Search Result Diversification with Guarantee of Topic Proportionality

Sheikh Muhammad Sarwar, Raghavendra Addanki, Ali Montazeralghaem, Soumyabrata Pal, and James Allan
College of Information and Computer Sciences, University of Massachusetts Amherst
{smsarwar,raddanki,montazer,spal,allan}@cs.umass.edu

ABSTRACT

Search result diversification based on topic proportionality considers a document as a bag of weighted topics and aims to reorder or down-sample a ranked list in a way that maintains topic proportionality. The goal is to show the topic distribution from an ambiguous query at all points in the revised list, hoping to satisfy all users in expectation. One effective approach, PM-2, greedily selects the best topic that maintains proportionality at each ranking position and then selects the document that best represents that topic. From a theoretical perspective, this approach does not provide any guarantee that topic proportionality holds in the small ranked list. Moreover, this approach does not take query-document relevance into account. We propose a Linear Programming (LP) formulation, LP-QL, that maintains topic proportionality and simultaneously maximizes relevance. We show that this approach satisfies topic proportionality constraints in expectation. Empirically, it achieves a 5.5% performance gain (significant) in terms of α-NDCG compared to PM-2 when we use LDA as the topic modelling approach. Furthermore, we propose LP-PM-2 that integrates the solution of LP-QL with PM-2. LP-PM-2 achieves 3.2% performance gain (significant) over PM-2 in terms of α-NDCG with term based topic modeling approach. All of our experiments are based on a popular web document collection, ClueWeb09 Category B, and the queries are taken from TREC Web Track's diversity task.

KEYWORDS

search result diversification, topic modeling, linear programming

ACM Reference Format:
Sheikh Muhammad Sarwar, Raghavendra Addanki, Ali Montazeralghaem, Soumyabrata Pal, and James Allan. 2020. Search Result Diversification with Guarantee of Topic Proportionality. In *Proceedings of the 2020 ACM SIGIR International Conference on the Theory of Information Retrieval (ICTIR '20), September 14–17, 2020, Virtual Event, Norway*. ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3409256.3409839

1 INTRODUCTION

Search Result Diversification (SRD) is an effective component of a web search engine, particularly when a user's information need is ambiguous so has more than one interpretation [2, 20]. It is a problem with a long standing importance to the World Wide Web (WWW), whether it is applied in search [1], recommendation [22, 28] or non-factoid question answering [23]. Typical users express their information need using very few keywords and expect the search engine to provide the specific facet of information they are interested in. On the other hand, search results retrieved against short search queries have many different aspects and traditional web search interfaces only show a very small subset – *i.e.,* the top-k ranked documents – to a user. As a result, one primary goal of a diversified ranking engine is to provide at least one relevant document in the top-k search results for every user [9].

To achieve the above mentioned goal an SRD algorithm penalizes redundancy and promotes novelty in a ranked list. Generally, all SRD algorithms re-rank the top n documents retrieved by a non-diversified ranker such as Query Likelihood (QL) and output the top k. The objective is to ensure that each document in the top k is dissimilar to or covers a different aspect of the query compared to other documents. To achieve this objective there exists two categories of SRD algorithms: *implicit*, and *explicit*. Implicit SRD approaches generally define a similarity metric to achieve the objective, while explicit approaches model a document as a vector of query topics and maximizes topic coverage in a ranked list. Usually explicit SRD algorithms are more effective compared to implicit ones, but their success depends on two factors: query topic identification, and query topic coverage in a ranked list.

Dang and Croft took a further step beyond that objective and proposed that coverage of a query topic in k documents should be proportional to its popularity in the set of $n \gg k$ documents retrieved against the query [12]. They proposed PM-2, a topic proportionality based diversification approach that outperformed the vast majority of the implicit and explicit diversification approaches [13]. They also proposed a term level query aspect identification technique that performs the best when used with PM-2.

PM-2 is an iterative algorithm: it re-ranks a given ranked list by selecting documents one by one starting from the beginning of the ranked list. Its document selection process is based on topic proportionality. At each ranking position PM-2 selects a topic that has been covered less in the previous ranks compared to the number of times it should be covered in the final ranked list. In this way it tries to reward prominent query topics. However, PM-2 does not provide any theoretical guarantee on topic proportionality and it might fail to present less popular topics in the final ranked list. The results from Dang and Croft show that PM-2 suffers from low Subtopic Recall (S-Recall), a metric that indicates how many of the subtopics or aspects of a query are covered in a search result list [13]. However, Dang and Croft also showed that PM-2 excels in

ICTIR '20, September 14–17, 2020, Virtual Event, Norway

ACM ISBN 978-1-4503-8067-6/20/09…$15.00
https://doi.org/10.1145/3409256.3409839

Precision-IA – a metric that indicates precision across all aspects of a query. It shows that PM-2 is an effective ranker of the documents for each topic.

That weakness of PM-2 motivated us to propose a model that comes with a guarantee of topic proportionality. Intuitively, such a model would have higher S-Recall. Thus we propose a set cover formulation of the diversification problem that considers documents as sets, their corresponding Query Likelihood (QL) scores as utilities of those sets, and topics as elements. Moreover, our set cover formulation models proportionality using proportionality constraints. We propose a solution to the set cover problem by formulating it as a Linear Program (LP) and then taking a randomized rounding approach. As the set cover solution is not a ranked list, we rank the set elements using query likelihood and call the result LP-QL. Our solution set is generated by maximizing relevance; we prove that proportionality constraints are maintained in expectation. Empirically, we achieve better S-Recall and show that further gain can be achieved by integrating this solution into the PM-2 framework.

Our proposed extension to PM-2, LP-PM-2, outperforms the original using the set cover solution from LP-QL. LP-PM-2 combines the proportional topic coverage guarantee from LP-QL and ranking effectiveness from PM-2. The contributions of this paper are:

- We model topic proportionality based search result diversification as a set cover problem with proportionality constraints. It is a framework under which proportionality based diversification can be theoretically studied.
- We reduce the set cover problem to a linear program and propose a randomized rounding scheme to solve it. Theoretically, we show that our approach satisfies the proportionality constraints in expectation.
- We propose LP-PM-2 that ranks the set cover solution obtained from our randomized rounding approach using PM-2. Empirically, LP-PM-2 achieves significant gain in terms of α-NDCG in comparison to PM-2.

2 RELATED WORK

In this section, we provide an overview of existing search result diversification approaches and topic models used to find query sub-topics for diversification.

2.1 Diversification Models

There are two categories of search result diversification models: *implicit* and *explicit*. Models that do not explicitly attempt to identify the topics or the features by which diversification happens are referred to as implicit models [5, 21]. Our study focuses on explicit diversification models and in the following sections we provide an overview of the topic focusing on approaches and evaluation techniques.

Explicit diversification approaches model a document as a bag of query sub-topics. These models are generally successful when topic annotation is available from any of the three sources: aspects generated from a larger taxonomy; human generated query aspects; and a list of aspects obtained from a commercial search engine [12]. In the absence of oracle topic annotations only a few of these approaches have been proven to be successful. Explicit diversification models have both unsupervised and supervised variants.

Unsupervised Explicit Diversification Approaches. Agrawal et al. [1] proposed an explicit diversification algorithm by modeling document topics using a topic taxonomy. They considered two documents similar if they are classified into one or more common categories in the topic taxonomy. Another model, xQuAD, achieves diversity in a ranked list by penalizing redundancy at every rank [21]. It is a greedy algorithm that selects a document at a specific rank based on four criteria: *topic importance*, *document coverage based on topic-document relevance*, *document novelty* and *document relevance*. xQuAD is a trade-off between relevance and novelty in the same way a popular implicit diversification model, MMR [6] is, but it assumes explicit topic representation. Another explicit diversification algorithm, PM-2, achieves diversity by maintaining topic proportionality. The proportionality constraint states that if a topic is covered a vast majority of the times in a large ranked list, it should have a proportional representation in a small sub-sample of that ranked list. A sub-sample will be diversified if proportionality constraints hold for all the query sub-topics. PM-2 has been shown to be more effective compared to xQuAD, but it does not provide any theoretical guarantee about proportional representation of query sub-topics.

Supervised Explicit Diversification Approaches. Supervised diversification approaches [14–16, 24, 27] generally yield better performance compared to the unsupervised ones and we very briefly discuss a few of them. Zhu et al. [27] proposed a new relational learning-to-rank approach to formulate the diversification task. Feng et al. [14] proposed a model based on Markov Decision Process (MDP) – to select a subset of documents from the candidate set – to satisfy as many different subtopics as possible. Montazeralghaem et al. [18] proposed a general reinforcement learning framework for relevance feedback that directly optimizes diversity metrics. Jiang et al. [16] introduced a learning framework for explicit result diversification where subtopics are modeled using an attention mechanism for the next document selection. Hu et al. [15] proposed a new hierarchical structure to represent user intents and using this representation they proposed two general hierarchical diversification models. Following the same line of work, Wang et al. [24] described the concept of hierarchical intents and proposed measures that could evaluate search result diversity with intent hierarchies. They created a new test collection containing intent hierarchies based on the existing TREC Web track 2009-2013 diversity test collections.

2.2 Topic Models for Explicit Diversification

Majority of the supervised approaches do not reach up to their potential when query aspects or topic descriptions are automatically generated, e.g., by a topic model. The automatic topic generation process involves obtaining a ranked list of documents with user query and applying topic models to find query sub-topics from those documents. PM-2 has been particularly effective with automatically discovered query sub-topics as shown in a study by Dang and Croft [13]. The authors proposed to model the query topics using unigrams from the top retrieved documents and apply PM-2 for diversification with those automatically derived topics. This challenging scenario is the focus of this study. However, we do not focus on how these topics are generated, rather we assume that there is

a model that can assign documents with topics and it is possible to derive proportionality constraints from those assignments.

There are different ways of modeling topics for search result diversification: one approach treats topic as a latent variable and models it as a distribution over terms [4]; another approach, a more applicable one in the retrieval landscape, models topics as terms or phrases [13]. The way in which topics are represented affects different methods differently. For example PM-2 is not robust to topics generated by Latent Dirichlet Allocation (LDA) as shown by Dang and Croft [13]. In our experiments we find the same result. In general, we refer to any model that finds query aspects as a topic model.

Term Level Topic Models. Term level topic models generate terms as query aspects. They require automatic identification of topic terms. DSPApprox, a topic term extraction algorithm proposed by Lawrie and Croft [17] for hierarchical multi-document summarization, is generally used for this purpose. The goal of DSPApprox is to select a small set of highly representative terms that best summarizes a set of documents. Dang and Croft [13] used this approach to find a hierarchical topic structure from a ranked list of documents retrieved against a query. The algorithm constructs a vocabulary of terms and phrases from these documents. If a sequence of terms in a document matches a sequence of terms in a Wikipedia title, then that sequence is considered as a phrase and is included in the vocabulary. If an item in the vocabulary and a query term appears within a window of size w, then the vocabulary item is considered as a topic term. Each of these terms is scored based on it's topicality and predictiveness. Topicality measures how informative a topic term is in describing a set of documents, while predictiveness indicates how much the occurrence of a topic term predicts the occurrences of other terms. The algorithm greedily selects a subset of topic terms for which topicality and coverage of the vocabulary is maximized. Once a set of topics, $T = \{t_1, t_2, \ldots, t_n\}$, underlying our query q is found – for each topic t_i, it's relatedness to a document d_j is computed using the following equation from Dang and Croft [13]:

$$P(d_j \mid t_i) = P(t_i \mid d_j) \prod_{q_j \in q} P(q_j \mid d)^{\frac{1}{|t_i|+|q|}} \tag{1}$$

The quantity $P(t_i \mid d_j)$ indicates how prevalent the topic t_i is in a document d_j. Thus a document is represented as a distribution over topics, $M(q, d_j) = [P(t_1|d_j), P(t_2|d_j), \ldots P(t_n|d_j)]$.

3 PROBLEM DEFINITION

Let $T = \{t_1, t_2, ..., t_n\}$ be a set of aspects or topics for a query q, whose topic popularity values are $P = \{p_1, p_2, \ldots p_n\}$, respectively. Query q retrieves $R = (d_1, d_2, \ldots, d_m)$, a ranked list of m documents. A topic model $M : (d_j, T) \rightarrow [0, 1]^n$ models a document as a probability distribution over query topics, $M(d_j, T) = [P(t_1|d_j), P(t_2|d_j), \ldots, P(t_n|d_j)]$, for each $d_j \in R$ retrieved with q. $P(t_i|d_j)$ indicates the probability of observing topic t_i in document d_j. There is a scoring function $F : d_j \in R, q \rightarrow \mathbb{R}$ that provides the score of document d_j retrieved with q. The task of proportionality based diversification is to select and rank a subset S from R, where the percentage of documents in which t_i appears is proportional to p_i for all values of i. Intuitively, it means that if a topic t_i is very likely in R, it should also be very likely in S – *i.e.*, S should be a proportional representation of R.

Generally, a user expects a diverse result set without explicitly providing query topics. She expects the system to discover the underlying query topics given q. Thus, a more realistic and challenging version of the problem is to assume the unavailability of the query topics set, T. In this case, the topic model M discovers T from R to provide input to the diversification algorithm. We study our proposed search result diversification approach under both the simple and complex settings.

As the notion of topic popularity is an important part of the problem definition, we discuss how it is estimated in practice. The set of topic popularity values, P is estimated from R with topic distributions from M. For any topic t_i, M provides $P(t_i|d_j)$, and we compute $p_i = \frac{\sum_{d_j \in R} P(t_i|d_j)}{|R|}$ as *popularity* because it indicates the proportion in which t_i is present in R compared to any other topic $t_{k,k\neq i} \in T$. For an optimal proportionality based diversification algorithm the popularity of topic t_i in $S \subset R$ remains the same as it is for R – i.e., $\frac{\sum_{d_j \in S} P(t_i|d_j)}{|S|} = \frac{\sum_{d_j \in R} P(t_i|d_j)}{|R|}$, for all values of i. This expectation is reasonable: if t_i is largely present or popular in R, then it should also be popular in the sub-sample S. As a result, t_i should be present in at least $\frac{\sum_{d_j \in R} P(t_i|d_j)}{|R|} \times |S|$ documents among $|S|$. Dang and Croft proposed the greedy PM-2 approach to address this constraint, but did not actually incorporate these inequalities as constraints into an optimization framework [12].

4 DIVERSITY WITH PROPORTIONALITY CONSTRAINTS

We model proportionality based search result diversification as a set cover problem with proportionality constraints. First we propose an Integer Linear Program (ILP) formulation of the problem, and then give a relaxation using a suitable Linear Program (LP). To solve the relaxation, we provide a randomized rounding solution to the LP and show that this approach does not violate the proportionality constraints *on expectation*. As a set cover solution does not return a ranked list, we rank the solution using Query Likelihood (QL) and Proportionality Model-2 (PM-2) approach.

4.1 Set Cover Formulation

Our set cover formulation considers documents as sets and topics as elements. We compute a bag-of-topic representation for a document to construct the topic set. The utility of a set/document is defined as the QL score of that document with respect to query q. The set cover formulation also leads us to a different estimation of popularity, P, that we introduced in the previous section.

4.1.1 Documents as Bag-of-topics. Generally, topic models provide a probabilistic association between a document and the query topics. In order to represent a document d_j as a bag-of-topics, we convert the probabilistic association between d_j and t_i, $P(t_i|d_j)$, to a deterministic association and define it as $C(t_i|d_j) \in \{0, 1\}$:

$$C(t_i|d_j) = \begin{cases} 1 & \text{if } P(t_i|d_j) \geq \delta \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

We describe how we compute the value of δ in Equation 2. Topic model $M(d_j, T)$ models a document as a probability distribution

over query topics, $dist(d_j) = [P(t_1|d_j), P(t_2|d_j), \ldots, P(t_n|d_j)]$, for each $d_j \in R$ retrieved with q. We find the most representative topic in d_j, $t_i^* = \arg\max_{t_i} P(t_i|d_j)$. We define a parameter, γ, and compute δ as a function of γ, $\delta = \frac{P(t_{i^*}|d_j)}{\gamma}$. If we increase the value of γ, δ will decrease and document d_j will be associated with more topics. Intuitively, it means that we should assign a topic to a document if it falls into the neighborhood of the most probable topic based on the association with the document.

The deterministic association between documents and topics results in a different number of topics for different documents. Hence the estimation of popularity P changes. In the previous section, we showed how P is determined from R with M. Here, we propose an alternative way to determine P from the deterministic association of documents and topics. We define p_i as the fraction of documents in R that cover t_i. For any topic t_i, M provides $P(t_i|d_j)$, and we compute $p_i = \frac{1}{|R|} \sum_{d_j \in R} C(t_i|d_j)$. The proportionality of a topic t_i is maintained in a sub-sample S of R of size k, if in that sub-sample t_i appears in least $p_i k$ documents and this holds for all $t_i \in T$.

4.2 ILP and Relaxed LP formulation of Set Cover

In this section, we model the set cover problem as an Integer Linear Program (ILP), and introduce the proportionality constraints in the formulation. As defined in section 3, we have a set of aspects or subtopics T and a ranked list of documents R from which we need to sub-sample S. We define a cost function $c : R \rightarrow \mathbb{R}^+$ such that $c(d)$ is the cost of including d, for every $d \in R$. We consider $c(d_i) = |QL(d_i, q)|$, where QL is the negative log likelihood of a query given a document As the scores are negative, taking an absolute value of QL assigns the lowest cost to the highest scoring document. This is reasonable as we find minimum cost set cover that favors documents with minimum cost, which is essentially maximum utility. Our ILP formulation is as follows:

$$
\begin{aligned}
&\text{minimize} \sum_{d \in R} c(d)x(d) \\
&\text{subject to} \sum_{d:C(t_i|d)=1} x(d) \geq 1, \ \forall t_i \in T \\
&\qquad \sum_{d:C(t_i|d)=1} x(d) \geq \alpha_{t_i} \cdot k, \ \forall t_i \in T \\
&\qquad \sum_{d:C(t_i|d)=1} x(d) \leq \beta_{t_i} \cdot k, \ \forall t_i \in T \\
&\qquad x(d) \in \{0, 1\}, \ \forall d \in R
\end{aligned}
$$

In this ILP formulation, we consider a tuple of values $(\alpha_{t_i}, \beta_{t_i})$ for each aspect $t_i \in T$ such that $\alpha_{t_i}, \beta_{t_i} \in [0, 1]$ and $\alpha_{t_i} \leq \beta_{t_i}$. Essentially, $\alpha_{t_i} = p_i - \epsilon$ and $\beta_{t_i} = p_i + \epsilon$ (Please refer to section 3 for a discussion on topic popularity, p_i). The objective is to find a minimum cost collection of documents – i.e., set cover – S such that for each aspect $t_i \in T$, the number of documents in S containing t_i should be at least $\alpha_{t_i} \cdot k$ and at most $\beta_{t_i} \cdot k$. Here, we multiply the size of the set cover $k = |S|$ with the lower and upper bounds to convert proportionality values into document counts. We define an indicator variable $x(d)$ for every document $d \in R$ which is 1 if d is in the set cover and zero otherwise. The objective of the constrained ILP is to minimize the cost by selecting the most relevant documents. For an example, assume we are sampling 20 documents from a ranked list of 100 documents with corresponding QL scores. If the topic "baseball" appears in 10 of the 100 documents its popularity is 10%. Our constraints enforce that in the sample of 20 documents, the topic "baseball" should appear slightly less or more than $20/10 = 2$ times.

$$
\begin{aligned}
&\text{minimize} \sum_{d \in R} c(d)x(d) \\
&\text{subject to} \sum_{d:C(t_i|d)=1} x(d) \geq 1, \ \forall t_i \in T \\
&\qquad \sum_{d:C(t_i|d)=1} x(d) \geq \alpha_{t_i} \cdot k, \ \forall t_i \in T \\
&\qquad \sum_{d} x(d) \leq k \quad \text{[constraint on the size of set cover]} \\
&\qquad x(d) \geq 0 \ ; \ x(d) \leq 1, \ \forall d \in R
\end{aligned} \tag{3}
$$

Relaxation of ILP. As solving the set-cover problem optimally is NP-hard [11], we relax the constraints in the ILP formulation to obtain a Linear Program (LP) by allowing the set variables to take fractional values rather than integer ones. This relaxation is a standard technique for designing approximation algorithms [25]. We further simplify our LP formulation by removing the upper bound constraint containing the variable β_{t_i} on topic coverage. The intention behind this approach is to obtain a feasible solution using an LP solver. It is likely for an LP solver to reach *infeasible* region of the solution space with many constraints. However, removal of the upper-bound constraint might lead us to a set cover of arbitrary size, and hence we enforce another constraint on the size of the set cover for a non-trivial solution. Otherwise we could always satisfy the constraints by selecting more documents. As a proxy for the upper bound constraint on each topic, we use an upper bound constraint over the size of the set cover k. The relaxed LP formulation is shown in Equation 3.

4.3 Randomized Rounding Model

We propose to solve the LP formulated in the previous section using a randomized rounding technique to obtain an approximate solution to the original ILP formulation of the set cover problem. As we can solve any LP in polynomial time [11], let $x^*(d) \in [0, 1]$ be the values of the variables corresponding to the solution of our LP. The main idea of our approach is to use these values as the probability of selecting the document $d_i \in R$ into our sub-sample S. To construct a set cover solution from the fractional solution $x^*(d)$ obtained by solving the linear program, we pick every document $d \in R$ with probability $x^*(d)$ independently into our set cover. This approach is well known to give a good approximation for set cover and is called *Randomized Rounding* [25]. In the following theorem, we argue that the solution obtained using randomized rounding *does not violate* the set-cover constraints on expectation.

Theorem 4.1. *In expectation, randomized rounding yields an optimal solution without violating the LP constraints.*

Proof. Let the set cover obtained using randomized rounding be S. Consider the cost of the set cover $\sum_{d \in S} c(d)$ that can be rewritten as $\sum_{d \in R} c(d)y(d)$, where $y(d) = 1$ if $d \in S$ and zero otherwise. Taking expectation, we have :

$$\mathbb{E}\left[\sum_{d \in R} c(d)y(d)\right] = \sum_{d \in R} c(d)\,\mathbb{E}\left[y(d)\right] = \sum_{d \in R} c(d)x^*(d)$$

The last statement follows because of the fact that $y(d) = 1$ with probability $x^*(d)$ and $y(d) = 0$ with probability $1 - x^*(d)$. Therefore, on expectation the cost of our set cover S is equal to the cost of optimal solution of LP relaxation. For subtopic $t_i \in T$, the number of documents selected with that topic is $\sum_{d:t_i \in d} y(d)$. Taking expectation over this quantity we have:

$$\mathbb{E}\left[\sum_{d:C(t_i|d)=1} y(d)\right] = \sum_{d:C(t_i|d)=1} \mathbb{E}\left[y(d)\right]$$
$$= \sum_{d:C(t_i|d)=1} x^*(d) \geq \alpha_{t_i} \cdot k \text{ (LP constraint)}$$

Similarly, we can show that for all $t_i \in T$ the remaining constraints are also satisfied $\mathbb{E}\left[\sum_{d:t_i \in d} y(d)\right] \geq 1$ and $\mathbb{E}\left[\sum_d y(d)\right] \leq k$ on expectation. □

4.4 Ranking Set Cover Solution

We propose LP-PM-2 which is combination of our proposed linear programming (LP) approach and the PM-2 approach proposed by Dang and Croft [12]. We propose a simple extension over PM-2 using our LP method. We briefly describe PM-2 and describe why and how we extend it.

4.4.1 ***PM-2 as a Document Ranker***. PM-2 is a greedy and iterative approach that promotes diversity by re-ranking an initial ranked list of documents retrieved with any algorithm. To select a document at ranking position $r + 1$, PM-2 scores each topic, $t_i \in T$, based on a heuristic, $\frac{v_i}{2s_i+1}$. Here, v_i is the popularity or the ideal proportionality of topic t_i, while s_i is the proportion of documents within rank r that contains t_i. Intuitively, this heuristic promotes topic proportionality – i.e., if a topic has been covered according to its popularity in the ranked list, it will receive less score from the heuristic. After scoring all the topics $t_i \in T$, t_{i^*}, the highest scored topic based on the heuristic, is selected. Finally, a document that best matches t_{i^*} is selected for rank $r + 1$. The best matching document, d^* with respect to t_{i^*} is selected using the formula below:

$$d^* \leftarrow \underset{d_j \in R}{argmax} \quad \lambda \times qt[i^*] \times P(d_j|t_{i^*}) + (1 - \lambda) \sum_{i \neq i^*} qt[i] \times P(d_j|t_i) \tag{4}$$

In Equation 4, t_i^* indicates the highest scoring topic according to the heuristic mentioned above, qt is a vector containing the scores of all the topics – computed using the heuristic, and i^* indicates the index of t_i*. The λ parameter Equation 4 is one of the components responsible for diversity. A higher value of λ suggests selecting a document that highly represents the best topic at the current rank. Usually a higher value would always decrease diversity as the ultimate goal of diversity is to give the users a flavor of all the topics in a ranked list as quickly as possible. This equation shows that PM-2 actually performs document ranking given a selected subtopic. The equation also suggests that PM-2 is a high precision ranker for all the query subtopics.

4.4.2 ***LP-PM-2 Approach***. We propose LP approach that models proportionality formally and has theoretical guarantee for holding proportionality. Empirically, we found that our LP approach performs better than PM-2 in terms of sub-topic recall, but fails in precision oriented metrics. This appears to be because our retrieved set lacks the effectiveness of the ranking component of PM-2. So, rather than ranking our retrieved set with QL, we use PM-2 as a ranker. This LP-PM-2 approach also selects the best topic at a specific rank using a heuristic. But at the time of picking a document given a topic, we restrict it to score documents only from the set we retrieved using our LP-QL approach.

Our final solution requires the computation LP, which has a constant computational complexity given our application. Theoretically, LP can be solved in polynomial time and having a fixed and very small number of variables for any query does not hamper efficiency. In our experiment, we re-rank only top-50 documents and thus solve an LP with only 50 variables. We did not find a large difference in runtime for PM-2 and LP-PM-2. We provide the output of our empirical analysis in the experimental results section.

5 EXPERIMENTAL SETUP

Query and Collection. We consider the same experimental setup and dataset as Dang and Croft [13]. The dataset contains 147 queries and relevance judgments gathered from three years (2009, 2010, and 2011) of the TREC Web Track's diversity task. We follow a cross-validation approach to tune parameters on queries from any two years and use the other year's queries as evaluation queries. There are 150 queries in total from three years, but we did not consider queries with identifiers 95, 100 and 143. We did this to ensure that the setup is the same as that described by Dang and Croft [13] – to ensure fair comparison with baselines. All code for experimentation is available[1] to allow results to be reproduced.

We use the ClueWeb09 Category B retrieval collection stemmed by the Krovetz stemmer and remove stopwords from queries using a small stopword list. All the diversification approaches are treated as a re-ranking of the Query Likelihood (QL) approach implemented in Indri. Similar to Dang and Croft [12], we include a spam filtering [10] and a stopword ratio component with QL to improve its score. The final score from this approach is calculated according the method of Bendersky et al. [3].

Evaluation Metrics. Our problem definition and methodology depend on the evaluation metrics for diversification. Evaluation of diversification is a well-studied topic and several metrics have been proposed over the years. These metrics are different from ranking and relevance evaluation metrics such as Mean Average Precision (MAP), Normalized Discounted Cumulative Gain (NDCG), etc. However, researchers evaluate diversification models using both relevance and diversification metrics as optimizing for diversification might focus too much on topic coverage and hence it is likely to lose relevant documents in top ranks. Usually, a better diversification approach achieves balanced results in both types

[1] https://github.com/sarwar187/SRD

of metrics. In this section, we discuss two diversification metrics, S-Recall and Precision-IA, that are used to discuss the motivation of our diversification technique.

Precision-IA. is the intent-aware version of precision proposed first by Agrawal et al. [1] along with some other intent-aware metrics. A subtopic is considered a distinct interpretation of the associated query according to the intent-aware measures. Now, given an interpretation or subtopic they proposed to compute standard evaluation measures on that subtopic. Finally, intent-aware measures are computed by taking a weighted average of the results computed from the various interpretations. In the TREC Web track, authors assume equal probabilities on each subtopic [8].

To compute Precision-IA at depth k, it is assumed that there are Q queries or topics and $N_q, 1 \leq q \leq Q$ is the number of subtopics associated with query topic q. Then we define Precision-IA exactly as do Clarke et al. [8]:

$$Precision - IA@k = \frac{1}{Q} \sum_{q=1}^{Q} \frac{1}{N_q} \sum_{i=1}^{N_q} \frac{1}{k} \sum_{j=1}^{k} r_q(i, j) \tag{5}$$

Here, $r_q(i, j) = 1$ if a document at depth j for topic q is judged relevant to subtopic i of topic q, otherwise $r_q(i, j) = 0$. It is possible to compute Precision-IA if the relevance assessors provide subtopic annotation along with relevance judgments. Intuitively, Precision-IA for a model will be higher if it can identify the underlying subtopics of a topic and correctly identify relevant documents for those subtopics.

S-Recall. emerged from subtopic retrieval problem that was first introduced by Zhai et al. [26]. The goal of subtopic retrieval as stated by the authors is to find documents that cover as many different subtopics of a general topic as possible. The output of a subtopic retrieval method is a ranked list that is similar to the output of a document retrieval method. Nonetheless, an optimal ranked list is one that includes documents that cover all the subtopics of a query/topic in the earliest rank. Thus subtopic retrieval evaluation metric, S-Recall, is a function of rank. Borrowing notation from the definition of Precision-IA above we define S-recall as:

$$S - Recall@k = \frac{| \cup_{j=1}^{k} subtopics(d_j) |}{N_q} \tag{6}$$

The $subtopics(d_j)$ function returns the set of subtopics associated with a document at depth j.

5.1 Topic Modeling Techniques and Parameter Settings

The performance of topic proportionality based diversification approaches depend on how query topics are found and assigned to documents. Dang and Croft [13] reported that PM-2 and xQuAD perform the best when topic popularity or proportionality is estimated from the top 50 documents retrieved against a query. They used these 50 documents to find underlying query topics and compute topic proportionality. They evaluated diversification approaches on the top 20 documents after re-ranking these 50 documents. They also showed that there is values in diversifying with topics represented as terms or unigrams. We exactly follow their settings and use their term based topic modeling technique in our experiments; we refer to this model as *unigram*. We also use Latent Dirichlet Allocation (LDA) for topic modeling and apply it's implementation provided by Scikit-Learn [19]. For LDA, we consider the number of topics in the range 2-10 as reported by Carterette and Chandar [7].

Topic modeling techniques provide a distribution over query aspects given a document. But, our LP approach requires a deterministic association between a topic and a document. Our approach to calculate the association between a document and all the query aspects is discussed in Section 4.1. Given a distribution over topics for a document we select a threshold δ, and any topic that has a probability of δ or higher is assigned to that document. For the LDA based approaches we consider δ as 20 equally-spaced values in the range of [0.005, 0.05]. However, for the unigram-based approach, this quantity is estimated differently, because for each query the unigram-based approach generates a different number of aspects, and hence it is difficult to compute a single threshold δ that would work for all the queries. In section 4.1.1, we described how we compute δ using γ. The threshold δ is computed by taking the best highest probable topic of a document and dividing that value by γ. We vary the value of γ from [1, 4] and consider 20 linearly spaced values in this range. When we considered values larger than 4, a document became associated with all the topics on an average. That's why we restricted the upper-bound to 4.

6 RESULT DISCUSSION

We provide experimental justification for combining our LP solution with PM-2 to obtain a better ranking.

Performance on Subtopic Retrieval. In diversification, one of the goals is to produce a ranked list that performs well in the subtopic retrieval task [26]. It is desirable to include documents from many different subtopics at early positions in the ranked list [26]. By ensuring a proportional representation, our LP-QL achieves a high Subtopic Recall (S-Recall) compared to PM-2. The results in Table 1 show significant gain in S-recall for both LDA and Unigram topic modeling techniques. This is intuitive as we have devised a method that guarantees proportionality in expectation and its subtopic recall should be higher. Even though LP-QL has higher S-Recall, it lacks precision in ranking documents for the subtopics – resulting in lower Prec-IA for the unigram topic modeling approach. Our LP approach retrieves a set and the underlying ranking method is Query Likelihood (QL). PM-2 also has a ranking method, that ranks documents by considering the association of a document to a topic $P(d_j|t_i)$. Accordingly, PM-2 wins in Prec-IA because of its capacity to rank documents for subtopics. We used this strength of PM-2 to derive our LP-PM-2 approach.

Overall Discussion on Performance. Table 1 shows the comparison of our LP-QL approach and LP-PM-2 approach compared to QL and PM2 baselines. We show that LP-PM-2 outperforms all the baselines under the unigram topic modeling [13] setting. Specifically, it achieves 3.2% performance gain over PM-2 in terms of α-NDCG. Overall, any diversification method with the unigram topic model gives better performance compared to LDA based topic models across all the metrics.

Table 1: Comparison of topic proportionality based diversification approaches with baselines. Symbol $^\bullet$ indicates that improvements of LP-QL and LP-PM-2 over PM-2 are statistically significant at 90% confidence intervals according to the student's paired t-test. Performance metrics are averaged across TREC 2009, 2010, and 2011 by considering each year's queries as the test queries, while queries of the other two years are used to tune parameters.

Topic Model	Diversification Model	α-NDCG	ERR-IA	Prec-IA	S-Recall	NRBP	NDCG
-	**QL**	0.3929	0.2684	0.1773	0.5513	0.2338	0.2771
LDA [7]	**PM-2 (Baseline)**	0.3555	0.2230	0.1171	0.5170	0.2050	0.2280
	LP-QL (Proposed)	0.3973$^\bullet$	0.2684$^\bullet$	0.1562$^\bullet$	0.5590$^\bullet$	0.2330$^\bullet$	0.2580$^\bullet$
	LP-PM-2 (Proposed)	0.3756	0.2491	0.1430$^\bullet$	0.5494$^\bullet$	0.2129	0.2273
Unigram [13]	**PM-2 (Baseline)**	0.4172	0.2921	**0.1858**	0.5448	0.2632	0.2978
	LP-QL (Proposed)	0.4178	0.2881	0.1705	0.5733$^\bullet$	0.2545	0.2764
	LP-PM-2 (Proposed)	**0.4360**$^\bullet$	**0.3208**$^\bullet$	0.1821	**0.6034**$^\bullet$	**0.2893**	**0.2980**

The cause of the failure of LDA based approaches is the assignment of non-relevant topics to documents. Consistent with the literature [13], our implementation of PM-2 performs worse compared to QL with LDA generated topics. In contrast, our LP-QL approach is robust to topic noise, because it performs better than QL in terms of α-NDCG even with LDA topics. We hypothesize that LP-PM-2 would perform better in comparison with LP-QL in this scenario. But it seems that the ranking component of PM-2 was not an effective addition to LP-QL with LDA. It cannot compensate for the noisy topic modeling. However, LP-PM-2 still performs better compared to PM-2. For the LDA topic model, PM-2 baseline is the worst among all the methods. PM-2 is less robust to noisy topic models.

With the unigram topics, PM-2 performed better compared to baseline QL. An interesting result to note here is LP-QL achieves better S-Recall than PM-2 with unigram topic models. On the other hand, PM-2 achieves better Precision-IA under the same scenario. LP-PM-2 is not the winner considering these two metrics, but it wins over all the methods in terms of the diversification metrics such as α-NDCG, ERR-IA and NRBP. Please note that α-NDCG and ERR-IA are the official evaluation metrics for TREC diversity track. Overall, we show that proportional coverage of topic and effective ranking of documents given a topic are two very important components of a diversification algorithm, and a method that optimizes these two properties achieves better diversification.

Efficiency Analysis. In general, linear programming is less efficient than a greedy algorithm. We compare the performance of PM-2 with LP-PM-2 in terms of execution time. We implemented both approaches in python and conduct timing experiments on a 12x2.66 GHz machine with 16GB RAM running Ubuntu Operating System. Figure 1 shows how much we lose in terms of efficiency by considering LP-PM-2 as an alternative to PM-2. We observe that with the increase in the number of query topics or aspects, the time taken by LP-PM-2 increases. For queries with 30-36 topics, it takes about 400 milliseconds to find a solution. It is unlikely for a query to have such a large number subtopics, but it eventually depends on the topic model being used.

Performance Analysis on Individual TREC year. Table 1 presents the performance of LP-PM-2 which is averaged over TREC 2009, 2010, 2011 and compares it with strong unsupervised baselines across a wide variety of evaluation metrics. But, it is also interesting to observe the performance of the methods across different years of

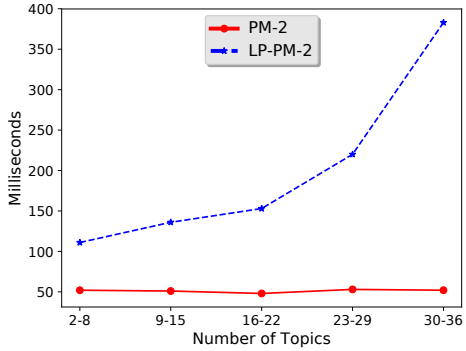

Figure 1: Efficiency comparison with increasing #topics

the TREC diversity track. To investigate the fold-wise performance, we report the α-NDCG@20 and ERR-IA@20, as they are the official evaluation metrics of the TREC diversity track.

Table 2 and Table 3 show the $\alpha - NDCG$ and ERR-IA values, respectively, obtained from applying various approaches using TREC provided subtopics. These subtopics are human provided and diversifying with respect to them generally results in much better performance compared to computational topic models. The results show that LP-PM-2 is not a straightforward winner here as PM-2 performs significantly well in TREC 2010. Table 4 and Table 5 show the performance metrics obtained from various approaches with unigram query topics [13]. As previously discussed, it is much harder to attain comparable performance to human provided topics. For TREC 2011 fold, LP-PM-2 achieves similar performance to PM-2 without any human topic annotation, which is promising.

7 CONCLUSION

In this study, we view the problem of search result diversification under the light of topic proportionality. We show that it is important for a diversification algorithm to satisfy proportionality constraints, at least in expectation. We propose such an algorithm, LP-QL, and show that it performs better in terms of S-Recall. LP-QL extracts a set cover solution and we can further boost its performance using PM-2 ranking component. This extension of PM-2 outperformed PM-2 as well as LP-QL. We also discussed that LP with a small

Table 2: α-NDCG@20 values for TREC subtopics

	TREC 2009	TREC 2010	TREC 2011
QL	0.2979	0.3236	0.5566
xQuAD	0.3300	0.4074	0.5724
PM-1	0.3076	0.4323	0.5774
PM-2	0.3473	**0.4546**	0.5886
LP-QL	0.3009	0.3330	0.5489
LP-PM-2	**0.3634**	0.4022	**0.5991**

Table 3: ERR-IA@20 values for TREC subtopics

	TREC 2009	TREC 2010	TREC 2011
QL	0.1953	0.2081	0.4387
xQuAD	0.2207	0.2671	0.4551
PM-1	0.2027	0.3071	0.4478
PM-2	0.2407	**0.3271**	0.4642
LP-QL	0.3190	0.2571	0.4620
LP-PM-2	**0.3292**	0.2621	**0.4863**

Table 4: α-NDCG@20 values for unigram subtopics

	TREC 2009	TREC 2010	TREC 2011
QL	0.2979	0.3236	0.5566
PM-2	0.3145	0.3899	0.5473
LP-QL	0.313	0.3632	0.5613
LP-PM-2	**0.3302**	**0.4029**	**0.5749**

Table 5: ERR-IA@20 values for unigram subtopics

	TREC 2009	TREC 2010	TREC 2011
QL	0.1953	0.2081	0.4387
PM-2	0.2173	0.2717	0.433
LP-QL	0.2161	0.2777	0.4412
LP-PM-2	**0.2192**	**0.287**	**0.4562**

number of variables can be solved in constant time and there is no significant difference between the run time of PM-2 and LP-PM-2. We validated the performance of LP-PM-2 across different years of TREC diversity track and showed that our approach is effective with both TREC provided query subtopics as well as automatically derived subtopics. In the future, we want to study the generalization of LP-PM-2 to other tasks such as diversification for non-factoid QA. We also want to explore neural approaches for finding topics underlying a query automatically.

ACKNOWLEDGMENTS

We thank Barna Saha for very useful discussions on algorithm design and analysis. This work was supported in part by the Center for Intelligent Information Retrieval. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

REFERENCES

[1] Rakesh Agrawal, Sreenivas Gollapudi, Alan Halverson, and Samuel Ieong. 2009. Diversifying Search Results. In *Proceedings WSDM (WSDM '09)*. ACM, New York, NY, USA, 5–14. https://doi.org/10.1145/1498759.1498766
[2] Aris Anagnostopoulos, Andrei Z. Broder, and David Carmel. 2005. Sampling Search-engine Results. In *Proceedings WWW (WWW '05)*. ACM, New York, NY, USA, 245–256. https://doi.org/10.1145/1060745.1060784
[3] Michael Bendersky, David Fisher, and W. Bruce Croft. 2010. TREC 2010 Web Track Notebook: Term Dependence, Spam Filtering and Quality Bias. In *TREC '10, Gaithersburg, Maryland, USA, November 16-19, 2010.*
[4] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research* 3, Jan (2003), 993–1022.
[5] Jaime Carbonell and Jade Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *In SIGIR*. 335–336.
[6] Jaime Carbonell and Jade Goldstein. 1998. The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries. In *Proceedings SIGIR*. 335–336.
[7] Ben Carterette and Praveen Chandar. 2009. Probabilistic models of ranking novel documents for faceted topic retrieval. In *Proceedings of the 18th ACM conference on Information and knowledge management*. ACM, 1287–1296.
[8] Charles L Clarke, Nick Craswell, and Ian Soboroff. 2009. *Overview of the trec 2009 web track*. Technical Report. WATERLOO UNIV (ONTARIO).
[9] Charles L.A. Clarke, Maheedhar Kolla, Gordon V. Cormack, Olga Vechtomova, Azin Ashkan, Stefan Büttcher, and Ian MacKinnon. 2008. Novelty and Diversity in Information Retrieval Evaluation. In *Proceedings SIGIR (SIGIR '08)*. ACM, New York, NY, USA, 659–666. https://doi.org/10.1145/1390334.1390446
[10] Gordon V Cormack, Mark D Smucker, and Charles LA Clarke. 2011. Efficient and effective spam filtering and re-ranking for large web datasets. *Information retrieval* 14, 5 (2011), 441–465.
[11] Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. 2009. *Introduction to algorithms*. MIT press.
[12] Van Dang and W. Bruce Croft. 2012. Diversity by proportionality: an election-based approach to search result diversification. In *Proceedings SIGIR*.
[13] Van Dang and W. Bruce Croft. 2013. Term level search result diversification. In *Proceedings SIGIR*. 603–612.
[14] Yue Feng, Jun Xu, Yanyan Lan, Jiafeng Guo, Wei Zeng, and Xueqi Cheng. 2018. From greedy selection to exploratory decision-making: Diverse ranking with policy-value networks. In *Proceedings SIGIR*. 125–134.
[15] Sha Hu, Zhicheng Dou, Xiaojie Wang, Tetsuya Sakai, and Ji-Rong Wen. 2015. Search result diversification based on hierarchical intents. In *Proceedings CIKM*. 63–72.
[16] Zhengbao Jiang, Zhicheng Dou, Wayne Xin Zhao, Jian-Yun Nie, Ming Yue, and Ji-Rong Wen. 2018. Supervised search result diversification via subtopic attention. *IEEE Trans. on Knowledge and Data Engineering* 30, 10 (2018), 1971–1984.
[17] Dawn J. Lawrie and W. Bruce Croft. 2003. Generating Hierarchical Summaries for Web Searches. In *SIGIR '03*.
[18] Ali Montazeralghaem, Hamed Zamani, and James Allan. 2020. A Reinforcement Learning Framework for Relevance Feedback. In *Proceedings SIGIR*. 59–68.
[19] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.
[20] Filip Radlinski and Susan Dumais. 2006. Improving Personalized Web Search Using Result Diversification. In *Proceedings SIGIR (SIGIR '06)*. ACM, New York, NY, USA, 691–692. http://doi.acm.org/10.1145/1148170.1148320
[21] Rodrygo L. T. Santos, Jie Peng, Craig Macdonald, and Iadh Ounis. 2010. Explicit Search Result Diversification Through Sub-queries. In *ECIR'2010*.
[22] Chun-Hua Tsai and Peter Brusilovsky. 2018. Beyond the Ranked List: User-Driven Exploration and Diversification of Social Recommendation. In *23rd International Conference on Intelligent User Interfaces*. ACM.
[23] Lakshmi Vikraman, W. Bruce Croft, and Brendan O'Connor. 2018. Exploring Diversification In Non-factoid Question Answering. In *Proceedings of the 2018 ACM SIGIR International Conference on Theory of Information Retrieval*.
[24] Xiaojie Wang, Zhicheng Dou, Tetsuya Sakai, and Ji-Rong Wen. 2016. Evaluating search result diversity using intent hierarchies. In *Proceedings SIGIR*. 415–424.
[25] David P Williamson and David B Shmoys. 2011. *The design of approximation algorithms*. Cambridge university press.
[26] Cheng Xiang Zhai, William W. Cohen, and John Lafferty. 2003. Beyond Independent Relevance: Methods and Evaluation Metrics for Subtopic Retrieval. In *SIGIR '03*. ACM, New York, NY, USA.
[27] Yadong Zhu, Yanyan Lan, Jiafeng Guo, Xueqi Cheng, and Shuzi Niu. 2014. Learning for search result diversification. In *Proceedings SIGIR*. 293–302.
[28] Cai-Nicolas Ziegler, Sean M. McNee, Joseph A. Konstan, and Georg Lausen. 2005. Improving Recommendation Lists Through Topic Diversification. In *Proceedings WWW (WWW '05)*. ACM.