

Improving Result Diversity using Probabilistic Latent Semantic Indexing*

May 28th, 2010

Peter Lubell-Doughtie[†]
Universiteit van Amsterdam
lubell@science.uva.nl

ABSTRACT

The *diversity* of a set of search results reflects that set's coverage of multiple interpretations of the query it is based upon. Interest in search result diversity has increased as search engines struggle to return relevant information for ambiguous queries from an increasingly large data set. We develop a result diversification system which uses probabilistic latent semantic indexing to form clusters that are then used to reorder search results and increase result list diversity scores. We show that adjusting the influence of rank in the reordering algorithms improves their performance by finding a balance between the importance of rank and the importance of generated clusters. Finally, by applying the result diversification system to the known clusters used in judging, we generate reordered lists that estimate upper-bounds of the diversity scores.

1. INTRODUCTION

The standard in ranking documents is to order them by probability of relevance to the information need, giving documents that are more relevant a higher rank. This method is referred to as the *probability ranking principle* (PRP) [13] and can be used to justify common information retrieval scores such as Precision, MAP, and nDCG. Given a query q , document d , and binary relevance function R , the PRP states that documents should be ordered by decreasing value of $P(R = 1|d, q)$. This formula assumes q is the actual information need, which is tenuous considering recent work showing that a significant number of queries are ambiguous [14] and may denote multiple differing information needs.¹ A further problem is that, under the PRP, two documents with the same content have the same probability of relevance and, supposing they have the highest relevance, will

*Written for Advanced Information Retrieval, Professor Marteen de Rijke, 2nd Semester 2009-2010

[†]UvA Net ID: 6095445

¹The inherent ambiguity of word senses demonstrates this difficulty. A common example of ambiguity is the query "jaguar", which may refer to the brand of car, the feline, the operating system, or perhaps something more obscure. Intertwined with sense ambiguity is dynamic sense reference, wherein the information needs expressed by a query, and their associated probabilities, change over time. The query "president of the united states" is a typical example of a query with a dynamic referent.

be ranked in the top 2 positions. In this case, the underlying problem is the PRP's assumption that the relevance of one document is independent from the other documents returned. In practice this assumption fails, a duplicate document adds little (if any) value.

In search, and especially web search, there is often a one to many mapping from the text of a user's query to the information needs that can be represented by this query. Therefore optimizing search results to satisfy only one of these many information needs will leave some users' needs unfulfilled and be biased towards redundancy within the chosen information need. To address this we can increase search result *novelty* (lack of redundancy) and *diversity* (coverage of multiple topics) by optimizing search results' coverage of the information needs a query could represent.

In addition to suggesting the content of our search result list, a set of topics for both a query and the documents it returns provides a method of evaluating our retrieval algorithm. The α -nDCG@ k metric introduced in [5] modifies nDCG@ k by rewarding result lists for novelty amongst their returned documents. The *intent aware precision*, or P-IA@ k , metric introduced in [1] modifies Precision@ k by summing the precision for each topic weighted by the probability that topic is intended by the query.

These topic based metrics and a general topic based solution to diversity raise two fundamental questions:

1. What topics are represented by a query and what is the probability distribution over the query's topics?
2. What topics are represented by each document and what is the probability distribution over each document's topics?

The result diversification system we present uses probabilistic latent semantic indexing to generate a topic model for the top n documents returned by a query, giving us a probability distribution of topics over documents. We then use these document topics to create a topic distribution for a query based on the documents returned for that query using standard retrieval methods. There are numerous alternative methods that can be used to create these topics and distributions, in Section 3 we examine some of these alternatives.

Using topic distributions, in combination with the result reordering algorithm IA-SELECT [1], we are able to improve on the baseline scores for α -nDCG@{5, 10, 20} and P-IA@{5, 20}. We analyze the effects of modifying the distribution of probability mass in the IA-SELECT algorithm and find that better results are achieved when a document's

rank significantly affects its probability of relevance. Furthermore, by generating topics and probabilities for the IA-SELECT algorithm based on the known topics (as used by the diversity evaluation metrics) we create a reordered list and metric scores that approximate the upper-bound on the performance of IA-SELECT.

We proceed in Section 2 by providing an overview of diversity metrics and reviewing a system that was built to optimize for these metrics. Next we present our experimental approach in Section 3 and our evaluation and results in Section 4. In Section 5 we discuss our results and we conclude in Section 6.

2. RELATED WORK

Diversity research in information retrieval has progressed in parallel with the two tasks of designing metrics that accurately measure the diversity of a list of search results and designing algorithms that optimize the order of search result lists to score highly on these metrics and satisfy the needs of users. The earliest diversity metric and algorithm that was formally explored is maximum marginal relevance (MMR). In the MMR strategy for ranking documents, we define marginal relevance as a document’s relevance to a query given its similarity to already returned documents, and then maximize a linear interpolation of the similarity between the document and the query minus the similarity between the document and the previously returned documents [3]. Given R is a ranked list of documents, $S \subset R$ is a set of previously selected documents, and $d_j^* = \max_{d_j \in S} Sim_2(d_i, d_j)$, we define:

$$MMR \equiv \arg \max_{d_i \in R \setminus S} [\lambda Sim_1(d_i, q) - (1 - \lambda)d_j^*] \quad (1)$$

where λ is an interpolation parameter, q is the user query, and the Sim_i functions are similarity metrics. Increasing λ gives preference to narrower results, while decreasing λ gives preference to broader results. In practice an effective search strategy has been to start with a broad result set and narrow over time by increasing λ . In MMR, diversity and novelty are produced through the choice of similarity functions.

2.1 MMR and subtopic approaches

One may view the differences between documents as an indication or approximation of their topical content, and a measure that takes these differences into account as dividing documents into different topics. Zhai, Cohen, and Lafferty do exactly this in applying the MMR strategy to the problem of subtopic retrieval [19]. The authors begin by defining scoring metrics that take subtopics into account by rewarding the inclusion of documents from many different subtopics early in the ranking while discouraging the inclusion of many documents from the same subtopic. The first metric, S -recall at rank k , equals the sum of the number of distinct subtopics found in the first k returned documents divided by the total number of subtopics. The second metric, S -precision at recall r , equals the minimum rank at which the optimally ranked set has S -recall r divided by the minimum rank at which the ranked set being evaluated has S -recall r .

The authors then derive a general scoring function by summing over a weighting assigned to all the probabilities for combinations of relevant or not relevant and new or not new attributes given a specific document. After some simplifying

assumptions, they define a scoring function:

$$s(d_i; d_1, \dots, d_{i-1}) \equiv p(q|d_i)(1 - \rho - value_N(\theta_i; \theta_1, \dots, \theta_{i-1})) \quad (2)$$

where $value_N(\theta_i; \theta_1, \dots, \theta_{i-1})$ is a newness or novelty value function – with θ_i the language model of document d_i , ρ is a ratio indicating the cost of seeing a non-relevant document compared to that of seeing a relevant but redundant document, and $p(q|d_i)$ is the query likelihood. In experiments the authors found that gains obtained by increasing the rank of novel documents were offset by the cost of ranking a non-relevant document higher. This is a common difficulty and one which we also experience in our experiments.

In work which places novelty measurement and subtopic retrieval in their broader context, Zhai and Lafferty explore characterizing information retrieval in terms of risk minimization [20]. They provide a risk minimization framework in which the objective is to choose a set of documents and a presentation strategy so as to minimize the integral of a loss function defined over the system parameters, user factors, and document source factors, while accounting for a posterior distribution on the system parameters. Both the original MMR from [3] and the modified MMR from [19] can be cast in this risk minimization framework. This characterization makes clear the potential shortcomings of assuming independence between document novelty and relevance. Under the independence assumption there is no “direct measure of relevance of the new information contained in a new document” [20]. An additional point of criticism is that subtopic coverage is not directly measured. This applies saliently to a diversity and novelty task where we would expect direct measurement of these factors, and an algorithm allowing direct optimization of this measurement, to yield improved results.

2.2 Category based approaches

Category based approaches address some of the above shortcomings of the MMR approach. Category based metrics are defined to explicitly measure subtopic (also known as category, nugget, and facet) retrieval. In contrast to the MMR approach based on document similarity, the work of Clarke et al. attempts to increase novelty and diversity by assigning *information nuggets* to the user’s query and the documents returned for this query. Their method defines the probability a document is relevant as the probability there exists a nugget in the intersection of the nuggets in the user’s query and those in the document [5]. Based on this, a modified gain vector is defined by equating the relevance of a document returned at position k to the sum over the relevant nuggets it contains, discounted relative to the number of documents up to position $k - 1$ that are also judged to contain these nuggets. The authors define a modified version of nDCG, called α -nDCG, which rewards novelty by using the modified gain vector. In exploratory experiments Clark et al. use nuggets and relevance judgements taken from the TREC 2005 and 2006 question answering tracks to show that using pseudo relevance feedback on a result list decreases α -nDCG score. Because pseudo relevance feedback focuses results on the content of the highest ranked results, and through this reduces list diversity, these experiments show that α -nDCG is functioning as a diversity metric.

One immediate shortcoming of α -nDCG is that all information nuggets have the same relevance to a document and a query. In practice there is no reason to expect a uni-

form distribution of nugget probabilities for a query. Certain nuggets are likely to be more popular and therefore more probable (e.g. the query “Michael Jordan” likely refers to the former basketball player and not the machine learning researcher). Similarly, it is highly probable that within documents the representation of nuggets will be unequally distributed. Agrawal et al. address this by introducing categories, which are synonymous to nuggets, and defining the problem of *result diversification*. Given query q , a set of documents D , a probability distribution of categories over the query $P(c|q)$, the probability $V(d|q, c)$ that a document d satisfies a user issuing a query q with intended category c , and an integer k , the goal of result diversification is to find $S \subseteq D$ with $|S| = k$ which maximizes

$$P(S|q) = \sum_c P(c|q) \left(1 - \prod_{d \in S} (1 - V(d|q, c))\right) \quad (3)$$

The product in Equation 3 makes the Naive Bayes assumption in calculating the probability that all documents fail to satisfy a given category. Summing over all categories, relative to their probabilities given the query, maximizes the likelihood that the user finds results satisfying their query – with respect to its category distribution – in the top k results.

Maximizing Equation 3 is NP-hard, however a greedy algorithm can be defined which is a $(1 - 1/e)$ -approximation [1]. Pursuing this, Agrawal et al. develop the IA-SELECT algorithm, which at each step chooses the document with the highest probability of satisfying the user assuming all documents so far fail to do so. They further define a general method for deriving *intent aware* versions of standard information retrieval metrics by taking the sum over all categories for a query of the probability of the category given the query multiplied by the value of the metric given the category. Using this formulation they present (precision) P-IA, (recall) R-IA, NDCG-IA, MAP-IA, and MRR-IA as metrics evaluated relative to each intent the user’s query could represent.

Agrawal et al. conducted retrieval experiments by first using the public ODP taxonomy² to classify intents (i.e. create categories) and then incorporating relevance scores given by both a propriety repository and users of Amazon Mechanical Turk.³ In these experiments they found that IA-SELECT outperforms three leading search engines on all the intent aware metrics. Although an important empirical test, if the goal is to apply diversity reordering to web search at large this approach is unsustainable because the relevance measurement methods used cannot scale to the size of the web.

Both the category approach of Agrawal et al. and the nugget approach of Clark et al. maximize diversity and penalize redundancy by defining measures for which a greedy search is appropriate. Inherent in this strategy is the possibility that a relevant but rare nugget will be excluded because it co-occurs only in documents containing other nuggets that have already been returned. To address this, we can maximize diversity without regard for redundancy by assuming that a document is relevant if it contains any relevant nuggets. Proceeding in this manner, Carterette and Chandar define *faceted topic retrieval* in which there is one “correct” interpretation of a query and the goal is to return

a set of documents which covers all the *facets* of this interpretation [4]. Because the system is specifically designed to maximize the number of facets retrieved, which are analogous to subtopics, it is evaluated with S-precision and S-recall [19].

Here we reformulate an equivalent version of the likelihood function presented in [4] so that it can be more easily compared with Equation 3. Given a query q , a set F of facets with $|F| = m$, and a set D of documents, the goal is to find $S \subseteq D$ with $|S| = k$ which maximizes the likelihood function:

$$L(q|F, D) = \prod_{j=1}^m \left(1 - \prod_{d \in S} (1 - P(F_j \in d))\right) \quad (4)$$

Note that the inner product term in Equation 4 is equivalent to the product term in Equation 3 if V is defined without q in the likelihood. This formulation is reasonable because we are attempting to cover all facets regardless of query likelihood. Equation 4 makes two independence assumptions: that facets occur in documents independently and that facets occur in documents independent of one another. Maximizing Equation 4 is NP-hard and the authors propose solving it by (i) using a greedy algorithm which takes the marginal likelihood conditional on previously selected documents (comparable to the method of IA-SELECT), (ii) relaxing y to a vector of real numbers and solving with conjugate gradient descent, and (iii) for each facet taking the document with maximum $P(F_j \in d)$. In evaluation the last method was found to produce significantly higher S-recall and lower redundancy scores when compared with the other methods.

As mentioned by Carterette and Chandar, the criteria in Equation 4 provides no information for ranking the documents. To account for this one could use Equation 4 for document selection and Equation 3 for document ranking. This model would first retrieve a document set that covers the query’s facets and then rank it according to the probability each facet is relevant to the query.

2.3 Risk based approaches and axiomatization

A unique approach to diversity has been taken by Wang and Zhu in which they draw on its commonalities with financial portfolio selection, equating the set of returned documents with a portfolio of financial instruments and then seeking to balance the mean value of this portfolio (the overall relevance of the documents) against its risk (the variance in relevance) [17]. They define the relevance of a document set of size n as:

$$R_n \equiv \sum_{i=1}^n w_i r_i \quad (5)$$

where r_i is the estimated relevance score of document i and w_i , such that $\sum_i w_i = 1$, is the importance of the relevance score at rank i . Given the expected relevance $E[r_i]$ and a matrix C_n such that $c_{i,j}$ is the covariance of the relevance scores for documents i and j we define:

$$E[R_n] = \sum_{i=1}^n w_i E[r_i] \quad (6)$$

$$Var(R_n) = \sum_{i=1}^n \sum_{j=1}^n w_i w_j c_{i,j} \quad (7)$$

²Open Directory Project (ODP): www.dmoz.org

³Amazon Mechanical Turk: www.mturk.com

$E[R_n]$ is the collection mean and $Var(R_n)$ is the collection variance. From Equation 7 we see that decreasing the covariance – the correlation between documents – will decrease the variance, or risk. Diversifying the documents within the collection will accomplish this and therefore reduce the uncertainty of the expected overall relevance.

The authors go on to define a ranking function that parameterizes and combines these measures so that the amount of risk for a ranked document list can be adjusted. In evaluations conducted on the TREC ad hoc retrieval and subtopic retrieval tasks the authors found that by increasing the importance of variance, and thereby increasing diversity, higher mean reciprocal rank (MRR) scores were returned. In contrast, decreasing the importance of the variance increased MAP scores, which is expected since diversity generally negatively affects MAP score. Neither in this work nor in other work we are aware of has this approach been evaluated against the previously mentioned metrics of MAP-IA or α -nDCG. We would expect increasing the importance of variance to be positively correlated with increased scores on these metrics.

Based on previous work in axiomatization of ranking and clustering systems, Gollapudi and Sharma list 8 axioms “each of which seems intuitive for the setting of diversification” [8]. They go on to prove that a diversification function cannot simultaneously satisfy all of the axioms. Given this, determining what axioms a system does satisfy provides a method for quantify choices made in the design of result diversification systems. The authors provide an example diversification system that maximizes the weighted sum over the relevance of returned documents and the dissimilarity amongst returned documents. The system they use is quite similar to MMR, as formulated in [3], although the specific weighting is different. They show that this system fails to satisfy the *stability* axiom, which requires that the output set does not change arbitrarily with output size. Formally, given an optimal set S_k where k denotes the size of the set, the stability axiom requires $S_k \subset S_{k+1}$. The authors note that the result diversification problem defined in [1], and upon which the IA-SELECT algorithm is based, violates the stability axiom and the *independence of irrelevant attributes* axiom. This axiom requires that the score of a set is not affected by most of the attributes of documents outside of the set. Given there is a consensus that the axioms provided by Gollapudi and Sharma are accurate, knowing what axioms a system does and does not satisfy can prove useful in determining which diversification system is best suited to a specific task. The diversification system used in this paper is based upon IA-SELECT and will therefore violate the stability and independence of irrelevant attributes axioms.

2.4 Optimizing for diversity

Experiments applying IA-SELECT, which are conducted in [1], show that diversity scores can be improved through reordering. In [6] Dou et al. present a reordering algorithm that is similar to [1] but more general with regard to inputs and formulation. Given $r(q, d)$ is the original relevance of a document d for a query q , their algorithm selects the next best document as

$$d_{|S|+1} = \arg \max_{d \in R \setminus S} [\alpha r(q, d) + \mathfrak{R}_{C \in \mathcal{C}} v(d, S, C)] \quad (8)$$

where α controls the importance of the the original relevance

versus the diversity rank, similarly to the MMR algorithm in Equation 1. \mathfrak{R} is an operator that combines multiple dimensions of subtopics. The dimensions used in [6] are anchor texts, search result clusters as calculate by the *learning to cluster* method presented in [18], and sites of search results. $v(d, S, C)$ is the importance of document d to category C given the documents in S are already selected. This is calculated as:

$$v(d, S, C) = \sum_{c \in C} w_c \cdot \phi(c, S) \cdot r(q_c, d) \quad (9)$$

where w_c is the weight of subtopic c , $r(q_c, d)$ is the importance of document d for subtopic c with respect to query q , represented as q_c , and $\phi(c, S)$ is the importance of subtopic c given the current selection of documents S . The authors calculate $\phi(c, S)$ as:

$$\phi(c, S) = \begin{cases} 1 & \text{if } S = \{\} \\ \prod_{d_s \in S} [1 - r(q_c, d_s)] & \text{otherwise} \end{cases} \quad (10)$$

which is analogous to the calculation in the inner product of Equation 3 used by the IA-SELECT algorithm.

3. RESULT DIVERSIFICATION SYSTEM

The diversification system we present uses the IA-SELECT algorithm in combination with subtopics mined from documents using probabilistic latent semantic indexing. The algorithm allows us to vary the influence of document rank in reordering by adjust its use in subtopic to query assignment and document relevance probability.

3.1 Diversification algorithm

The IA-SELECT algorithm, as presented in [1], depends on $P(c|q)$, a probability distribution of subtopics over a query, and $V(d|q, c)$, an estimate of the probability that document d can satisfy subtopic c for query q . The methods used for selecting both of these values significantly influence the performance of the system. Given a set of documents R returned for a query q we calculate:

$$P(c|q) = \sum_{d \in R} r(c, d)^{\phi_p(\text{rank}(q, d))} \quad (11)$$

where $r(c, d)$ is the probability d is a member of the subtopic c and $\text{rank}(q, d)$ is the rank of document d for query q . The function $\phi_p(x)$ was evaluated as constant, identity, and $1 + \log(x)$. We also experimented with a uniform distribution that assigns equal probability mass to each subtopic for a query.

The document probability function was similarly defined as:

$$V(d|q, c) = r(c, d)^{\phi_v(\text{rank}(q, d))} \quad (12)$$

where all functions used are defined equivalently as in Equation 11 except that $\phi_v(x)$ is additionally evaluated as x^2 and x^3 . In Equation 11 ϕ_p determines the importance the rank of a document plays in calculating the probability that document’s subtopics are relevant to a query. In Equation 12 ϕ_v determines the importance a document’s rank plays in calculating its own relevance. When ϕ_i is equal to a constant, document rank is irrelevant, otherwise the greater the convexity of ϕ_i the greater the influence of rank.

Given a set S of documents that have already been re-ordered, the IA-SELECT algorithm first initializes $U(c|q, S)$,

the conditional distribution over subtopics given S , using the query subtopics $P(c|q)$. To return a reordered list k documents long the algorithm adds the document with the highest marginal utility to S . If ties exist the first document found to have the highest marginal utility is chosen. The marginal utility of a document, $g(d|q, c, S)$, can be interpreted as the probability that this document satisfies the user given all previously selected documents fail to do so and is calculated as:

$$g(d|q, c, S) \leftarrow \sum_{c \in C(d)} U(c|q, S)V(d|q, c) \quad (13)$$

After this document is chosen and added to S , the conditional probabilities are updated using Bayes rule to account for the new document being a member of S :

$$U(c|q, S) = (1 - V(d'|q, c))U(c|q, S \setminus \{d'\}) \quad (14)$$

where d' is the newly added document. This process is repeated until $|S| = k$.

3.2 Mining subtopics

To create subtopics we use the clustering algorithm probabilistic latent semantic indexing based on the implementation presented in [15] and used in the Lemur Toolkit [11]. We investigate generating topic membership probabilities using two different partition methods. In the first method documents were taken as a single collection and topic probabilities were generated for all documents at once. This method is much less resource intensive, however it goes against the spirit of category based diversity, in which categories are relevant to a query and optimized for documents relative to this query.

In the second method, upon which later experiments are based, cluster probabilities were generated per query for the documents returned. For each query the procedure is as follows:

1. run TFIDF retrieval for the query
2. select the top k most relevant documents (which will later be reordered)
3. generate cluster probabilities for these k documents

This procedure creates a separate set of clusters and distributions over these clusters for each query evaluated.

4. EVALUATION

We evaluate our result diversification system using the WebCLEF 2007 question answering corpus, which is presented in Jijkoun and de Rijke [9]. To convert this corpus into a retrieval task with subtopics we parse the assessments file, letting the topic of each question form the query and the nuggets for answers to the question form the subtopics of the query. Figure 1 presents an excerpt of the XML file used. The content of the `topic_title` tag forms our query and for each query (`topic`) we have a set of subtopics (`nuggets`), each described by a set of excerpts (`spans`) that contain text relevant for the specific subtopic (`nugget`). We perform a text search of the WebCLEF document corpus for each `span` within a subtopic to generate a list of documents relevant to that subtopic.

After the above processing we have produced a list of tuples containing queries, subtopics, and documents that form

the `qrels` file which is used in scoring our retrieval results. A total of 30 topics are used to form queries, however the number of subtopics is not equal for all queries. After removing a query for which there are no subtopics (topic 12), the number of subtopics per query ranges from 2 to 40 with a mean of 12. The final diversity scores are calculated by taking the arithmetic mean of the diversity scores for each query and we therefore expect differences produced by different numbers of subtopics per query to be insignificant.

4.1 Baseline experiment

To create a baseline ordering of documents and diversity scores we retrieve the 200 top ranked documents according to TFIDF using BM25 [10]. Later, when using the IA-SELECT algorithm to reorder results, this is the result list which will be reordered. Even given an oracle that can solve the NP-hard result diversification problem, it will still be restricted to reordering only documents in this list.

4.2 PLSI cluster reordering experiments

In the first experiment subtopics are induced for a query by running PLSI clustering on the top 20 returned results. The number 20 was chosen because the top 20 results are the largest number of results that will be considered by the scoring algorithms. To evaluate how increasing the number of documents to be reordered effects diversity scores we perform a second experiment in which PLSI based subtopics are formed for the top 200 returned results and then reordered based on their subtopics. In both experiments the PLSI algorithm clustered documents into 20 subtopics. In Figure 2 we present results for α -nDCG@{5, 10, 20} comparing a baseline with results reordered using IA-SELECT and clusters induced from 20 and 200 documents. The reordering algorithm uses $\phi_p(x) = 1 + \log(x)$ and $\phi_v(x) = x^2$ for both the 20 and 200 document runs. In Table 1 we present the full results for α -nDCG and P-IA metrics.

Reordering based on 20 documents, experiment IA-SELECT 20, gives the best results, outperforming on all diversity scores except P-IA@10. In contrast, reordering based on 200

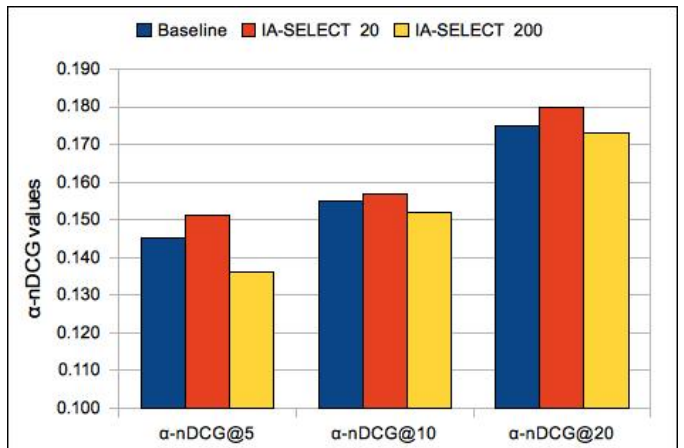


Figure 2: α -nDCG@{5,10,20} scores without reordering (Baseline), reordering using PLSI clustering based on 20 documents (IA-SELECT 20), and reordering based on 200 documents (IA-SELECT 200).

```

...
<topic id="webclef2007_topic_2">
  <topic_title>Symptoms Avian Influenza or bird flu</topic_title>
  <known_spans/>
  <nuggets>
    <nugget id="nugget_109">
      <nugget_name>not symptoms</nugget_name>
      <span id="span_376">H5N1 HPAI can be spread from birds to people as a result</span>
      ...
    </nugget>
    ...
  </topic>
  ...

```

Figure 1: Excerpt from the WebCLEF 2007 Assessments XML file.

Experiment	α -nDCG@5	α -nDCG@10	α -nDCG@20	P-IA@5	P-IA@10	P-IA@20
Baseline	0.145	0.155	0.175	0.051	0.046	0.031
IA-SELECT 20	0.151	0.157	0.180	0.055	0.044	0.049
IA-SELECT 200	0.136	0.152	0.173	0.049	0.040	0.032

Table 1: Diversity scores for all result diversification systems. IA-SELECT algorithms use $\phi_p(x) = 1 + \log(x)$ and $\phi_v(x) = x^2$.

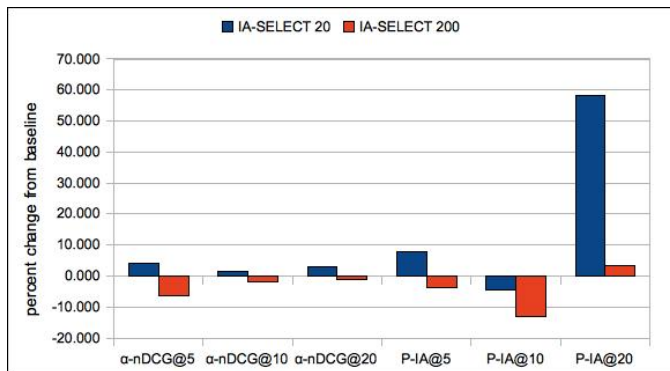


Figure 3: Percent change from baseline score according to all diversity measures for IA-SELECT reordering with 20 and 200 documents.

documents, experiment IA-SELECT 200, underperforms, giving the worst performance on all diversity scores except P-IA@20. To some extent as k increases the performance of the reordering algorithms improves. Figure 3 presents the reordering experiments' percent changes from the baseline for all diversity scores. Increasing k monotonically increases α -nDCG score for reordering 200 documents but not for reordering 20 documents. A greater improvement of the 200 document score with increasing k , as compared to the 20 document score, should be expected because, once k reaches 20, the 200 document reordering can improve its score by adding documents not in the top 20 as well as reordering documents already within the top 20, whereas the 20 document reordering can only improve its score by reordering documents within the top 20. Still, this advantage is not enough to improve beyond the score achieved by only reordering the top 20 documents.

4.3 Changing the influence of rank

Variations to the ϕ_v function are able to produce significant changes in the diversity scores. Increasing the influence of rank by increasing the convexity of the ϕ_v function will increase diversity scores up to a point. In Table 2 we present diversity scores for various ϕ_v functions using the reordering algorithm with 20 documents. As can be seen, $\phi_v(x) = x^2$ produces the best scores in all cases except P-IA@10.

Up to and including $\phi_v(x) = x^2$ α -nDCG scores increase as function convexity increases. With the P-IA scores we also see an increasing trend but it is much less pronounced. Ignoring rank, by setting ϕ_v to a constant, produces the lowest scores in all runs except P-IA@20, which is generally an outlier in terms of the relationship between its score and the ϕ_v function used. Increasing the exponential, and setting $\phi_v(x) = x^3$, decreases scores, which shows that excessively increasing the influence of rank will degrade performance.

4.4 Empirical upper bound

A drawback of the intent aware diversity metrics is that the range is not necessarily in $[0, 1]$. Unless there is a single perfect ordering for all subtopics the maximum value of IA scores will be less than 1 and the only so far known way to calculate the maximum value is through exhaustive search [1]. In addition to the possibility that the maximum IA score is analytically below 1, the specific empirical methods used in these experiments place additional limits on the maximum obtainable score. The reordering algorithms only take into account the top 20 or top 200 documents according to their BM25 ranking score. Therefore, if the perfect ordering relies on a document outside of the set being considered for reordering, it will be impossible to achieve the maximum diversity score. This limitation will apply equally to both α -nDCG and IA metrics.

To estimate an upper bound for the scoring metrics, given the experimental procedure used, we devise a method to produce reordered lists based on the known queries and

$\phi_v(x)$	α -nDCG@5	α -nDCG@10	α -nDCG@20	P-IA@5	P-IA@10	P-IA@20
constant	0.108	0.129	0.166	0.040	0.036	0.032
$1 + \log(x)$	0.129	0.150	0.171	0.047	0.045	0.033
x	0.142	0.155	0.176	0.046	0.046	0.031
x^2	0.151	0.157	0.180	0.055	0.044	0.049
x^3	0.149	0.156	0.179	0.052	0.040	0.032

Table 2: Diversity scores for IA-SELECT 20 reordering algorithm using $\phi_p(x) = 1 + \log(x)$ and variations in $\phi_v(x)$ as shown.

subtopics for documents. We first assign subtopics to documents based on the qrels file and then assign a uniform probability distribution to $P(c|q)$ in Equation 11 and $V(d|q, c)$ in Equation 12. Because we are reordering the list based on the subtopics and subtopic assignments that will be used when the list is evaluated, we expect the produced list to approximate the scoring metric’s upper bound.

The results of reordering based on subtopics mined from the qrels file are presented in Table 3. As predicted, these scores are significantly higher than the baseline scores or scores achieved through reordering based on clusters generated by PLSI in all cases except P-IA@20 for QRELS 20. In this case the QRELS 20 score beats the baseline and reordering with 200 documents scores but is significantly less than reordering with 20 documents. Note that these scores are presented as only *estimated* upper bounds. There is no guarantee that these are true upper bounds (indeed we see an example where this is not the case) and testing for true upper bounds would require an exhaustive search. The α -nDCG@{5, 10, 20} score is perfect for 1 query when reordering based on 20 results and perfect for 2 queries when reordering based on 200 results. For both 20 and 200 results the P-IA scores are never perfect overall and for all evaluated metrics there remain some queries where the score is 0.

In additional results using known subtopics we find that modifying $V(d|q, c)$ to give more weight to documents with higher rank decreases diversity metric scores, the opposite of what occurs when using induced subtopics. This result is expected if the subtopics we are using are more relevant to improving diversity scores than the rank of documents, which we presume them to be if we are using accurate subtopics. This provides anecdotal evidence that for some scores these estimates can provide a reasonable upper bound.

4.5 Examining scores per query

As mentioned before, subtopics are not evenly distributed among queries and we might expect that this will have an impact on how and why our diversification system produces the scores it does. To measure this effect, as well as measure why reordering with 20 results is able to improve on the baseline scores, we review the α -nDCG@5 scores per query as presented in Figure 4. The line graph represents the scaled number of subtopics per query and shows there is no significant pattern between the number of subtopics in a query and the baseline or reordered score for that subtopic.

By looking at individual queries we see that the reordered list produces better results by matching or edging out the baseline on most queries and significantly beating the baseline on a few queries. For queries 2 and 11 the reordered list scores significantly higher, for query 11 the baseline score is

0. This method of improving the baseline is likely a result of the heavy influence put upon document rank by the algorithm. It is a conservative approach, where reordering occurs in cases where both document rank and subtopic probability are high.

5. DISCUSSION

The results presented above show that a simple clustering system with largely untuned parameters is capable of improving diversity scores through reordering. However, the improvement is small and requires placing significant weight on document rank. This could be caused by poor quality subtopics, problems in the diversification algorithm and/or the probability measures that it uses, or by poorly set parameters. Most likely a combination of all these factors is limiting the evaluated systems’ improvement in diversity scores.

5.1 Generating accurate subtopics

In an approach based on reordering results according to subtopics, the effectiveness of the system is dependent on the ability to generate accurate subtopics, or the ability to generate subtopics aligned with the subtopics that will be used by the scoring function (let these be the *known clusters*). A lack of agreement between generated clusters and known clusters will result in a poorly performing system, but this is not necessarily an indication of a poorly performing clustering algorithm. There are multiple ways to cluster documents and the generated clusters may be accurate but different from the known clusters. Putting this aside, let us

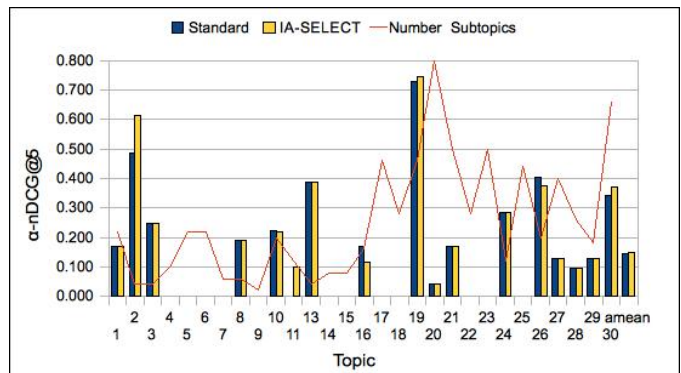


Figure 4: α -nDCG@5 per query for IA-SELECT with 20 documents and the baseline. The last column is the arithmetic mean. The line plots the number of subtopics per query normalized to the range [0, 0.8].

Experiment	α -nDCG@5	α -nDCG@10	α -nDCG@20	P-IA@5	P-IA@10	P-IA@20
QRELS 20	0.324	0.305	0.287	0.080	0.050	0.033
QRELS 200	0.611	0.632	0.621	0.134	0.102	0.079

Table 3: Empirically estimated diversity score upper bounds. QRELS 20 reorders the top 20 documents based on the known subtopics and QRELS 200 reorders the top 200 documents.

assume the known clusters are the “correct” clusters, that a perfect clustering algorithm generates these clusters, and that the better the clustering algorithm we use the closer to the known clusters we come.

Under these conditions, it’s important to consider that PLSI is only one clustering algorithm among many and no longer the dominant algorithm. Latent Dirichlet allocation [2], pachinko allocation models [12], and many other methods ([16] and [21] among others), have been shown to produce superior performance in clustering tasks. We should determine if these alternative clustering algorithms can produce more representative clusters and through this higher diversity scores.

Additionally, the specific number of clusters was set at 20 throughout all experiments. Future experiments should evaluate how modifying the number of clusters affects diversity scores. Overestimating the number of clusters should produce excessive reordering as documents are reordered due to clusters that they’re not evaluated against. However, if the clusters are accurate, this shouldn’t pose a problem because clusters that form sub-clusters of a known cluster should be similar in content and reordered in a similar manner. On the other hand, underestimating the number of clusters should produce too little reordering and be detrimental to score. We therefore expect that having the number of clusters closer to the number of known clusters will increase scores and that overestimating the number of clusters generally produces better scores than underestimating. This hypothesis should be empirically tested. Note that adjusting cluster size based on the known number of subtopics for a query would produce unrealistic results since in a real experiment, or in the real world, this number would not be known. This does not preclude an adaptive algorithm that finds the maximum diversity score for a variable number of clusters. An alternative method is to parameterize the cluster size and modify it depending on perhaps the cardinality of the vocabulary of the documents being clustered, or other heuristics.

We suspect that the success of using methods that put significant emphasis on a document’s rank is due primarily to the poor quality of induced clusters. As demonstrated by experiments generating an upper bound using the known clusters, increasing cluster quality and decreasing rank’s importance correlate with increasing diversity scores. In evaluations of alternative clustering methods various strategies for incorporating rank should be used so that the diversity benefits of a new clustering algorithm aren’t masked by heavily weighting rank.

5.2 Diversification algorithm

The significant impact modifying the influence of rank has on the diversification algorithm informs us of the function of this algorithm in addition to the quality of the subtopics. Although the ϕ_v function was tested in multiple forms there

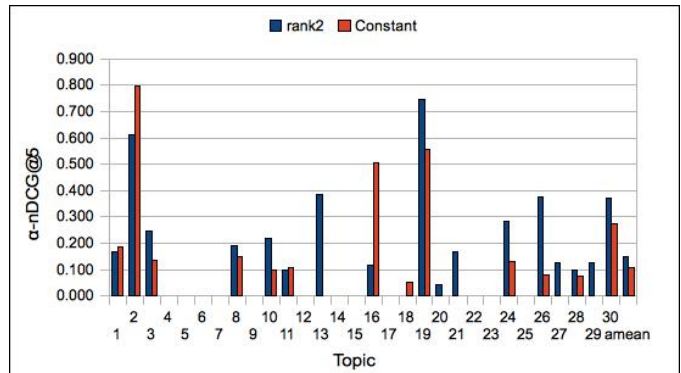


Figure 5: α -nDCG@5 per query for IA-SELECT with 20 documents using $\phi_v(x) = x^2$ (rank2) and $\phi_v(x) = 1$ (Constant).

are still other variations that may produce improved diversity scores. A starting point is to find the value $1 < n < 3$, in $\phi_v(x) = x^n$, such that diversity scores are maximized. As Table 2 shows, $n = 2$ doesn’t maximize all diversity scores and changing n will likely increase some scores while lowering others. The fact that increasing n leads to decreasing scores at some point is an encouraging result because it shows that the clusters are providing us with valuable information about how to reorder results and increasing the influence of rank to infinity (thereby generating the original ordering) is not optimal.

A more effective algorithm would be capable of biasing towards the original ranking of documents when doing so benefits diversity scores and biasing away from the original ranking when it is detrimental to diversity scores. Figure 5 presents the α -nDCG@5 score per query using a method that heavily weights rank, $\phi_v(x) = x^2$, and a constant method that ignores rank, $\phi_v(x) = 1$. Although the overall score of the constant method is much lower, we see that on certain individual queries (queries 2 and 16) it significantly outperforms the method that heavily weights rank. We suspect that the reason the constant method outperforms on these queries is that the clusters generated for them closely match the known clusters.

Considered at a general level, the diversification algorithm presented is quite limited because it only uses clusters to create its reordering. As shown in [6], incorporating multiple features in the reordering can improve diversity scores. Future implementations should investigate various methods of accounting for additional document features when reordering.

5.3 Parameter tuning

Beyond what has been mentioned above, adjusting various other parameters may improve performance. The number

of documents reordered, k , was tested at 20 and 200. As Figure 3 shows, the improvement in α -nDCG is highest for the 20 document reordering at 5, the smallest document set size at which it was evaluated, and highest for the 200 document reordering at 20, the largest document set size at which it was evaluated. This pattern, in combination with the increasing α -nDCG score for reordering 200 documents as the evaluation size increases, suggests that there is an optimal value of k for improving diversity metrics evaluated at their various document sizes.

Patterns seen in α -nDCG scores do not carry over to P-IA scores where the size of the evaluation set seems uncorrelated with improvements over the baseline. Interestingly, the estimated upper bounds have a very difficult time improving P-IA score and at 20 documents actually score lower on P-IA@20 than the IA-SELECT reordering algorithm using 20 documents, although still an improvement over scores for the baseline and reordering the top 200 documents. PLSI is able to capture something informative about reordering the top 20 documents that is not found by reordering using known clusters. Although, when we increase the upper bound estimation algorithm to reordering 200 documents it dramatically improves the P-IA@20 score and whatever advantage PLSI had is lost.

The upper bound diversity scores increase across the board as the number of documents is considered. We expect that increasing the considered number of documents further would continue to improve scores monotonically, but by decreasing amounts. This is because the algorithm will not inaccurately reorder documents if it knows the correct subtopics, and can therefore only benefit from having more documents to reorder.

5.4 Parameterizing diversity

In many of the reviewed diversification systems there is an adjustable value (such as λ in Equation 1 or α in Equation 8) specifying how broad a user's information needs are, or how tolerant a user is of lower relevance or higher redundancy. This is best construed as a dynamic value that varies for different types of users or searches and can change while interacting with the search system. In the result diversification system we present, we would expect changes in the importance of diversity to be expressed by changing the influence of rank in the ϕ_v function. We would further expect that the more influential the ranks of documents are, the less diverse reordered results are, and vice versa. However, Table 2 demonstrates that this is not strictly the case, diversity scores improve as the influence of rank increases. To more accurately characterize the relationship between rank and diversity we propose that after the influence of rank has been increased to produce the maximum diversity scores, further increasing the influence of rank reduces the importance of diversity.

In recent work Fuhr defines a cost function based on the effort and average benefit of viewing reordering results. From this he defines a probabilistic ranking principle for interactive information retrieval (PRP for IIR) which selects a result ordering based on minimizing the cost function [7]. The presented result diversification system can be integrated with this PRP for IIR by adjusting the importance of diversity (through adjusting the choice of ϕ_v) in response to the predicted effort and benefit of viewing redundant versus novel documents.

6. CONCLUSIONS

With the exceptional growth in online documents and internet users comes an increase in query ambiguity as well as possible information needs, and with this a justification for retrieval methods designed to satisfy these multifarious needs. Diversity research has moved beyond independent analysis of document novelty and relevance (as in MMR) to measuring a document's contribution in relation to the additional information it provides, given an external partition on information content, i.e. categories or nuggets. In this paper we have shown that using PLSI to create an external partition that is used to reorder search results improves diversity. We have seen that the functioning of the reordering algorithm is sensitive to, and can be tuned through, changes in the influence of a document's original rank. Decreasing the influence of rank puts more trust in the accuracy of induced clusters while increasing the influence of rank puts less trust in clusters.

Future work includes inducing clusters with alternative clustering algorithms as well as using different values for parameters used by the clustering algorithms. By using the known clusters in generating the reordering we see that significant improvements over the baseline score can be achieved. The known cluster reordering scores highly precisely because it uses accurate clusters. This demonstrates that as we improve cluster accuracy, and reduce reliance on rank, improved scores are obtainable.

Diversification is a relatively new area of research in information retrieval with links to tasks such as subtopic retrieval, question answering, and ad hoc retrieval. Diversity research can benefit by incorporating ideas from all these areas. We expect significant benefits to diversity can be obtained by incorporating ideas from topic modeling, where creating accurate clusters with accurate interrelations between clusters is a primary goal.

7. REFERENCES

- [1] R. Agrawal R. et al. Diversifying Search Results. *WSDM '09*, 2009.
- [2] D. Blei, A. Ng, and M. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research* 3, pp. 993-1022, 2003.
- [3] J. Carbonell and J. Goldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. *SIGIR '98*, 1998.
- [4] M. Carterette and P. Chandar. Probabilistic Models of Novel Document Rankings for Faceted Topic Retrieval. *CIKM '09*, 2009.
- [5] C. Clark et al. Novelty and Diversity in Information Retrieval Evaluation. *SIGIR '08*, pp. 659-666, 2008.
- [6] Z. Dou et al. Microsoft Research Asia at the Web Track of TREC 2009. <http://trec.nist.gov/pubs/trec18/papers/microsoft-asia.WEB.pdf>.
- [7] N. Fuhr. A Probability Ranking Principle for Interactive Information Retrieval. *Information Retrieval*, Springer 2008.
- [8] S.Gollapudi and A. Sharma. An Axiomatic Approach for Result Diversification *WWW 2009*, 2009.
- [9] V. Jijkoun and M. de Rijke. Overview of WebCLEF 2007. *Advances in Multilingual and Multimodal Information Retrieval*. Springer, 2008.

- [10] K. Jones, S. Walker, and S. Robertson. A Probabilistic Model of Information Retrieval: Development and Comparative Experiments. *Information Processing and Management*, 36(6):779-840, 2000.
- [11] The Lemur Toolkit. <http://www.lemurproject.org/>.
- [12] W. Li and A. McCallum. Pachinko Allocation: DAG-Structured Mixture Models of Topic Correlations. *Proceedings of the 23rd International Conference on Machine Learning*, 2006.
- [13] C. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.
- [14] M. Sanderson. Ambiguous Queries: Test Collections Need More Sense. *SIGIR '08*, 2008.
- [15] A. Schein et al. Methods and Metrics for Cold-Start Recommendations. *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2002.
- [16] H. Wallach, D. Mimno, and A. McCallum. Rethinking LDA: Why Priors Matter. *NIPS*, 2009.
- [17] J. Wang and J. Zhu. Portfolio Theory of Information Retrieval. *SIGIR '09*, 2009.
- [18] H. Zeng et al. Learning to Cluster Web Search Results. *SIGIR '04*, 2004.
- [19] C. Zhai, W. W. Cohen, and J. Lafferty. Beyond independent relevance: Methods and evaluation metrics for subtopic retrieval. *Proceedings of SIGIR*, pp. 10-17, 2003.
- [20] C. Zhai and J. Lafferty. A risk minimization framework for information retrieval. *Information Processing & Management*, pp. 31-55, January 2006.
- [21] D. Zhou and C. Burges. Spectral Clustering and Transductive Learning with Multiple Views. *Proceedings of the 24th International Conference on Machine Learning*, 2007.