per query; the average sentence length is 18.74. Sentences in the relevant-documents set are marked as relevant or non-relevant as a whole to the information need; i.e., these are *binary relevance judgments*.

The motivation for using these two datasets is two fold. First, they include annotations of relevant passages in documents. Second, the average fraction of relevant text in a relevant document for the INEX dataset is .40, and the average fraction of relevant sentences in a relevant document for the AQUAINT dataset is .37. Hence, focused retrieval can potentially be of much merit given that most text in relevant documents is marked as non-relevant.

4.1 Initial passage lists and clustering

We apply our cluster-based passage re-ranking method, termed **ClustPsg**, on three different initial passage lists, $\mathcal{L}_{psg}$, of n passages. The first initial list is produced using the QuerySimFuse method [4, 22], **QSF** in short, which scores a passage g using $(1-\lambda)\frac{Sim(q,g)}{\sum_{g'\in\mathcal{S}_{psg}} Sim(q,g')} + \lambda\frac{Sim(q,d_g)}{\sum_{d'\in\mathcal{L}_{doc}} Sim(q,d')}$, where $\mathcal{S}_{psg}$ is the set of all passages of documents in $\mathcal{L}_{doc}$ and λ is a free parameter.

The next two initial lists are retrieved by LTR-based methods. The first method, **MKS**, utilizes all the features proposed in [28] for retrieving answer sentences to non-factoid questions. The second LTR-based passage ranker, **MKS+**, is our proposed extension of MKS that uses two additional query-independent passage priors utilized above for our cluster ranking approach: (*i*) the entropy of the term distribution in the passage; and (*ii*) the fraction of stopwords in the INQUERY list that appear in the passage. In Section 5 we demonstrate the merits of MKS+ with respect to MKS.

To cluster the passages in an initially retrieved passage list, $\mathcal{L}_{psg}$, we employ a nearest-neighbors clustering method which was shown to be effective for cluster-based document retrieval [16, 18] and for grouping relevant passages together [22]. For each passage g in $\mathcal{L}_{psg}$ we create a cluster that contains g and the $k-1$ passages g_i ($g_i \neq g$) in $\mathcal{L}_{psg}$ that yield the highest $Sim(g, g_i)$.

LTR for passages and passage clusters. We train and apply the MKS and MKS+ methods on an initially retrieved passage list: the 1500[1] most highly ranked passages by the QSF method described above. Then, our ClustPsg method is trained and applied on the n ($\in \{50, 100\}$) passages most highly ranked by one of the three initial passage ranking methods: QSF, MKS and MKS+; n is relatively small following: (*i*) recommendations in work on cluster-based document re-ranking [10, 15, 16, 18], and (*ii*) a study of the cluster hypothesis for passages which showed that the hypothesis holds to a larger extent for short retrieved passage lists [22].

We use linear RankSVM [9] in all methods that utilize LTR. For the INEX dataset, to induce a relevance label for an item — a passage or a passage cluster[2] — we use the fraction of relevant text in the item, denoted **rFrac**. If $rFrac > 0$ the item is marked relevant, otherwise ($rFrac = 0$) the item is marked non-relevant. In addition, we define five relevance grades: 0: $rFrac = 0$; 1: $0 < rFrac < .25$; 2: $.25 \leq rFrac < .50$; 3: $.50 \leq rFrac < .75$; 4: $.75 \leq rFrac$. MKS, MKS+ and ClustPsg are trained using these graded relevance labels. The AQUAINT dataset contains sentence-level binary relevance judgments used to train MKS and MKS+. We train ClustPsg using the above described bucket-based approach for graded relevance judgments by replacing rFrac with the fraction of relevant sentences in the cluster (i.e., this is the precision in the cluster).

Additional experimental details. We used titles of topics for queries. Krovetz stemming was applied to queries and documents, and stopwords from the INQUERY list were removed only from queries. For the INEX dataset we use non-overlapping fixed-length windows of 300 terms for passages[3]. The INEX focused retrieval task [1, 7] explicitly forbids retrieving overlapping text segments as these are considered redundant. For the AQUAINT dataset we use the marked sentences for passages.

Passage retrieval performance over INEX is measured using iP[x]: the interpolated precision at early recall points $x \in \{.01, .1\}$ and MAiP: the mean (over 101 standard recall points) interpolated average precision [1, 7]. For AQUAINT we use the precision of

[1] We use the top-1500 retrieved passages for both INEX and AQUAINT following the INEX guidelines [1, 7].
[2] To this end, we concatenate the passages in a cluster.
[3] Our methods are not committed to a specific type of passages.

Table 2: The effectiveness of the 10 passages which are either the top-retrieved by QSF, MKS and MKS+, or belong to the optimal cluster (of size 10) among those created from QSF's top-retrieved passages. 'q', 'm' and 'p': statistically significant difference with QSF, MKS and MKS+, respectively. Boldface: the best result in a column. Bonferroni correction for multiple comparisons was applied.

	INEX		AQUAINT
	iP[.01]	rFrac	p@10
QSF	.485	.381	.624
MKS	.541	.403	.659
MKS+	.553	.413	$.682^{q}$
Optimal cluster	$\mathbf{.794}^{q}_{mp}$	$\mathbf{.686}^{q}_{mp}$	$\mathbf{.911}^{q}_{mp}$

the top 5 (p@5) and 10 (p@10) sentences, and MAP (@n). We use the two-tailed paired t-test with a 95% confidence level for testing statistical significance. Where needed, Bonferroni correction for multiple comparisons was applied. We also report the reliability of improvement (RI) [19]: $\frac{|Q_+|-|Q_-|}{|Q|}$; Q is the set of queries; Q_+ and Q_- are the sets of queries for which the MAiP (for INEX) or MAP (for AQUAINT) is higher and lower, respectively, than that of the initial passage ranking. The indri toolkit[4] was used for experiments.

We used leave-one-out cross validation over queries to set the free-parameter values of the QSF method, and to train the LTR methods as follows: we randomly split the train set to train (70%), and validation (30%); the latter was used to tune the hyper parameters. Once the values of the hyper parameters were set, we used all the queries (except the one left for test) to learn the final model. All feature values were min-max normalized on a per-query basis. To avoid overfitting while using ClustPsg over AQUAINT, we applied iterative backward feature elimination using the validation set: features whose removal resulted in the highest MAP improvements were eliminated. MAiP and MAP served as the optimization criteria for setting values of (hyper-) parameters in passage retrieval.

We set the Dirichlet smoothing parameter to 1000 for the initial document retrieval [29] and for measuring inter-passage similarities; for QSF, the smoothing parameter was set to values in $\{500, 1500, 2500\}$. The value of the interpolation parameter λ in QSF is in $\{0.1, 0.2, ..., 0.9\}$. RankSVM's regularization parameter is set to values in $\{10^{-4}, 10^{-2}, 10^{-1}\}$. We found that our ClustPsg method is more effective with clusters of $k = 10$ passages than $k = 5$. Hence, hereafter we use clusters of $k = 10$ passages.

5 EXPERIMENTAL RESULTS

We start by providing empirical motivation for devising cluster-based passage retrieval methods. Table 2 depicts the effectiveness of the 10 passages most highly ranked by QSF, MKS and MKS+, which are used to induce an initial passage ranking. (See Section 4.1 for details.) We cluster the top $n = 100$ passages retrieved by QSF using the nearest-neighbor clustering approach described above. The result is 100 overlapping clusters of 10 passages. We define the **optimal cluster** as the one with the maximal relevance degree, where the degree is rFrac for INEX and the ratio of relevant sentences for AQUAINT. (See Section 4.1 for details.)

[4]www.lemurproject.org

We see in Table 2 that if we were able to identify the optimal cluster and were to position its constituent passages at top ranks, the resultant retrieval effectiveness would have been dramatically better than that of highly effective passage retrieval methods that do not utilize inter-passage similarities. These findings echo those in work on using document clusters for document retrieval [11, 25].

Main result. We see in Table 3 that ClustPsg outperforms — often substantially — the three passage-ranking methods used to create the initial passage list on which it operates. The improvements are statistically significant in a majority of the relevant comparisons — 2 initial list sizes (n) $\times$ 4 (INEX)/3 (AQUAINT) evaluation measures $\times$ 3 initial lists. The always positive, and often high, RI performance numbers provide further support to ClustPsg's effectiveness with respect to the initial lists. We conclude that ClustPsg is highly effective in re-ranking different initially retrieved lists produced by highly effective passage ranking methods.

Passage-relevance priors. ClustPsg utilizes passage relevance priors. To ensure that its effectiveness over lists created by QSF and MKS is not solely due to these priors which are not utilized by QSF and MKS, we also applied it to the list created by MKS+— an extension of MKS which uses these priors. (See Section 3.1.1.) Table 3 shows that MKS+ outperforms MKS in all relevant comparisons; several of the improvements were found to be statistically significant. Furthermore, ClustPsg always outperforms MKS+, with the vast majority of improvements being statistically significant.

Feature analysis. Feature ablation analysis performed for ClustPsg where QSF was used to induce the initial list revealed the following. (Numbers are omitted due to space considerations and as they convey no additional insight.) For INEX, the removal of any feature results in a statistically significant performance drop with respect to at least one of the three evaluation measures (MAiP, iP[.01] and iP[.1]) and the two list sizes: $n \in \{50, 100\}$. The two features with the highest number of cases of statistically significant drops are SW and NumDoc. In addition, the use of all features outperforms in all relevant comparisons the removal of any single feature.

For the AQUAINT dataset, the only feature whose removal results in a statistically significant performance drop is Ent[5]. This result is aligned with our findings from above regarding the effectiveness of query-independent passage priors.

Cluster ranking. Heretofore, we studied the passage retrieval performance of ClustPsg which first ranks clusters, and then transforms the cluster ranking to passage ranking by replacing the clusters with their passages. We now turn to focus on the cluster ranking effectiveness of ClustPsg. Figure 1 presents the average fraction of relevant text (rFrac, left), and the average fraction of relevant sentences (p@10, right) in the 5 clusters of size $k = 10$ which are the most highly ranked by ClustPsg or by an Oracle that ranks the clusters by the true rFrac and p@10. Clusters were created from the $n = 100$ passages most highly ranked by QSF. The top ranked cluster of the Oracle is the optimal cluster. The performance of the top-10 ranked passages by QSF serves for reference.

Figure 1 shows that for both datasets, the performance of using the passages in each of the 5 clusters most highly ranked by

[5]Experiments show that using only Ent and NumDoc results in performance that is on par with that of using all features for AQUAINT.

Table 3: Main result. The effectiveness of ClustPsg in re-ranking different initially retrieved passage lists (QSF, MKS and MKS+). rFrac is computed over the top 10 ranked passages only for INEX as this dataset contains focused relevance judgments for passages in contrast to AQUAINT which provides binary relevance judgments for sentences. Statistically significant differences with the initial ranking ("init") are marked with 'i'. Boldface: the best result for evaluation measure per initial list.

		INEX										AQUAINT							
		$n = 50$					$n = 100$					$n = 50$				$n = 100$			
		MAiP	iP[.01]	iP[.1]	rFrac	RI	MAiP	iP[.01]	iP[.1]	rFrac	RI	MAP	p@5	p@10	RI	MAP	p@5	p@10	RI
QSF	init	.145	.557	.343	.381	–	.174	.569	.399	.381	–	.099	.652	.624	–	.161	.652	.624	–
	ClustPsg	**.154^{i}**	**.601^{i}**	**.371^{i}**	**.436^{i}**	.200	**.182**	**.629^{i}**	**.442^{i}**	**.439^{i}**	.267	**.109^{i}**	**.692**	**.698^{i}**	.380	**.183^{i}**	**.684**	**.703^{i}**	.660
MKS	init	.139	.600	.372	.403	–	.172	.604	.422	.403	–	.123	.664	.659	–	.211	.664	.659	–
	ClustPsg	**.150^{i}**	**.612**	**.395**	**.420**	.350	**.185^{i}**	**.627**	**.447**	**.410**	.300	**.127^{i}**	**.698**	**.687**	.340	**.222^{i}**	**.722^{i}**	**.710^{i}**	.480
MKS+	init	.147	.612	.386	.413	–	.180	.618	.437	.413	–	.128	.686	.682	–	.221	.686	.682	–
	ClustPsg	**.155^{i}**	**.647^{i}**	**.401**	**.428**	.250	**.191^{i}**	**.644**	**.453**	**.428**	.100	**.139^{i}**	**.754^{i}**	**.737^{i}**	.320	**.232^{i}**	**.744^{i}**	**.740^{i}**	.220

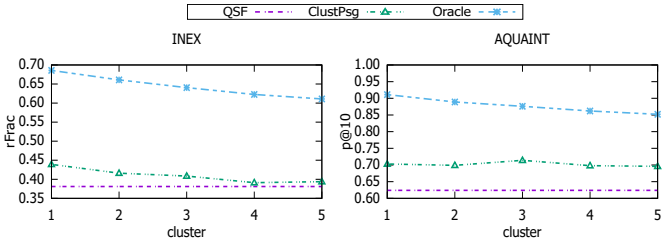

Figure 1: The average rFrac (INEX) and p@10 (AQUAINT) of passages in each of the 5 clusters of size $k = 10$ that are the most highly ranked by Oracle or ClustPsg. The performance for the top-10 ranked passages by QSF is presented for reference. Graphs are not to the same scale.

ClustPsg transcends that of QSF; the improvements for AQUAINT are more substantial than for INEX. These findings attest to the merits of our cluster ranking approach, although there is much room for improvement as implied by the Oracle performance.

6 CONCLUSIONS

We presented a novel passage retrieval approach that is based on ranking clusters of similar passages using a learning-to-rank approach. Empirical evaluation demonstrated the clear merits of the approach with respect to highly effective passage ranking methods.

Acknowledgments. We thank the reviewers for their comments and Anna Shtok for all her help. This paper is based upon work supported in part by the German Research Foundation (DFG) via the German-Israeli Project Cooperation (DIP, grant DA 1600/1-1).

REFERENCES

[1] Paavo Arvola, Shlomo Geva, Jaap Kamps, Ralf Schenkel, Andrew Trotman, and Johanna Vainio. 2011. Overview of the INEX 2010 ad hoc track. In *Comparative Evaluation of Focused Retrieval*. 1–32.
[2] Michael Bendersky, W Bruce Croft, and Yanlei Diao. 2011. Quality-biased ranking of web documents. In *Proc. of WSDM*. 95–104.
[3] David Buffoni, Nicolas Usunier, and Patrick Gallinari. 2010. Lip6 at INEX: OWPC for ad hoc track. In *Focused Retrieval and Evaluation*. 59–69.
[4] James P. Callan. 1994. Passage-Level Evidence in Document Retrieval. In *Proc. of SIGIR*. 302–301.
[5] Ruey-Cheng Chen, Evi Yulianti, Mark Sanderson, and W Bruce Croft. 2017. On the Benefit of Incorporating External Features in a Neural Architecture for Answer Sentence Selection. In *Proc. of SIGIR*. 1017–1020.
[6] Daniel Cohen and W Bruce Croft. 2016. End to end long short term memory networks for non-factoid question answering. In *Proc. of ICTIR*. 143–146.
[7] Shlomo Geva, Jaap Kamps, Miro Lethonen, Ralf Schenkel, James A Thom, and Andrew Trotman. 2010. Overview of the INEX 2009 ad hoc track. In *Focused retrieval and evaluation*. 4–25.
[8] Nick Jardine and C. J. van Rijsbergen. 1971. The use of hierarchic clustering in information retrieval. *Information storage and retrieval* 7, 5 (1971), 217–240.
[9] Thorsten Joachims. 2006. Training linear SVMs in linear time. In *Proc. of KDD*. 217–226.
[10] Oren Kurland. 2009. Re-ranking search results using language models of query-specific clusters. *Information Retrieval* 12, 4 (2009), 437–460.
[11] Oren Kurland and Carmel Domshlak. 2008. A rank-aggregation approach to searching for optimal query-specific clusters. In *Proc. of SIGIR*. 547–554.
[12] Oren Kurland and Eyal Krikon. 2011. The opposite of smoothing: a language model approach to ranking query-specific document clusters. *Journal of Artificial Intelligence Research* 41 (2011), 367–395.
[13] Oren Kurland and Lillian Lee. 2004. Corpus structure, language models, and ad hoc information retrieval. In *Proc. of SIGIR*. 194–201.
[14] Xiaoyong Liu and W Bruce Croft. 2004. Cluster-based retrieval using language models. In *Proc. of SIGIR*. 186–193.
[15] Xiaoyong Liu and W Bruce Croft. 2006. *Experiments on retrieval of optimal clusters*. Technical Report. Technical Report IR-478, Center for Intelligent Information Retrieval (CIIR), University of Massachusetts.
[16] Xiaoyong Liu and W Bruce Croft. 2008. Evaluating text representations for retrieval of the best group of documents. In *Proc. of ECIR*. 454–462.
[17] Vanessa Graham Murdock. 2006. *Aspects of sentence retrieval*. Ph.D. Dissertation. University of Massachusetts Amherst.
[18] Fiana Raiber and Oren Kurland. 2013. Ranking document clusters using markov random fields. In *Proc. of SIGIR*. 333–342.
[19] Tetsuya Sakai, Toshihiko Manabe, and Makoto Koyama. 2005. Flexible pseudo-relevance feedback via selective sampling. *TALIP* 4, 2 (2005), 111–135.
[20] Gerard Salton, James Allan, and Chris Buckley. 1993. Approaches to passage retrieval in full text information systems. In *Proc. of SIGIR*. 49–58.
[21] Aliaksei Severyn and Alessandro Moschitti. 2015. Learning to rank short text pairs with convolutional deep neural networks. In *Proc. of SIGIR*. 373–382.
[22] Eilon Sheetrit, Anna Shtok, Oren Kurland, and Igal Shprincis. 2018. Testing the Cluster Hypothesis with Focused and Graded Relevance Judgments. In *Proc. of SIGIR*. 1173–1176.
[23] Ian Soboroff. 2004. Overview of the TREC 2004 Novelty Track. In *Proc. of TREC*.
[24] Ian Soboroff and Donna Harman. 2003. Overview of the TREC 2003 Novelty Track. In *Proc. of TREC*. 38–53.
[25] Anastasios Tombros, Robert Villa, and C. J. Van Rijsbergen. 2002. The effectiveness of query-specific hierarchic clustering in information retrieval. *Information processing & management* 38, 4 (2002), 559–582.
[26] Ellen M. Voorhees. 1985. The cluster hypothesis revisited. In *Proc. of SIGIR*. 188–196.
[27] Liu Yang, Qingyao Ai, Jiafeng Guo, and W Bruce Croft. 2016. aNMM: Ranking short answer texts with attention-based neural matching model. In *Proc. of CIKM*. 287–296.
[28] Liu Yang, Qingyao Ai, Damiano Spina, Ruey-Cheng Chen, Liang Pang, W Bruce Croft, Jiafeng Guo, and Falk Scholer. 2016. Beyond factoid QA: Effective methods for non-factoid answer sentence retrieval. In *Proc. of ECIR*. 115–128.
[29] Chengxiang Zhai and John Lafferty. 2001. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proc. of SIGIR*. 334–342.