Predicting the Size of Candidate Document Set for Implicit Web Search Result Diversification

Yasar Baris Ulu and Ismail Sengor Altingovde(✉)

Middle East Technical University, Ankara, Turkey
{yasar.ulu,altingovde}@ceng.metu.edu.tr

Abstract. Implicit result diversification methods exploit the content of the documents in the *candidate set*, i.e., the initial retrieval results of a query, to obtain a relevant and diverse ranking. As our first contribution, we explore whether recently introduced word embeddings can be exploited for representing documents to improve diversification, and show a positive result. As a second improvement, we propose to automatically predict the size of candidate set on *per query* basis. Experimental evaluations using our BM25 runs as well as the best-performing ad hoc runs submitted to TREC (2009–2012) show that our approach improves the performance of implicit diversification up to 5.4% wrt. initial ranking.

1 Introduction

Diversification of web search results is a well-known approach to handle queries that are ambiguous, underspecified, or including multiple aspects [23]. Diversification methods in the literature are broadly categorized as *implicit* or *explicit*. The implicit methods essentially make use the documents in the *candidate set*, i.e., the initial retrieval results for the query. In contrary, explicit methods exploit the knowledge of query aspects, which is usually inferred from a topic taxonomy [1] or query log [20]. The exhaustive experiments in the literature confirm that the latter type of additional information is very useful, as explicit methods consistently outperform the implicit ones [10,23]. This finding does not render the implicit diversification less valuable, as in many scenarios the query aspects are not readily available or not easy to infer (e.g., for the rare queries in Web search) [15], but rather calls for approaches to improve their performance.

Our contributions in this paper are two-fold: First, we re-visit the implicit diversification using recently introduced word embeddings, and show that using the latter to represent documents is superior to traditional vector space model (with tf-idf weights). However, we observe that using either type of representations, implicit diversification can hardly beat even the initial –non-diversified– ranking (confirming the observations of [10]). These findings, obtained using the best-performing trade-off parameter λ (i.e., used to tune the weight of relevance vs. diversity in a ranking, as explained in Sect. 2) and a candidate set size $N = 100$ documents (an ad hoc yet intuitive choice made in several earlier works

J. M. Jose et al. (Eds.): ECIR 2020, LNCS 12036, pp. 410–417, 2020.
https://doi.org/10.1007/978-3-030-45442-5_51

[10,15,21]), imply that a more customized tuning of parameters is required for implicit diversification. An earlier work also recognized such a need for selective diversification, and proposed to predict the trade-off parameter λ on a per query basis [21]. However, the second parameter that is equally important, the size of the candidate set (N), on which diversification is applied, is left unexplored. We believe that for the implicit methods, where the evidence used for diversification is based solely on the content of the documents, tuning the candidate set size is crucial: a small set with documents relevant to only the main query might not cover any alternative intents (aspects) of the query, while a too large set is likely to include several noisy documents and hence, mislead the implicit methods.

In the light of above discussion, as our second contribution, we propose to predict the candidate set size, N, on a *per query* basis, to achieve a more customized diversification of query results. To this end, we employ a rich set of features that capture the retrieval effectiveness (i.e., query performance predictors [6,14,21,24]) and pairwise similarity of documents (using alternative document representations). All features are computed over the candidate set at several rank-cutoffs (actually, from 10 to 100 with a step size of 10). Before the diversification for a query, we predict N (as well as λ, as in [21]), based on these features.

In our evaluations, we employ MMR [5] as a representative implicit method, as it is widely employed in the literature, has fewer parameters to tune and fast. Our findings over the homemade runs (based on the BM25 function) as well as the representative runs from the previous TREC campaigns (2009 to 2012) are promising. By predicting the parameters on a per query basis and employing word embeddings, implicit diversification can outperform the non-diversified baselines (with relative gains up to 5.4%), as well as the diversification baseline with parameters based on majority voting (with even larger gains).

Related Work. Word embeddings are employed for various tasks related to diversification of search results (such as expanding the queries in tweet search [16], generating diversified query expansions [13], inferring query aspects [25]). However, as far as we know, the impact of employing word embeddings to represent the documents for implicit diversification has not been explored.

Earlier works proposed several implicit diversification methods [23]. While most of these works employ a fixed N, such as 50 or 100 (e.g., [7,10,15,16,18]), a few works (such as [12]) identified N (and/or λ) over a training set, (i.e., as our Majority Voting baseline presented in Sect. 4). Santos et al. [21] suggested a selective diversification approach, where only λ is predicted for each query, using kNN approach. None of these works predict the candidate set size on a per query basis for result diversification. Finally, in an *unpublished* thesis work [2], preliminary experiments for candidate set size prediction are presented for explicit diversification. In contrary, our work addresses implicit diversification, which requires features that capture inter-document similarity and are not used in the latter work. Furthermore, we predict both parameters N and λ (consecutively) using the same set of features, which is different than the setup in [2].

2 Document Representation for Implicit Diversification

There are several implicit methods in the literature [23], and in this work, we use MMR [5] as a representative method due to two reasons. First, being a simple, intuitive and efficient method, MMR is employed as a baseline and/or representative approach in a large number of works (e.g., [10,21,26,27]). Secondly, we conducted preliminary experiments with some other candidates (namely, MSD [8], MMC and GNE [26]) and did not observe meaningful performance differences wrt. MMR. Actually, only GNE produced slightly better results, however as it is based on a greedy local search, its execution time is considerably longer than MMR. Therefore, we proceed with MMR as a representative method. In what follows, we first briefly review MMR and then discuss how word embeddings are employed to represent documents in this context.

Maximal Marginal Relevance (MMR) [5]**.** This is a greedy best-first search approach that aims to choose the document that maximizes the following scoring function in each iteration.

$$MMR(d,q,S) = \lambda * rel(d,q) - (1-\lambda) * \max_{d_j \in S} sim(d,d_j) \tag{1}$$

Given a query q and a candidate result set D of size N, MMR constructs a diversified ranking S of size s (typically, $s < N$) as follows. At first, the document with the highest relevance score is inserted into S. Then, in each iteration, the document that maximizes Eq. 1 is added to S. While computing the score of a document $d \in D - S$, its relevance to q, denoted as rel(q,d), is discounted by the d's maximum similarity the previously selected documents into S. In Eq. 1, $sim(d,d_j)$ is typically computed by the Cosine distance of documents that are represented as tf-idf weighted vectors. Finally, λ is a trade-off parameter to balance the relevance and diversity in the final result set S.

Word Embeddings for Document Representation. In this preliminary work, we take a simplistic approach and represent each document d based on the embedding vectors of their terms $t \in d$. In the literature, different approaches are proposed for this purpose, such as computing the minimum, maximum or average for each dimension of the embedding vectors over all terms in the document [4]. The aggregation operation can also be weighted, e.g., by IDF values of the terms. In this work, based on our preliminary experiments, we represent each document as a concatenation of minimum and maximum vectors, as in [4]. Thus, $sim(d,d_j)$ in Eq. 1 is computed as the Cosine distance between the latter type of vectors.

3 Predicting the Candidate Set Size

Implicit diversification methods do not exploit any external information (in contrary to their explicit competitors) and their diversification decision is essentially based on the content of the documents. While the size of the candidate set, N, is an important parameter for all diversification approaches, it is more crucial for the implicit methods: In particular, a very large candidate set is likely to have more irrelevant documents towards the tail of the ranking, yet such documents -yielding smaller similarity to the relevant ones that are ranked higher- are more likely to be scored high by Eq. 1, and hence, would decrease the relevance of the final ranking. In contrary, setting N too small will risk to have any diverse document in the final ranking, and hence, reduce the diversity. This implies that the value of N should be determined on a *per query* basis.

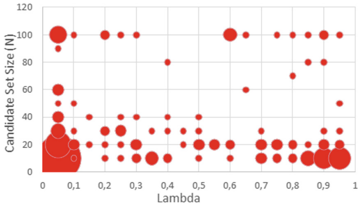

Fig. 1. Best-performing (N, λ) pair over BM25 runs for 198 TREC topics.

Table 1. List of features computed over each ranking.

Feature	Description	Count
scoreRatio [17]	Ratio of top to last document's score	10
scoreMean [14]	Mean of scores in document set	10
scoreMeanDecrease	Decrease in mean of scores in document set	9
scoreMedian	Median of scores in document set	10
scoreStandardDev [14,24]	Standard deviation of scores	10
scoreVariance	Variance of scores	10
coefficientOfVariation	Coefficient of variation	10
NQC [24]	Scores of Normalized Query Commitment	10
PairwiseTfIdfSimilarity	Pairwise (td-idf vector) similarity (min, max, avg)	30
PairwiseWESimilarity	Pairwise (WE vector) similarity (min, max, avg)	30
PairwiseEntitySimilarity	Pairwise (Entity list) similarity (min, max, avg)	30

As a further motivation, consider Fig. 1, a bubble chart that presents the best-performing (λ, N) pairs (for diversification with MMR) for 198 queries used in the TREC Diversity track (2009-2012). The initial runs are obtained using BM25 and the size of the bubble denotes the frequency of a pair. Clearly, there is no single λ or N that optimizes all queries, and indeed, values are quite scattered.

In this work, we propose to predict N. Since our approach requires determining an optimal cut-off point in the candidate ranking, we compute each of the following features over a ranking of top-n documents, where $n \in \{10, 20, \ldots N\}$. Our features (shown in Table 1) can be grouped into two categories. The first group are based on well-known query performance predictors [6,14,24], and as in [21], they are intended to capture the quality of the ranking (i.e., in terms of relevance). The second group of features is intended to reflect the diversity of a ranking. To this end, we propose to compute the pairwise similarity of the documents, and aggregate these scores using minimum, maximum and average functions. While computing such similarities, we employ both tf-idf and word embedding based document representations (as discussed in Sect. 2). Finally, as entities are found helpful in earlier works [21], as a third option, we represent each document based on the named entities it contains (see Sect. 4 for details).

A training instance for a query includes a vector of these features computed for each top-n ranking ($n \in \{10, 20, \ldots 100\}$), i.e., including 10 variants for each feature. For each query, we apply parameter sweeping over $N \in \{10, 20, \ldots, 100\}$ and $\lambda \in \{0.05, 1.0, \ldots, 0.95\}$, and determine the best performing values (for diversification with MMR), to serve as the ground truth (categorical) class labels. For a test query, we first predict N as the class label. Next, we predict λ (as in [21]) by using the aforementioned features and the *predicted* N value, as an additional feature (i.e., as in the *classifier chain* approach in [19]). Our preliminary experiments with Weka [9] revealed that best results are obtained using a lazy learning algorithm, kNN (as in [21]). Thus, for a test query, N (and then, λ) are predicted using majority voting among the class labels of its k neighbors.

Table 2. Diversification performance (α-nDCG@10) of MMR using TF-IDF vectors ($\text{MMR}_{\text{TfIdf}}$) vs. word embedding vectors ($\text{MMR}_{\text{WordEmb}}$) for BM25 and TREC runs.

	BM25 runs			TREC runs		
Topic set	NonDiv	$\text{MMR}_{\text{TfIdf}}$	$\text{MMR}_{\text{WordEmb}}$	NonDiv	$\text{MMR}_{\text{TfIdf}}$	$\text{MMR}_{\text{WordEmb}}$
2009	0,2520	0,2360	**0,2531**	0,2530	0,2533	**0,2544**
2010	0,2427	0,2461	**0,2573**	0,3716	0,3634	**0,3718**
2011	0,4680	0,4581	**0,4693**	0,5312	0,5312	**0,5315**
2012	**0,3218**	0,2911	0,3215	0,4942	0,4926	**0,4962**

4 Evaluation Setup and Results

Dataset and Runs. We employ topic sets that are introduced in “Diversity Task” of TREC Web Track between 2009 and 2012. Each topic set includes 50 queries (except 2010, which has 48), their official aspects and the relevance judgments at the aspect level. We have two types of runs created as follows. First, we used our own retrieval system to index ClueWeb09 collection Part-B (with 50M documents) and then, for each topic set, we generated an an initial ranking of top-100 documents per query, using the well-known BM25 function. These are referred to as `BM25 runs`. Secondly, we selected the best-performing run (again, on ClueWeb-B) submitted to ad hoc retrieval track of TREC (2009-2012). As in [3, 10], as these runs are not diversified, they can safely serve as initial retrieval results (with various ranking methods beyond BM25), and best-performing run is the one that yields the highest α-nDCG@10 score. These are referred to as `TREC runs`. The ids of the selected runs for each year are as follows: Ucdsift (2009), Uogtr (2010), Srchvs11b (2011) and Qutparabline (2012).

Experimental Parameters. To represent documents with embeddings (Sect. 2), we employ the pre-trained Glove word vectors (with 100 dimensions) for 400K words. Tf-idf vectors are based on the document and collection statistics, as usual. To extract the named entities in documents (to compute some features in Table 1), we used an entity list (of people, locations, etc.) from DBpedia together with the dictionary-based entity recognition approach of [22].

For prediction of N and λ, kNN algorithm is applied with 5-fold cross validation over each run. We employ the best performing k. We observed that, especially for the TREC runs, setting k to 1 is adequate in several cases. The size of the final ranking S is 10, and we report α-nDCG@10 scores.

Results for Document Representation Experiments. As our first research question, we focus on the impact of using word embeddings for representing documents during diversification. In this experiment, we set $N = 100$ as typical (e.g., [10,21]), and report results for the best-performing λ for each run.

Table 2 shows that the performance of MMR$_{\text{TfIdf}}$ is inferior to the non-diversified ranking for the majority of the cases, i.e., its application yields even less diverse rankings. This finding confirms [10], where MMR is rarely found to provide any significant gains. In contrary, MMR$_{\text{WordEmb}}$ outperforms the MMR$_{\text{TfIdf}}$ in all cases (underlined cases in Table 2 are statistically significant using paired t-test at 0.05 confidence level). Furthermore, MMR$_{\text{WordEmb}}$ is also superior to the non-diversified baseline for seven (out of eight) runs, but with a small difference in most cases. These findings indicate that word embeddings are useful for MMR, but not adequate for impressive diversification performance.

Table 3. Diversification performance (α-nDCG@10) of MMR$_{\text{WordEmb}}$ with parameters obtained via Orcl$_{100,\lambda}$ (best λ [21]), Orcl$_{N,\lambda}$ (best N and λ), Majority Voting and kNN.

	BM25 runs					TREC runs				
TSet	NonDiv	Orcl$_{100,\lambda}$	Orcl$_{N,\lambda}$	MV	kNN	NonDiv	Orcl$_{100,\lambda}$	Orcl$_{N,\lambda}$	MV	kNN
2009	0,2520	0,2917	0,3044	0,2469	**0,2612**	0,2530	0,2853	0,3075	0,2562	**0,2589**
2010	0,2427	0,2876	0,3137	0,2443	**0,2554**	0,3716	0,4020	0,4032	0,3595	**0,3768**
2011	0,4680	0,4888	0,5194	0,4279	**0,4750**	0,5312	0,5468	0,5507	0,5284	**0,5379**
2012	0,3218	0,3270	0,4452	0,3257	**0,3392**	**0,4942**	0,5066	0,5102	0,4772	0,4890

Results for Predicting Parameters. We evaluate the performance of predicting the parameters N and λ only for MMR$_{\text{WordEmb}}$ (due to the aforementioned findings). Table 3 reports the results both for BM25 runs and TREC best runs. We provide α-nDCG@10 scores for non-diversified (NonDiv) ranking, as well as two oracle approaches (discussed later). The traditional baseline Majority Voting (MV) sets N and λ to the most frequent value in training folds, respectively.

We make several observations from Table 3. First, the MV baseline cannot beat the initial non-diversified ranking (NonDiv) for several cases. When kNN is applied to predict the parameters N and λ, the diversification performance is superior to MV baseline (in all cases), and outperforms the NonDiv ranking in all runs but one (i.e., *Qutparabline* from 2012). For BM25 runs, kNN based diversification provides relative gains wrt. the non-diversified ranking ranging from 1.5% to 5.4%. For more competitive TREC runs, the relative gains are in the range 1.2% to 2.3% (except the 2012 run, where there is a relative degradation of 1%). Given that the latter runs are employing sophisticated approaches far beyond BM25, our findings are promising. Note that, in some cases (underlined in Table 3) improvements wrt. MV are statistically significant (using paired t-test at 0.05 confidence level), while there is no significant degradation wrt. MV or NonDiv. The latter is a contrary and encouraging finding in comparison to [10], where MMR is observed to yield only significant degradation in most cases.

Table 3 also reports two oracle approaches: In $Oracle_{N,\lambda}$, the best-performing N and λ is used for each query. In $Oracle_{100,\lambda}$, we fixed N as 100, and only employed the best-performing λ. The latter oracle aims to provide an upper-bound for predicting only λ as in [21], while the former one presents the upper-bound for our approach, predicting both parameters. We see that our approach can yield higher performance (for all runs), and in certain cases, the possible gain is considerably larger than that of predicting only λ. As a further observation, a comparison of kNN performance to $Oracle_{N,\lambda}$ indicates that there is still room for improvement, i.e., if better prediction of parameters can be achieved.

Conclusion. We showed that implicit diversification benefits from word embeddings based document representation, but it still yields rather small gains in diversification effectiveness wrt. the initial ranking. As a remedy, we proposed to predict N, the candidate set size, using a rich set of features. By predicting N (together with λ, as in [21]) and employing word embeddings, we achieved better diversification. In our future work, we plan to use document embeddings (e.g., Doc2Vec [11]) for document representation. We will also exploit additional (e.g., click-based) features for better prediction of the diversification parameters.

Acknowledgements. This work is partially funded by The Scientific and Technological Research Council of Turkey (TÜBİTAK) grant 117E861 & TÜBA GEBIP (2016) award.

References

1. Agrawal, R., Gollapudi, S., Halverson, A., Ieong, S.: Diversifying search results. In: Proceedings of WSDM, pp. 5–14. ACM (2009)
2. Akcay, M.: Analyzing and boosting the performance of explicit result diversification methods for web search. Master's thesis, Middle East Technical University (METU) (2016)
3. Akcay, M., Altingovde, I.S., Macdonald, C., Ounis, I.: On the additivity and weak baselines for search result diversification research. In: Proceedings of ICTIR, pp. 109–116 (2017)
4. Boom, C.D., Canneyt, S.V., Demeester, T., Dhoedt, B.: Representation learning for very short texts using weighted word embedding aggregation. Pattern Recogn. Lett. **80**, 150–156 (2016)

5. Carbonell, J.G., Goldstein, J.: The use of MMR, diversity-based reranking for reordering documents and producing summaries. In: Proceedings of SIGIR, pp. 335–336 (1998)
6. Carmel, D., Kurland, O.: Query performance prediction for IR. In: Proceedings of SIGIR, pp. 1196–1197 (2012)
7. Dang, V., Croft, W.B.: Diversity by proportionality: an election-based approach to search result diversification. In: Proceedings of SIGIR, pp. 65–74 (2012)
8. Gollapudi, S., Sharma, A.: An axiomatic approach for result diversification. In: Proceedings of WWW, pp. 381–390 (2009)
9. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: an update. SIGKDD Explor. **11**(1), 10–18 (2009)
10. Kharazmi, S., Scholer, F., Vallet, D., Sanderson, M.: Examining additivity and weak baselines. Trans. Inf. Syst. **34**(4), 23 (2016)
11. Le, Q.V., Mikolov, T.: Distributed representations of sentences and documents. In: Proceedings of ICML, pp. 1188–1196 (2014)
12. Limsopatham, N., McCreadie, R., Albakour, M., Macdonald, C., Santos, R.L.T., Ounis, I.: University of glasgow at TREC 2012: experiments with terrier in medical records, microblog, and web tracks. In: Proceedings of TREC (2012)
13. Liu, X., Bouchoucha, A., Sordoni, A., Nie, J.: Compact aspect embedding for diversified query expansions. In: Proceedings of AAAI, pp. 115–121 (2014)
14. Markovits, G., Shtok, A., Kurland, O., Carmel, D.: Predicting query performance for fusion-based retrieval. In: Proceedings of CIKM, pp. 813–822 (2012)
15. Naini, K.D., Altingovde, I.S., Siberski, W.: Scalable and efficient web search result diversification. ACM Trans. Web **10**(3), 15:1–15:30 (2016)
16. Onal, K.D., Altingovde, I.S., Karagoz, P.: Utilizing word embeddings for result diversification in tweet search. In: Proceedings of AIRS, pp. 366–378 (2015)
17. Ozdemiray, A.M., Altingovde, I.S.: Query performance prediction for aspect weighting in search result diversification. In: Proceedings of CIKM, pp. 1871–1874 (2014)
18. Ozdemiray, A.M., Altingovde, I.S.: Explicit search result diversification using score and rank aggregation methods. JASIST **66**(6), 1212–1228 (2015)
19. Read, J., Pfahringer, B., Holmes, G., Frank, E.: Classifier chains for multi-label classification. Mach. Learn. **85**(3), 333–359 (2011)
20. Santos, R.L.T., Macdonald, C., Ounis, I.: Exploiting query reformulations for web search result diversification. In: Proceedings of WWW, pp. 881–890 (2010)
21. Santos, R.L.T., Macdonald, C., Ounis, I.: Selectively diversifying web search results. In: Proceedings of CIKM, pp. 1179–1188 (2010)
22. Santos, R.L.T., Macdonald, C., Ounis, I.: Voting for related entities. In: Proceedings of RIAO, pp. 1–8 (2010)
23. Santos, R.L.T., Macdonald, C., Ounis, I.: Search result diversification. Found. Trends Inf. Retrieval **9**(1), 1–90 (2015)
24. Shtok, A., Kurland, O., Carmel, D., Raiber, F., Markovits, G.: Predicting query performance by query-drift estimation. ACM Trans. Inf. Syst. **30**(2), 11 (2012)
25. Ullah, M.Z., Shajalal, M., Chy, A.N., Aono, M.: Query subtopic mining exploiting word embedding for search result diversification. In: Proceedings of AIRS, pp. 308–314 (2016)
26. Vieira, M.R., et al.: On query result diversification. In: Proceedings of ICDE, pp. 1163–1174 (2011)
27. Zuccon, G., Azzopardi, L., Zhang, D., Wang, J.: Top-k retrieval using facility location analysis. In: Proceedings of ECIR, pp. 305–316 (2012)