Component-based Analysis of Dynamic Search Performance*

AMEER ALBAHEM, DAMIANO SPINA, FALK SCHOLER, and LAWRENCE CAVEDON, RMIT University, Australia

In many search scenarios, such as exploratory, comparative, or survey-oriented search, users interact with dynamic search systems to satisfy multi-aspect information needs. These systems utilize different dynamic approaches that exploit various user feedback granularity types. Although studies have provided insights about the role of many components of these systems, they used black-box and isolated experimental setups. Therefore, the effects of these components or their interactions are still not well understood. We address this by following a methodology based on ANalysis Of VAriance (ANOVA). We built a Grid Of Points that consists of systems based on different ways to instantiate three components: initial rankers, dynamic rerankers, and user feedback granularity. Using evaluation scores based on the TREC Dynamic Domain collections, we built several ANOVA models to estimate the effects. We found that: (i) although all components significantly affect search effectiveness, the initial ranker has the largest effective size; (ii) the effect sizes of these components vary based on the length of the search session and the used effectiveness metric, and (iii) initial rankers and dynamic rerankers have more prominent effects than user feedback granularity. To improve effectiveness, we recommend improving the quality of initial rankers and dynamic rerankers. This does not require eliciting detailed user feedback, which might be expensive or invasive.

CCS Concepts: • **Information systems** → **Retrieval effectiveness**; **Information retrieval diversity**; **Users and interactive retrieval**; Query reformulation; Document structure.

Additional Key Words and Phrases: Subtopic retrieval, interactive search, evaluation, component-analysis, ANOVA

ACM Reference Format:
Ameer Albahem, Damiano Spina, Falk Scholer, and Lawrence Cavedon. 2021. Component-based Analysis of Dynamic Search Performance. *ACM Transactions on Information Systems* 1, 1, Article 1 (January 2021), 49 pages. https://doi.org/10.1145/3483237

1 INTRODUCTION

In many search scenarios, users engage with search systems to find and synthesize relevant information in order to perform certain tasks or acquire knowledge. During these scenarios, users might issue multiple queries, paginate through search results or annotate documents. To serve users better, several dynamic search systems that can learn from user feedback and adapt subsequent results have been developed [17, 35, 43, 48, 78, 87, 88].

*This is a preprint of an article accepted for publication in ACM Transactions on Information Systems (TOIS) on August 2021

Authors' address: Ameer Albahem, ameer.albahem@gmail.com; Damiano Spina, damiano.spina@rmit.edu.au; Falk Scholer, falk.scholer@rmit.edu.au; Lawrence Cavedon, lawrence.cavedon@rmit.edu.au, RMIT University, Melbourne, Australia.

1046-8188/2021/1-ART1 $15.00
https://doi.org/10.1145/3483237

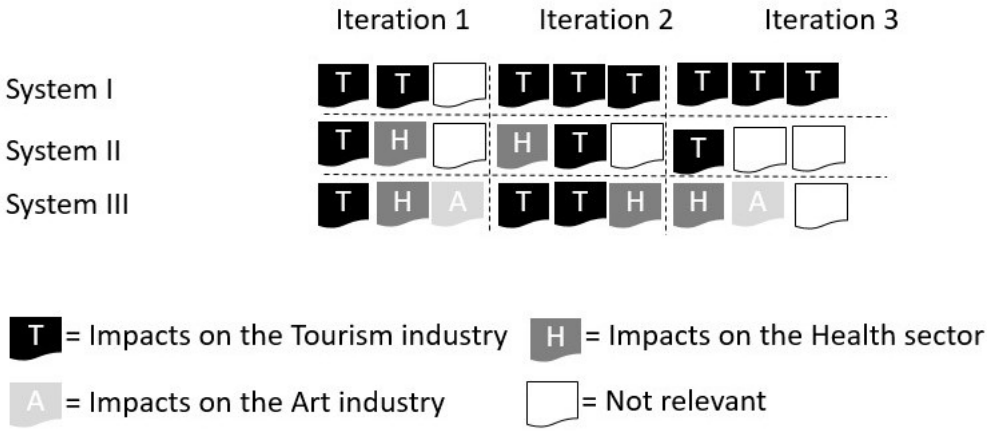


Fig. 1. Different dynamic search systems addressing a multi-aspect information need ("Economic impacts of COVID-19").

In tackling complex information needs, it might not be enough to focus on finding relevant documents. For instance, consider a person who wants to know about the economic impacts of the coronavirus pandemic (COVID-19) that started in 2019. To fully understand and comprehend the topic, they might need to read about multiple impacts of COVID-19 on the tourism industry, the health sector, and the art industry. In such scenarios, studies have shown that user satisfaction with the search systems depends on many search qualities such as retrieving relevant documents (topical relevance), reducing redundancy (novelty or diversity), and minimizing user effort [33].

Ultimately, to serve users better, search systems should have components that accommodate the above aspects in the ranking process. To that end, there are many components that could affect the performance of search systems. One component is user feedback. Ideally, the more we know about the information needs and the relevance of seen documents in a search session, the more likely we could serve the users better in consecutive interactions. Consider the examples of different dynamic search systems that a user might interact with to understand the economic impacts of COVID-19 (shown in Figure 1). Consider the case that the user indicates to the system that a document contains key relevant information for a query aspect. Since the user might have sufficient information to satisfy the aspect with the found key information, the system might focus on other aspects. However, suppose the user marks documents as relevant or not relevant. In this case, the system might retrieve similar documents to the inspected documents, which might be less useful for the user. Many studies have shown the benefits of utilizing relevant passages [10], highlighted and annotated texts [29, 68]. Other studies showed the benefits of utilizing topically structured user feedback [43].

Another source of variance in dynamic search is the initial ranker. Consider **System I** in Figure 1 with the high precision and low subtopic recall in its first response to a user query. Even though a dynamic algorithm might effectively incorporate user feedback, the ranking might be suboptimal. The initial ranker will always provide documents related to seen subtopics.

The dynamic approach or algorithm that utilizes user feedback also affects the effectiveness of an interactive search system. The dynamic approach has been the main focus of various interactive search systems. Systems have utilized different approaches such as query expansion [40], content similarity [9], interactive diversification [54] and reinforcement learning [43].

Although researchers have exploited various granularity types of user feedback using different dynamic ranking approaches, they mostly carried out a black-box evaluation. In a black-box evaluation, we attribute the effectiveness attained by systems to the systems themselves without consideration of the steps or the building blocks that might affect the performance. Ferro and Silvello [27] succinctly summarized the limitations of this experimental evaluation. First, it obscures the ability to understand how different components interact with each other and contributed to overall effectiveness of systems. Second, it assumes search system designers know the best configuration beforehand, which combination of components is best studied for a given task. Third, it hinders the possibility of determining which components are more convenient to invest effort and resources because they or their combinations have the most significant effects on performance.

Recently, a white-box evaluation of a search system's effectiveness has received attention [25–27]. In this setup, we investigate the effects of the search systems' internal components on effectiveness. Ferro and Silvello [26] introduced the component-based analysis to evaluate search systems. The framework first builds a Grid Of Points (GoP) that consists of all systems of combinations of ways to instantiate the search system's components. We then determine each component's contribution via a variance analysis of the effectiveness scores– ANalysis Of VAriance (ANOVA). The effectiveness scores are measured by metrics and treated as the dependent variable. The components are treated as independent variables.

This research is a first step toward estimating the effects of various factors in dynamic search. More specifically, we focus on a dynamic search setup that tackles multi-aspect information needs with explicit user feedback, as instantiated in the TREC Dynamic Domain track [79–81]. This is motivated by a number of reasons. First, the TREC Dynamic Domain setup caters to satisfying the overall information needs of users; hence it aligns more with satisfying multi-aspect information needs than the multi-query session track setup [13], which focuses on satisfying the last query issued by a user. Second, these test collections include graded relevance judgments, at both the aspect and passage levels. Therefore, it allows us to investigate the effects of various components of dynamic search that have typically been studied in isolated setups. Third, the richness of the resources also helps to avoid issues related to users' ability to submit and formulate effective queries. Given this setup, we study three factors: the initial ranker, the dynamic reranker, and user feedback granularity.

In this work, we aim to investigate following main research question: *How do different components of dynamic search systems affect search performance?* We further decompose it into three research sub-questions:

- RQ 1: What are the effects of the topics and system factors on the performance of dynamic search?
- RQ 2: What are the effects of the three factors of a dynamic search system (initial ranker, dynamic reranker, and user feedback granularity)?
- RQ 3: Do the effects of factors differ as we progress in the search session?

The rest of the paper is organized as follows. In Section 2, we review related work. In Section 3, we describe how we apply the ANOVA-based component analysis [27] to analyze effects of various factors on the performance of dynamic search. Section 4 describes the setup of our experiments. We report and discuss the analysis in Section 5. Lastly, Section 6 concludes the work.

2 RELATED WORK

2.1 Dynamic Search

Dynamic search refers to various interactive search tasks. Indeed, researchers have organized many tasked to tackle these tasks. The Interactive track [55] evaluates an interactive search setup with

live users where the focus was using user feedback to improve ad hoc search. The TREC Relevance Feedback track [11] addresses the lack of advancements of relevance feedback algorithms. This was attributed to the difficulty of separating the impact of factors such as the initial ranker, the feedback documents used, and the feedback algorithms themselves [11]. The TREC Session track [13] facilitates research on multi-query search sessions. A scenario where users issue several queries to find documents that satisfy their information needs. The systems were evaluated based on their ability to satisfy the mth query. Therefore, it was not evaluating the overall process. The TREC Dynamic Domain track [79] was created to foster research in complex search tasks, which are exploratory and run over multiple rounds of interactions. The final system output is then judged based on how well the system performed in satisfying the user goal of the multi-aspect information needs. As this task addresses evaluating the overall system performance of multi-aspect retrieval, we utilized its resources.

2.2 Retrieval Approaches

Many retrieval approaches have been proposed to tackle multi-aspect information needs in static and dynamic setups. We briefly describe these approaches.

Search result diversification. Search result diversification is a research area that initially focuses on tackling ambiguous and underspecified queries [65]. The essence of search result diversification is how to rank documents to balance novelty and coverage. Novelty means that documents should contribute new relevant contents (aspects) to what a user might have seen. On the other side, coverage relates to the depth of covering a specific aspect or interpretation. Diversification methods can be divided based on their approach to measure the utility of documents with respect to the aspects (subtopics) of a query into implicit and explicit [65]. Implicit diversification methods utilize the content of the documents to diversify search results. Carbonell and Goldstein [12] proposed the seminal Maximum Marginal Relevance (MMR) method that balances the relevance of a document to a query by its dissimilarity to other documents already selected to present to users. Explicit diversification methods tackle diversification by directly estimating the relevance of documents to the aspects of the query. Agrawal et al. [2] proposed a framework that maximizes the probability of selecting at least one relevant document for each aspect (IA-Select). Santos et al. [64] proposed a probabilistic framework called xQuAD (eXplicit Query Aspect Diversification) that represents the multiple aspects of a query as a set of sub-queries. They also proposed methods to generate the sub-queries and estimate their importance using query reformulations and their statistics from commercial search engines. Using TREC Web diversification collection [19], they showed that xQuaAD is superior to MMR [12] and IA-Select [2]. Raman et al. [60, 61] proposed a two-level algorithm that diversifies search results in a first ranking to allow users find documents relevant to multiple subtopics. Raman et al. [60] first predicted which multi-query sessions are about multi-aspects information needs, then developed a two-level ranking to present search results to users. Recently, Maxwell et al. [50] found searchers are more successful with diversified rankings at recognizing relevant documents and learning more about the subtopics of the search tasks. In recent years, different supervised approaches that adapt learning-to-rank and neural ranking methods to search result diversification have been proposed. These methods extend both implicit [74–77, 89] and explicit [34, 45, 57] diversification models. Although these approaches outperform unsupervised search result diversification approaches, they have yet not been tested – to our knowledge – in dynamic search scenarios. We leave to future work the design of more sophisticated dynamic search systems that incorporate diversity-aware learning-to-rank and neural ranking methods [77].

Interactive diversification. Many systems have used various explicit or implicit diversification techniques as a principle approach to balance exploitation and exploration in tackling complex information in an interactive setup. Moraes et al. [54] utilized the xQuAD diversification method and its recent hierarchical intent variant [31] to diversify search results, which performed better than other diversification techniques such as topic modeling or k-means clustering utilized by Joganah et al. [37]. Similarly, Zhang et al. [84] used the Rocchio algorithm and the subtopic-based and passage-based feedback to generate multiple queries used by xQuAD. Their results achieve competitive results to other approaches that utilize deep reinforcement learning algorithms [71, 78]. Using many explicit diversification methods, Moraes [52] experimented with various methods to model subtopics using subtopic-, passage-based and graded relevance feedback. These methods were superior to the MMR diversification and relevance model methods.

Query expansion. Much research in interactive search has exploited user interaction to reformulate user queries to better approximate user information needs. Rocchio [62] proposed a query reformulation method within the vector space model that selects terms based on their weights after subtracting the weights of terms found in the query and relevant documents by their weights in non-relevant documents. Lavrenko and Croft [40] proposed the Relevance Model that expands queries using top weighted terms generated by a language model induced from the (pseudo) relevant documents. Levine et al. [42] extended the relevance model to incorporate the dynamic changes in user information needs in multi-query search sessions. Rahimi and Yang [58] explored different relevance models to expand user queries using fine-grained user feedback.

Similarity distance. This approach treats the search process as finding more similar documents given the user feedback or moving the document search space away from non-relevant documents. Bo et al. [9], Jiyun and Yang [36] ranked documents based on their relevance to the user query and similarity to seen relevant documents. In two-level interactive diversified rankings, Raman et al. [61] found after building the first level (page) of diversified search results, generating the second level of ranking based on the cosine similarity to clicked documents in the first page was competitive to many methods that use a supervised machine learning approach to generate the second level of rankings.

Reinforcement learning. With the existence of user interaction, many studies have modeled interactive search as a reinforcement learning problem. That is given a set of actions, the search system needs to choose an action that maximizes a reward function. The reward function is usually a metric that we calculate using the implicit or explicit user feedback. Jin et al. [35] instantiated a multi-page search on a diversification collection that leveraged user feedback in the first page to develop better rankings in the next pages. Li et al. [43] investigated using a query pool in finding relevant documents, where they modeled selecting which queries to search with as a multi-armed bandit problem. Using the number of relevant documents as a reward policy, their approach was superior to the Rocchio algorithm in various collections. In multi-query search sessions, Luo et al. [48] modeled the interaction between a search system and a user as a dual-agent game, where these agents cooperatively aim to maximize their long term rewards. Yang and Yang [78] and Tang and Yang [71] utilized a contextual bandit algorithm that learns the best action to perform in interactive search with explicit user feedback. Recently, Zhiwen and Yang [87] proposed to compress an entire corpus into a global low-dimension deep learning representation, then they designed a reinforcement agent to utilize rich user relevance feedback from previous iterations to select documents.

2.3 Component-based Analysis of Search Systems

Information retrieval systems are inherently complex. Moreover, offline evaluation of information retrieval systems and the Cranfield paradigm are built on the notion that topic variability plays an important role in the understanding of system performance [30]. Building a simple ad hoc search method requires many steps or components. A researcher or practitioner needs to determine the tokenizer, the stemming algorithm, and the stop-word list. A typical diversification method consists of two components: an initial ranker that generates an initial ranked list of documents for reranking, and a diversification algorithm that reranks documents in the initial list [65]. The diversification algorithm itself might consist of multiple components. For instance, the xQuAD framework consists of parts such as relevance estimation, sub-query generation, and importance estimation. Many of the interactive search systems submitted to the TREC Dynamic Domain track consist of multiple complex components. Moraes et al. [53] classified their components into four types: initial ranker, dynamic reranker, subtopic modeling, and stopping strategy.

Studies have found that different instantiations of system components lead to different search effectiveness as it is the case in ad hoc search [27], diversification algorithms [22, 64], and interactive search [50, 53].

In the following, we describe several research lines that focus on understanding the search quality of a search system under different instantiations of its building blocks.

Parameter sensitivity. Many search methods we utilize have parameters. The BM25 retrieval method [69] has the b and k parameters, whereas a language modeling approach with the Dirichlet smoothing method has the μ free parameter. Diversification algorithms, such as xQuAD and MMR, have a free parameter (λ). The parameter balances the contribution of the relevancy and the diversity of documents in the ranking. A common practice is analyzing the sensitivity of a method to its free parameters by running different values and evaluating the system's effectiveness. For instance, Santos et al. [65] studied the sensitivity of the free parameter λ.

Black-box analysis. The second line of research performs a black-box evaluation to understand the impacts of various choices in instantiating complex search systems [11, 53, 64]. There are three analysis types in this approach: component-of-interest, one-component at a time, and Grid of Points. In these analysis types, we assess the impacts of components by comparing their overall scores on a particular effectiveness metric. The difference between these types is the detailed exploration of all combinations of various instantiations of components.

In the component-of-interest analysis, researchers investigate two or more system components while holding all other components fixed. For instance, Dang and Croft [22] proposed term-level diversification that uses terms instead of queries as explicit representations of the aspects of information needs, then studied how the term-level representation performs with two different diversification algorithms: xQuAD [64] and PM-2 [21].

In the one-component at a time analysis, researchers iteratively pick a component to study, while keeping other components fixed with particular instantiations. In analyzing the parts of xQuAD, Santos et al. [64] studied each of them one at a time. They then summarized the impacts of these components on the overall search system performance. Moraes et al. [53] followed this approach in understanding the impacts of interactive search components in specialized domains.

Another setup is the Grid of Points (GoP) black-box comparison. Here the researchers will evaluate all possible combinations of the various instantiations of the search system components [49].

Ferro and Silvello [27] identified few limitations of a black-box evaluation approach. First, the evaluation does not show how much variance of the system's effectiveness is attributed to particular

components or their interactions. Therefore, researchers can not tell which component to focus on or study. Second, the approach does not consider the effect of the topic on the analysis, which significantly affects system variance [27]. In addition to these limitations, with many components and different instantiations, presenting or interpreting the GoP setup results might be difficult.

Factorial design analysis (White-box evaluation). Ferro and Silvello [26] proposed to perform a crossed factorial design-based analysis. The basic idea is that we treat each component as a factor or independent variable that affects the system's effectiveness (the dependent variable). For each factor (component), there are different levels (instantiations of the component). A factorial design is then set up by evaluating each topic under all systems built by utilizing various combinations of the components. We then perform an ANalysis Of VAriance (ANOVA) to understand the effects of these components. Back in 1999, Banks et al. used ANOVA– in addition to other strategies – to analyze ad-hoc retrieval systems in TREC-3 [8]. They found that there is a strong interaction effect between system and topic factors in terms of average precision. Recent studies have followed this methodology to estimate the effects of different factors in the performance of different information retrieval systems. Voorhees et al. used sharding of a test collection to replicate system comparisons, and applied a version of ANOVA based on bootstrapping to analyze system-topic interactions to increase the sensitivity of the evaluation of system effectiveness [72]. Ferro and Sanderson [25] proposed several ANOVA models that also incorporate the effects of sharding document collections to study ad hoc search performance. A recent reproducibility study by Faggioli and Ferro [23] found that more traditional ANOVA approaches such as the ones proposed by Ferro and Sanderson [25] are generally more stable with respect to random shards, and less computationally expensive, while bootstrap ANOVA [72] has more sensitivity in terms of identifying statistically significant differences among systems. Zampieri et al. [82] used the ANOVA framework to estimate the effects of various factors such as the topics and the system components in topic difficulty. Roitero et al. [63] used ANOVA to model the effects of user attributes such as user age, profile age, and gender in user interactions in music recommendation.

2.4 Analysis of Dynamic Search Components

A few attempts have sought to understand the impacts of components of dynamic search systems on effectiveness. The TREC Relevance Feedback aimed to uncover the effects of the initial retrieval, the number of relevant documents, and the relevant documents' quality. Brondwine et al. [10] studied the impacts of exploiting unit granularity of user feedback to expand user queries. Li et al. [43] investigated exploring the topical granularity within the relevance feedback and multi-armed bandits algorithms. Thought it showed that using bandit algorithms is better than the Rocchio relevance feedback algorithm, it used a fixed baseline method and assumed the overall performance was indivisible. Moraes et al. [53] classified systems submitted to the Dynamic Domain track into four main components: an initial ranker, a dynamic reranker, a subtopic modeler, and a stopping strategy. The authors then performed a one-component-at-time analysis. The analysis has shown that the initial ranker, dynamic ranker, and subtopics modeler components will impact performance. However, the study focused mainly on diversification algorithms and fine-grained relevance feedback. It also does not informs us how much variance in performance was due to these components or their interactions. The evaluation also was carried out using the Average Cube Test ACT metric [79], which prior research have found it might produce counter-intuitive results [5] and inferior in its ability to measure topical relevance, diversity, and user effort [4]. Nevertheless, we incorporated Moraes et al. [53]'s taxonomy in the identification of the factors to investigate.

2.5 Additivity of Search Effectiveness

The work in this paper is complementary to the work of search performance additivity. Armstrong et al. [7] first introduced the concept of performance additivity and the problem of comparing against weak baselines. Although performance additivity in a dynamic search system is not our focus, it sheds light on the importance of component-based analysis in evaluating complex search systems. In many search tasks, such as learning to rank, search result diversification, and interactive search, we utilize initial rankers to generate a candidate lists of documents for reranking. In search result diversification, Akcay et al. [3] and Kharazmi et al. [39] found that diversification algorithms tend to improve the performance of weak baseline methods, which are usually their initial rankers. It showed that the choice of the initial ranker contributed to the performance of the sophisticated search system and shed light on the importance of understanding the contribution of the system components to performance. In dynamic search, Carterette et al. [13] investigated the possibility of dynamic search methods to improve under certain conditions that represent different details of user signals in the multi-query search sessions. Nevertheless, the evaluation setup was a black box or additivity oriented evaluation. As a result, it did not facilitate knowing whether performance differences were due to topics or the components utilized such as the user behavior signal conditions, the dynamic approach, and the initial ranker.

3 METHODOLOGY

In this work, we aim to estimate the effects of dynamic search components that have shown to impact on search effectiveness. In order to achieve that, we follow Ferro and Silvello [27]'s Generalized Linear Mixed Model (GLMM), in which ANalysis Of VAriance (ANOVA) is a special case where factors are categorical values. This model aims to explain the variance of data (an evaluation metric representing the dependent variable) in terms of controlled variables (factors we are interested in studying).

Ferro and Silvello [27]'s framework followed the Cranfield paradigm, based on experimental collections consists of a set of documents (corpus), topics, and relevance judgments. In this framework, we perform four main steps: identify factors, create Grid of Point (GoP) systems, evaluate effectiveness of the GoP systems, construct and estimates ANOVA models. In the following we describe our approach to perform these steps.

3.1 System Components

3.1.1 Initial rankers. Several studies utilize a retrieval method such as BM25 to present the first batch of results to users or generate an initial ranked list for secondary rankers such as a diversification algorithm. Several studies have found that the choice of the initial ranker has effects in search effectiveness. One of the TREC Relevance Feedback track's main findings is that the number of relevant documents in the initial retrieval before applying relevance feedback impacts on performance [11]. In multi-query search sessions, utilizing a different initial ranker method impacts on user behavior [50]. In search result diversification, the choice of the initial ranker significantly determines whether a search result diversification method is effective or not [3, 39]. Moraes et al. [53] found that the quality of the initial ranker in terms of precision and recall correlates with search effectiveness. As a result, we use the initial ranker as one of the main components to investigate.

To generate the initial ranked list of documents for reranking, we experiment with several well-known and widely used retrieval methods from ad hoc search such as the Vector Space Model (TFIDF), Language Modeling (LM) [56] with Dirichlet smoothing, Divergence From Randomness (DFR) [6], BM25 [69] and Pseudo Relevance Feedback [40] (PRF). The first four methods are bag-of-words standard retrieval methods. They also are widely used methods in academia and industry.

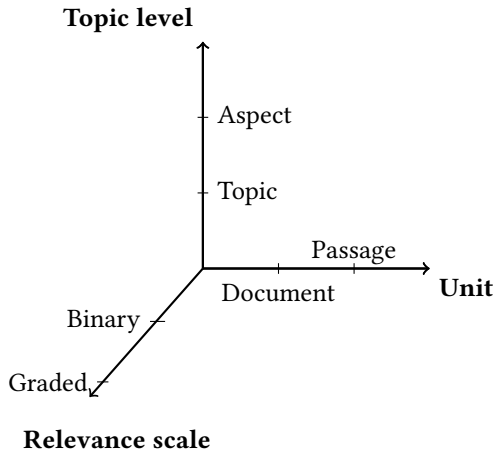


Fig. 2. User relevance feedback granularity dimensions.

We used the implementation and default parameter values found in the Lucene[1] retrieval toolkit. We experiment with PRF due to its strong performance in ad hoc search. In this method, we first retrieve the top 10 documents using LM; we then apply RM3 [40] to expand the initial user query and generate a new ranked list of documents. In all methods, we retrieve the top 1000 ranked documents and used them as the initial ranked list of documents.

Dynamic reranker (DR). In Section 2, we discussed many approaches to utilize user feedback such as: query expansion [40, 58, 62], finding similar documents (similarity distance) [9, 36, 41], search result diversification [53, 54, 84], and reinforcement learning [43, 48, 71, 78]. In this work, we focus on three approaches: query expansion, finding similar documents, and interactive diversification. We leave reinforcement learning and other supervised learning approaches such as neural ranking models [20, 44, 51], and diversity-aware learning-to-rank methods [77] for future work.

User feedback granularity (UFG). Generally, a user might provide feedback with more granularity in three main dimensions: topical level granularity, unit granularity, and relevance scale granularity (shown in Figure 2), where:

- Unit granularity: The detailed level of content that users rate. At the document level, users rate the overall relevance of a document. At the passage level, users rate the relevance of passages or text chunks from documents.
- Topic level granularity: The detailed level of the information needs to which the feedback is given. In topic-level feedback, users rate the relevance of information units to the overall topic of the information needs. In aspect-level feedback, users rate the relevance of information units concerning the aspects or the subtopics of the information needs.
- Relevance scale granularity: The relevance grades scale assigned to the information units. In a binary scale, users either judge units as relevant or not. In a graded scale, users assign a relevance grade to the information units such as key information, relevant, partially relevant, or not relevant.[2]

Given the dimensions, we have eight granularity conditions of user feedback as follows:

[1] https://lucene.apache.org/core/8_0_0/

[2] In extreme cases, users might provide information in 100 level or as probabilities. However, such approaches are less common within the research community.

- **TDB**: Topic-Document based-Binary relevance feedback.
- **TDG**: Topic-Document based-Graded relevance feedback.
- **TPB**: Topic-Passage based-Binary relevance feedback.
- **TPG**: Topic-Passage based-Graded relevance feedback.
- **ADB**: Aspect-Document based-Binary relevance feedback.
- **ADG**: Aspect-Document based-Graded relevance feedback.
- **APB**: Aspect-Passage based-Binary relevance feedback.
- **APG**: Aspect-Passage based-Graded relevance feedback.

In Section 3.2, we describe how we adapt existing dynamic rerankers to work under the above granularity conditions.

3.2 Grid of Points

Once we have identified the factors we are interested in studying, the second step is building a Grid of Point (GoP) systems that enable the cross-factorial design analysis. In this work, we create a GoP on the experimental collections by extending commonly used dynamic approaches that leverage explicit user feedback with different conditions of user feedback granularity.

The Grid of Points consists of different instantiations of three components: the initial ranker, the dynamic reranker, and the user feedback granularity condition. In the previous section, we identified five initial rankers, three dynamic rerankers, and eight different conditions of user feedback granularity. The task of the initial ranker is to retrieve a ranked list of documents for reranking. The dynamic reranker then utilizes the user feedback according to the user feedback condition we are experimenting with. The dynamic reranker then gives the next batch of documents to the simulator, a program that reads from relevance judgments. The feedback is then fed back to the dynamic reranker to rerank the remaining un-presented documents from the initial ranked list and present the next batch of documents. Each interaction of presenting a batch of documents to the simulated user is then counted as an iteration. We run this process for ten iterations. We now describe the dynamic rerankers and how we modified them to utilize various user feedback granularity conditions.

3.2.1 Query expansion. As discussed in Section 2, many dynamic search methods used query reformulation or expansion. As a result, our first dynamic reranker is the Relevance Model (RM) [40], a widely utilized and actively researched query expansion method.

This model aims to rank documents using terms that are representatives of information needs. It assumes that given a query q, there is a relevance language model RM that generates the terms in q and documents relevant to q. Given this model, we then rerank documents based on their similarities to the relevance model, which can be given using Kullback-Leibler divergence as follows:

$$\mathsf{KL}(\mathsf{RM}||d) = \sum_{w} P(w|\mathsf{RM}) \ \log \frac{P(w|\mathsf{RM})}{P(w|d)}. \tag{1}$$

Initially, with no retrieval being performed, $p(w|\mathsf{RM})$ is approximated using a maximum likelihood estimation model induced from the query q:

$$P(w|\mathsf{RM}) \approx P(w|q) = \frac{tf(w,q)}{|q|}, \tag{2}$$

where $tf(w,q)$ is the frequency of the term w in the query. In a blind pseudo relevance feedback, we approximate $P(w|\mathsf{RM})$ by the linear mixture models of the top k documents ($D_{q,k}$) from the initial ranked list of documents D_q in what is known as the RM1 model shown in the following

equation:

$$P(w|\text{RM1}) = \sum_{d \in D_{q,k}} \frac{P(d|q)}{\mathcal{Z}} P(w|d), \tag{3}$$

where $\mathcal{Z}$ is a normalizing factor, $P(w|d)$ is estimated using a unigram language model. It is a standard practice also to use only the top-weighted terms according to RM1 and normalize the model. In our implementation, we use the top 10 terms. Finally, RM1 is then interpolated with the original query to prevent query drift producing what is known as the RM3 model (Equation 4).

$$P(w|\text{RM3}) = \lambda\, P(w|q) + (1 - \lambda)\, P(w|\text{RM1}). \tag{4}$$

In this work, we set the parameter λ to 0.5, the default value in the Anserini implementation.[3] Generally, we expand the initial query using Equation 4. However, we might model RM1 differently based on the granularity condition of the user feedback. We describe these models below.

Modeling graded relevance. Although Lavrenko and Croft [40] originally proposed the relevance model to work with pseudo relevance feedback, researchers use this method to expand queries from relevant documents $D_F \subseteq D_q$ that users rate as relevant. In particular, the standard practice is to estimate $P(d|q)$ using the standard document model with Dirichlet smoothing ($P_{dir}(d|q)$) [83]. When users give binary-scale relevance feedback, $P_{dir}(d|q)$ allows us to approximate the relevance of documents to the query. With graded relevance feedback, the relevance grades or ratings of documents $g(q, d)$ have been used to model $P(d|q)$ [58, 86]. In this research, we also use relevance grades and induce a Graded Relevance Model (GRM1) as follows:

$$P(w|\text{GRM1}) = \sum_{d \in D_F} \frac{\mathcal{G}(g(q,d))}{\mathcal{Z}} P(w|d), \tag{5}$$

where $\mathcal{G}$ is a function that estimates the probability of document d being relevant to the query q.[4] In particular, we use a function that assigns higher probabilities to the documents with larger relevance grades. As a result, we use the probabilistic estimation provided in many evaluation metrics, such as the Expected Reciprocal Rank metric [15] as follows:

$$\mathcal{G}(g(q,d)) = \frac{2^{g(q,d)} - 1}{2^{gmax}}, \tag{6}$$

where $g(q, d)$ is relevance grade assigned to document d and $gmax$ is the maximum grade that could be assigned in the relevance judgments.

Modeling aspects. Several researchers have shown the benefits of selecting terms using multiple relevance models estimated from different sets of documents such as different clusters of relevant documents [66], multiple subsets of documents [59] or sets retrieved based on query variations [46]. Similarly, when expanding user queries using subtopic or aspect level user feedback, we are using multiple sets of documents that the user has explicitly structured based on the aspects of the query. Recently, Lu et al. [46] compared different methods to rank documents using multiple relevance models. The study found that selecting terms based on their average $P(w|*)$ from different models performed better than many methods such as the sum or the geometric mean of $P(w|*)$. As a result, we use this method to expand the query as follows:

$$P(w|\text{AriRM}) \approx \frac{1}{|\mathcal{A}_i|} \sum_{a \in \mathcal{A}_i} P(w|\text{RM}^{D_a}), \tag{7}$$

[3] https://github.com/castorini/anserini/releases/tag/anserini-0.6.0

[4] With binary relevance scale, $\mathcal{G}(g(q,d))$ will assign the same probability to all documents as the documents have a relevance grade of 1, hence $P_{dir}(d|q)$ facilitates differentiating which of these documents are more relevant to the query.

Table 1. Relevance model estimation under different granularity conditions of user feedback.

Condition	$P(w\|\text{RM})$	$P(d\|q)$
TDB	$P(w\|\text{RM1})$	$P_{dir}(d\|q)$
TDG	$P(w\|\text{GRM1})$	$\mathcal{G}(g(q,d))$
TPB	$P(w\|\text{RM1})$	$P_{dir}(\text{concat}(d_{psg})\|q)$
TPG	$P(w\|\text{RM1})$	$\mathcal{G}(\text{avg}(g,d_{psg}))$
ADB	$P(w\|\text{AriRM})$	$P_{dir}(d\|q)$
ADG	$P(w\|\text{AriRM})$	$\mathcal{G}(g(a,d))$
APB	$P(w\|\text{AriRM})$	$P_{dir}(\text{concat}(d_{psg})\|q)$
APG	$P(w\|\text{AriRM})$	$\mathcal{G}(\text{avg}(g,d_{psg}))$

where $\mathcal{A}_i$ is the set of aspects that have relevant documents up to iteration i and RM^{D_a} is the relevance model induced from seen relevant documents to aspect a (D_a). Note that we induce the model using GRM1 (Equation 5) or RM1 (Equation 3) based on the relevance scale granularity condition of the user feedback.

Modeling passages. In utilizing texts of passages from documents, studies have followed two approaches. In the first approach, we concatenate texts of relevant passages to represent a relevant document [10, 29], we then induce a relevance model from this new text representation. The second approach estimates the language model directly from the passages treating them as individual documents themselves [28, 58]. The first approach implicitly assumes binary relevance judgments. That is as all passages are equally relevant to the information needs, thus it is not an issue if we concatenate them. The second approach suffers from the fact the documents might be very short to model the importance of terms. It also suffers from the term mismatch problem. Based on initial experiments, we found that many passages might not contain the query terms as they might focus on certain subtopics of information needs. As a result, we follow the first approach as it allows better estimation of document relevance to a query and avoids the mismatch problem. When users provide passage-based feedback with graded relevance, we aggregate the relevance grades of passages to estimate the document relevance grade. In particular, we use the arithmetic mean, which Wu et al. [73] have shown to have a high correlation with document-level relevance grades.

To summarize, we expand the original query using the ten words that have the highest probabilities $P(w|\text{RM})$ induced from the relevance model (RM). Each relevance model also depends on how we estimate $P(d|q)$. Table 1 shows the various models to estimate $P(w|\text{RM})$ under different granularity conditions, where $\text{concat}(d_{psg})$ is the concatenation of all relevant passages (d_{psg}) found in the relevant document d and $\text{avg}(g, d_{psg})$ is the arithmetic mean of relevance grades of the relevant passages.

3.2.2 Similarity Distance. Another traditionally used method in utilizing user feedback is the content similarity calculation. The basic idea is to rank documents based on their relevance to the user query and content similarity to presented or seen relevant documents as shown in Equation 8 [9, 36].

$$\text{score}(d,q) = \lambda\ \text{rel}(q,d) + (1-\lambda)\ \text{SimDist}(d, D_F). \tag{8}$$

In the following, we introduce different methods to calculate $\text{SimDist}(d, D_F)$ under different user feedback granularity conditions. In introducing these methods, we aim to align them with the

Table 2. Similarity distance calculation under different granularity conditions of user feedback.

Condition	$\text{SimDist}(d, D_F)$	$w_{d,d'}$	d'
TDB	$\arg\max_{d' \in D_F} w_{d,d'}\ \text{Cosine}(d, d')$	1	d'
TDG	$\arg\max_{d' \in D_F} w_{d,d'}\ \text{Cosine}(d, d')$	$\mathcal{G}(g(q, d'_{psg}))$	d'
TPB	$\arg\max_{d' \in D_F} w_{d,d'}\ \text{Cosine}(d, d')$	1	$\text{concat}(d'_{psg})$
TPG	$\arg\max_{d' \in D_F} w_{d,d'}\ \text{Cosine}(d, d')$	$\mathcal{G}(\text{avg}(g, d'_{psg}))$	$\text{concat}(d'_{psg})$
ADB	$\sum_{a \in \mathcal{A}_i} \arg\max_{d' \in D_a} w_{d,d'}\ \text{Cosine}(d, d')$	1	d'
ADG	$\sum_{a \in \mathcal{A}_i} \arg\max_{d' \in D_a} w_{d,d'}\ \text{Cosine}(d, d')$	$\mathcal{G}(g(a, d'))$	d'
APB	$\sum_{a \in \mathcal{A}_i} \arg\max_{d' \in D_a} w_{d,d'}\ \text{Cosine}(d, d')$	1	$\text{concat}(d'_{psg})$
APG	$\sum_{a \in \mathcal{A}_i} \arg\max_{d' \in D_a} w_{d,d'}\ \text{Cosine}(d, d')$	$\mathcal{G}(\text{avg}(g, d'_{psg}))$	$\text{concat}(d'_{psg})$

motivation of the approach, which is finding more relevant documents. Here the main focus is not to model novelty or to reduce redundancy.

Basic model. With binary, document-level and topic-level feedback, we calculate the similarity as follows:

$$\text{SimDist}(d, D_F) = \arg\max_{d' \in D_F}\ \text{Cosine}(d, d'). \tag{9}$$

where $\text{Cosine}(d, d')$ is calculated using the traditional TFIDF term weighting scheme.

Modeling graded relevance. Ideally, we want to rank a document that is similar to a highly relevant document higher than a document similar to a partially relevant document. To this end, if we have graded feedback, then we weight the similarity scores of an unseen document to various relevant documents as follows:

$$\text{SimDist}(d, D_F) = \arg\max_{d' \in D_F}\ \mathcal{G}(g(q, d')) \times\ \text{Cosine}(d, d'), \tag{10}$$

where $\mathcal{G}(g(q, d'))$ is calculated using Equation 6. Note this is a general form of the basic model with binary-scale feedback of Equation 9 where the similarity scores to relevant documents have equal weights.

Modeling aspects. If users provide aspect-based feedback, then we would like to rank documents that are similar to relevant documents covering many aspects higher than documents similar to documents covering a single aspect. In addition, as Equation 10 is general form for the basic model, we then can extend it further to support aspect-based similarity distance calculation as follows:

$$\text{SimDist}(d, D_F) = \sum_{a \in \mathcal{A}_i} \text{SimDist}(d, D_a) \tag{11}$$

Modeling passages. If users provide passage-based feedback, then passage texts allow us to make better similarity calculation as it helps to remove noise and non-relevant content from relevant documents. As a result, we concatenate all passages texts into one unit that represents the relevant document, and aggregate relevance grades of passages using the arithmetic mean.

To summarize, in the similarity distance reranking approach, we rerank documents using Equation 8, but the calculation of $\text{SimDist}(d, D_F)$ depends on the user feedback granularity condition. Table 2 lists the different conditions and how we calculate similarity.

3.2.3 Interactive search result diversification. A conventional method to manage exploration and exploitation of various aspects of an information need is search result diversification. There are two main approaches to tackle diversification: implicit and explicit. When users provide feedback at the aspect or subtopic level, then explicit diversification be a natural extension to utilize aspect-based feedback to cover multiple aspects of the information needs. As a result, we utilize the eXplicit Query Aspect Diversification xQuAD. The framework is a widely used and effective search result diversification method. This framework was the basis for many interactive diversification methods that exhibit competitive or superior performance to other dynamic approaches in many studies [52, 54, 84]. The framework also allows us to utilize graded relevance to model the coverage and novelty of documents. Following Moraes et al. [53, 54] interactive xQuAD implementation, at each iteration i, xQuAD greedily selects the document d that has the maximum gain in terms of the relevance to the user query and the diversity score for the different query aspects:

$$\text{xQuAD}(q, d, \mathcal{A}_i, \mathcal{D}) = 1 - \lambda \ \text{rel}(q, d) + \lambda \ \text{div}(q, d, \mathcal{A}_i, \mathcal{D}), \tag{12}$$

where $\mathcal{A}_i$ is the set of the query aspects that have relevant documents in the user feedback, $\text{rel}(q, d)$ denotes the normalized score of the initial ranker component, $\mathcal{D}$ is the set of documents that have been presented, or selected to present to the user , and $\text{div}(q, d, \mathcal{A}_i, \mathcal{D})$ is as follows:

$$\text{div}(q, d, \mathcal{A}_i, \mathcal{D}) = \sum_{a \in \mathcal{A}_i} P(a|q) \ P(d|q, a) \prod_{dj \in \mathcal{D}} (1 - P(d_j|q, a), \tag{13}$$

where $P(a|q)$ is uniform for all aspects, hence we assume the importance of aspects remains the same in a given session. $P(d|q, a)$ denotes the extent to which the document is relevant to the aspect of the query. The product $\prod_{d_j \in \mathcal{D}}(1 - P(d_j|q, a)$ models the novelty of the document d to the aspect a given documents ($\mathcal{D}$). In each iteration i, the diversity score might change based on the accumulated user feedback. We estimate $P(d|q, a)$ differently based on whether the document has received feedback or not as follows.

$$P(d|q, a) = \begin{cases} \mathcal{G}(g(a, d)) & if \ d \in D_F \\ P(d|a) & if \ d \notin D_F \end{cases} \tag{14}$$

The estimation $P(d|a)$ also varies based on the feedback unit granularity. When a user rates passages, then we follow the approach of Moraes et al. [53, 54] that estimates $P(d|a)$ using passages relevant to the aspect, a_{psg}, as follows:

$$P(d|a) = \arg\max_{p \in a_{psg}} P(d|p) \tag{15}$$

when feedback is provided at the document level, then we concatenate all relevant documents to build a single large document a_{ld} and estimate $P(d|q, a)$ as follows:

$$P(d|q, a) = P(d|a_{ld}) \tag{16}$$

Table 3 lists the different ways to calculate $P(d|q, a)$ when we have aspect-based feedback. In the case where feedback is provided at the topic level, then we do not perform the search using interactive xQuAD. As a result, our design is not fully balanced.

In this work, for interactive diversification, we utilized methods that directly estimate the relevance of documents to aspects of information needs, and their novelty using user feedback. Another approach to utilize user feedback in xQuAD is followed by Zhang et al. [84]. The essence of their approach is to use the fine-grained user feedback to generate a sub-query for each aspect, then plug these queries into xQuAD. However, as our focus is to utilize the approaches that follow different paradigms in ranking documents, we think using relevance feedback to generate the sub-queries being used by xQuAD might introduce a confounding variable in our analysis. The method will be a hybrid between the relevance feedback algorithm and xQuAD.

Table 3. The calculation of xQuAD under different granularity conditions of user relevance feedback.

Condition	$P(d\|q,a), d \in D_F$	$P(d\|q,a), d \notin D_F$
ADB	$P(d\|a_{ld})$	$P(d\|a_{ld})$
ADG	$\mathcal{G}(g(a,d))$	$P(d\|a_{ld})$
APB	$\arg\max_{p \in d_{psg}} P(d\|p)$	$\arg\max_{p \in a_{psg}} P(d\|p)$
APG	$\arg\max_{p \in a_{psg}} \mathcal{G}(g(a,p))$	$\arg\max_{p \in a_{psg}} P(d\|p)$

3.3 Evaluation Metrics

To measure the effects of the different factors, we measure the various systems in the Grid of Points systems using various metrics.

Across the years, researchers have proposed metrics that measure some or all of the search qualities the affect user satisfaction in dynamic search. In search result diversification [65], metrics such as subtopic recall st-rec, and α-nDCG [18] have been used to evaluate the ability of systems to retrieve relevant documents that satisfy diverse query aspects. Sloan and Wang [67] used these metrics to evaluate multi-page search systems. The session Discounted Cumulative Gain (sDCG) [38], that models graded relevance, and user effort is used to evaluate dynamic search methods in multi-query search sessions. Recently, the TREC Dynamic Domain (DD) track [79] adopted the Cube Test (CT) [47], and the Average Cube Test (ACT) [79] metrics, which model relevance, diversity, and effort, to evaluate dynamic search systems tackling multi-aspect information needs.

Recently, Albahem et al. [4] evaluated extensively the suitability of many of these metrics to evaluate multi-aspect dynamic search and suggested evaluating dynamic search methods with the normalized Cube Test (nCT) and the α-nDCG complex metrics that measure the overall performance of dynamic search. We also use precision (prec) and subtopic recall (st-rec) to represent simple metrics that assess individual search qualities such as topical relevance and diversity.

3.4 ANOVA Design

This paper uses an ANOVA framework to estimate the effects of various components of a dynamic search system. Before introducing the various models to estimate these effects, we describe below the overall ANOVA design that we followed in all of the models (described in Section 5).

ANOVA aims to explain the data (dependent variable) in terms of the experimental conditions (the model) and an error component (Equation 17) [25]:

$$Data = Model + Error. \tag{17}$$

In designing the "Model" component of the equation, we follow a factorial design. In particular, we use a repeated measures design where each topic's performance (the subject) is evaluated under each system.

Generally, ANOVA makes some assumptions, whose fulfillment increases the confidence of its results:

- Normal population distribution: The dependent variable measurements (evaluation metric scores) for each factor level (component instantiations) are normally distributed.
- Equal variance (homogeneity of variance): the variance of each factor level is (near) equal.
- The dependent variable is an unbounded continuous variable measurement on an interval scale.
- The independence of the cases. That is, the choices of the factor levels are independent of each other.

Generally, evaluation metrics in information retrieval violate the first two assumptions. We also observed similar trends for the metrics we are using in this study. Nevertheless, several researchers have found that ANOVA analysis is robust to these violations. A general observation by Ito [32] is that the F-test is not sensitive to violations of normality. In information retrieval, several researchers have found the ANOVA framework to be robust to these violations as well [14, 70]. Nevertheless, we applied *sqrt* transformation to address the normality violation and utilized MacKinnon and White's (1985) heteroskedasticity robust standard errors.

The metrics used in this paper also violate the third assumption. To address this, we applied the recently proposed transformation of Ferrante et al. [24] that uses the ranking of systems according to the evaluation metric to overcome this violation.

Finally, with respect to the independence of cases, we use a repeated measures study by design. Thus this assumption is not satisfied. Nevertheless, our components and topics are not dependent on each other. Our analysis uses thousands of observations obtained from evaluating tens of topics in more than 100 search systems; thus, we expect the impact of this violation to be limited.

4 EXPERIMENTAL SETUP

In the previous sections, we discussed the system components and how we built the Grid of Point systems. In this section, we describe the evaluation setup within the ANOVA-based performance analysis. We describe the evaluation collections, prepossessing steps, and evaluation metrics used.

Test collections. To perform analysis with granularity conditions, we need a collection that facilitates simulation of various granularity conditions of user feedback. To that end, we utilize the TREC Dynamic Domain (DD) collections from TREC DD 2016 [80] and TREC DD 2017 [81]. There are two domains in TREC DD 2016: Ebola, and Polar, with 26 and 27 queries for each domain respectively. The Ebola dataset mainly focuses on web pages from news, government, and not-profit organization sites discussing the Ebola epidemic outbreak in Africa. The queries are also related to the Ebola epidemic. In the Polar domain, the data is related to climate change scientific research crawled from the Web. In TREC DD 2017, there were two domains: Ebola and News. The Ebola domain was reused from TREC DD 2016. The news domain was based on the New York Times collection where the topics are 60 analytical and intellectual search tasks. In our analysis, for TREC DD 2017, we only use the New York Times collection.

Data prepossessing and indexing. In both collections, we will use the ground truth to simulate user feedback. However, the track actual relevance judgments are subtopic, passage, and graded relevance judgments. Therefore, we need to transform them from fine to more coarse granularities. To achieve that we perform the following:

- Passage to document conversion: Any document is relevant if it contains any relevant passage. To convert graded relevance at the passage level to the document level, we aggregate relevance grades of the passages using the arithmetic mean.
- Aspect to topic : A document is relevant to the query if it contains any relevant information to any aspect.

We created a separate index for each domain collection and ran the search systems on these isolated indices. In TREC DD 2016, we first extracted the main content of the web pages using the Boilerpipe extraction library, then indexed the content.[5] In TREC DD 2017, we concatenated the title and body content of the news articles. We normalized texts with the English Porter stemmer, lower casing, and stop-word[6] removal in all collections. We applied the same steps when processing

[5]We use a fork version of the library found at https://github.com/aalbahem/boilerpipe/tree/v.2.1
[6]We used the default Lucene English stop-word list supplied in version 8.0.0.

the user queries or the user feedback. In particular, we further cleaned any text HTML entities or tags found in the user feedback.

In the search process, the initial retrieval methods were performed using the Lucene library.[7] For *RM3*, we use the default implementation of the Anserini toolkit.[8] We then implement all other methods discussed in Section 3.2.

5 ANOVA MODELS AND ANALYSIS

In this section, we gradually introduce a series of ANOVA models that allow us to estimate the effects of factors outlined in Section 3.1.

5.1 Effects of the Topic and System Factors

In our first ANOVA model, we analyze search effectiveness based on the classical information retrieval model that allows us to answer our first research question which is **"What are the effects of the topics and system factors on the performance of dynamic search?"** The ANOVA model is as follows:

$$\text{Effectivness}_{ij} = \mu_{..} + \text{topic}_i + \text{system}_j + \varepsilon_{ij}, \tag{18}$$

where $\mu..$ is the grand mean of all observations, topic_i represents the i-th topic's main effect, system_j is the j-th system's main effect, and ε_{ij} is the error committed by the model in predicting the effectiveness metric score of the i-th topic for the j-th system. For TREC DD 2016, we estimated the model using 100 systems and 53 topics. For TREC DD 2017, we estimated the model using 100 systems and 60 topics.

Table 4 reports the estimated effect size ω^2 [26] of the topic and system factors according to many metrics based on the TREC DD 2016 and TREC DD 2017 collections. In each cell, we indicate whether the effect of a factor is statistically significant with a p-value < 0.05, and the category of the effect size such as small, medium, and large using the categorization followed in [25–27].[9]

From the table, we make several observations. First, both the topic and system factors have statistically significant effects on the performance of search systems. The significance is consistent across iterations, metrics, and collections. In addition, the effect of the topic is larger than that of the system factor, which is the case also in ad hoc search [25, 27]. The system factor has a statistically significant effect and a large or medium effect size in general. The effect is consistent across iterations, collections, and metrics except for subtopic recall (st-rec) on TREC DD 2017.

5.2 Effects of System Components

In the previous section, the ANOVA model allowed us to estimate the effects of the topics and the system factors. In this section, we introduce an ANOVA model that allows us to answer the second research question (**RQ 2**). That is we aim to estimate the effects of the system-wise components. As discussed in Section 3.1, each dynamic search system is composed of instantiations of three components: Initial Ranker (IR), Dynamic Reranker (DR), and User Feedback Granularity (UFG). The results above also showed that the system factor generally has large and statistically significant effects on search performance. To investigate what contributes to the significant variance of the system factor, we decompose the system factor in Equation 18 into factors that represent

[7] We use version 8.0.0

[8] We did not perform the document model clipping step. In the general implementation of RM3, we usually clip the final relevance model induced from all relevant documents, that is keeping the top n weighted terms of the model and discard the rest. However, in the Anserini toolkit, each document model is also clipped to 10-terms, a step that is not widely adopted or reported in the literature. As a result, we only clipped the model after inducing it from the relevant documents

[9] The complete ANOVA tables for each metric with various ANOVA measures such as Sum of Squares, Mean Squares, F-statistic, P-value and effect size ω^2 are all reported in the Appendix.

Table 4. Effects of factors of Equation 18, on TREC Dynamic Domain collections. Each cell reports the effect size ω^2 for each factor, and within parentheses, the p-value for the effect. Small effects are in light blue; medium effects are in blue; and large effects are in dark blue.

Iter	Effects	prec	st-rec	α-nDCG	nCT
		TREC DD 2016			
1	Topic	0.5910 (0.00)	0.6088 (0.00)	0.6130 (0.00)	0.6131 (0.00)
1	System	0.2140 (0.00)	0.1683 (0.00)	0.2370 (0.00)	0.1788 (0.00)
2	Topic	0.6071 (0.00)	0.6201 (0.00)	0.6430 (0.00)	0.6177 (0.00)
2	System	0.2019 (0.00)	0.1402 (0.00)	0.2125 (0.00)	0.1356 (0.00)
5	Topic	0.7169 (0.00)	0.5706 (0.00)	0.6654 (0.00)	0.6281 (0.00)
5	System	0.1698 (0.00)	0.1253 (0.00)	0.2424 (0.00)	0.1129 (0.00)
10	Topic	0.7997 (0.00)	0.5976 (0.00)	0.6719 (0.00)	0.7378 (0.00)
10	System	0.1179 (0.00)	0.0761 (0.00)	0.2481 (0.00)	0.0986 (0.00)
		TREC DD 2017			
1	Topic	0.7013 (0.00)	0.6761 (0.00)	0.7176 (0.00)	0.8056 (0.00)
1	System	0.3232 (0.00)	0.1494 (0.00)	0.2595 (0.00)	0.2426 (0.00)
2	Topic	0.7337 (0.00)	0.6002 (0.00)	0.6999 (0.00)	0.8269 (0.00)
2	System	0.2158 (0.00)	0.0865 (0.00)	0.2161 (0.00)	0.1571 (0.00)
5	Topic	0.7403 (0.00)	0.5054 (0.00)	0.7039 (0.00)	0.8406 (0.00)
5	System	0.1623 (0.00)	0.0581 (0.00)	0.2388 (0.00)	0.1322 (0.00)
10	Topic	0.7803 (0.00)	0.4499 (0.00)	0.6954 (0.00)	0.8676 (0.00)
10	System	0.1508 (0.00)	0.0245 (0.00)	0.2484 (0.00)	0.1236 (0.00)

the components and their interaction effects. We model the main and interaction effects of the system-wise components as follows:

$$\begin{aligned}\text{Effectiveness}_{ijkl} = \mu_{....} + \overbrace{\text{topic}_i + \text{IR}_j + \text{DR}_k + \text{UFG}_l}^{\text{Main effects}} \\ + \overbrace{\text{IR}_j \times \text{DR}_k}^{\text{Interaction effect}} + \overbrace{\text{IR}_j \times \text{UFG}_l}^{\text{Interaction effect}} + \overbrace{\text{DR}_k \times \text{UFG}_l}^{\text{Interaction effect}} + \overbrace{\text{IR}_j \times \text{DR}_k \times \text{UFG}_l}^{\text{Interaction effect}} \\ + \varepsilon_{ijkl},\end{aligned} \tag{19}$$

where $\mu_{....}$ is grand mean of all observations, topic_i represents the topic's main effect, and IR_j is the effect of the initial ranker, DR_k is the effect of the dynamic reranker, UFG_l is the effect of the user feedback granularity condition, and $\times$ indicates an interaction effect. We estimated the model using five initial rankers, three dynamic rerankers and eight conditions of user feedback granularity — see Section 3 for the summary of these components. In the following, we first describe the main effects, then the interaction effects. In Section 5.3, we discuss the relation between the effect size and the length of the search session in more details.

5.2.1 Main effects. Table 5 presents the main effects of the individual components of a dynamic search system based on the TREC DD 2016 and TREC DD 2017 collections.[10] We omitted the effects at iteration 1 because the initial ranker was the only factor that has effect, its effect is identical to the effect of the system factor reported in Table 4.[11] Overall all, comparing Table 5 to Table 4, we can see that the system variance is largely due to the initial ranker factor (IR) as it has the most prominent effect size.[12] The effect is statistically significant with a noticeable effect size, generally large or medium. The effects of the dynamic reranker and user feedback granularity factors differ across iterations, metrics, and collections. In the following, we discuss in details the effects of the individual factors.

Effects of the initial rankers. The statistically significant effect of the initial ranker on performance, with a large effect size, might be due to the different ways in which initial rankers could impact the search process.

The first way is that initial rankers determine inputs of dynamic rerankers. The performance of a dynamic search system is directly affected by the initial ranked list produced by the initial ranker. Moraes et al. [53] showed through prototyping initial rankers, with various levels of precision and recall, that dynamic search systems need to have an initial ranker with high precision to perform well in early iterations and a high recall initial ranker to perform well in late iterations. Our analysis aligns with their findings. We found that the initial ranker factor has a large or medium and significant effect at early and late iterations.

The second way initial rankers might impact on performance is to retrieve relevant documents as early as possible. In the used dynamic rerankers discussed in Section 3.2, we have utilized only positive feedback to rerank documents. Therefore, if no explicit positive feedback is found, we present the next batch of documents from the initial ranked list of documents produced by the initial reranker. As a result, for topics with zero precision at the first iteration, the initial rankers are the only source of variance. Figure 3 reports the precision of the five initial rankers we used in our experiments: BM25 [69], Divergence From Randomness (DFR) [6], Language Model (LM) [56], relevance model [40] with Pseudo Relevance Feedback (PRF) and the Vector Space Model with the TFIDF weighting. In the figure, we could see the performance of the initial rankers at the first iteration varies greatly. For many topics, some methods are performing poorly with zero precision, especially the TFIDF ranker.

The third way initial rankers could affect the performance is the number of relevant documents they retrieve in the first iteration. Prior findings from the TREC Relevance Feedback track [11] indicate that interactive search methods tend to improve with more relevant documents. As shown in Figure 3, the initial rankers differ in the number of relevant documents retrieved, hence they vary in their precision scores.

Effects of the dynamic rerankers. Concerning the effects of the dynamic reranker component, we notice a statistically significant difference between the different dynamic rerankers in their ability to retrieve relevant documents, measured by precision (prec). The effect size is more noticeable in TREC DD 2017 than TREC DD 2016. The effect size is also generally larger than that of TREC DD 2016. The larger effect size in TREC DD 2017 might be due to the ability of the initial rankers to retrieve relevant documents at the first iteration. As discussed previously, many initial rankers perform better in TREC DD 2017 as compared to their performances in TREC DD 2016. Therefore, with

[10]Following our approach in the previous section, we report only synthesized data of effect size, indicating statistical significance and the category of the effect size. The complete ANOVA measures are provided in the Appendix.

[11]Recall, the user feedback is available only from the second iteration, hence the dynamic reranker will start to affect performance from the second iteration.

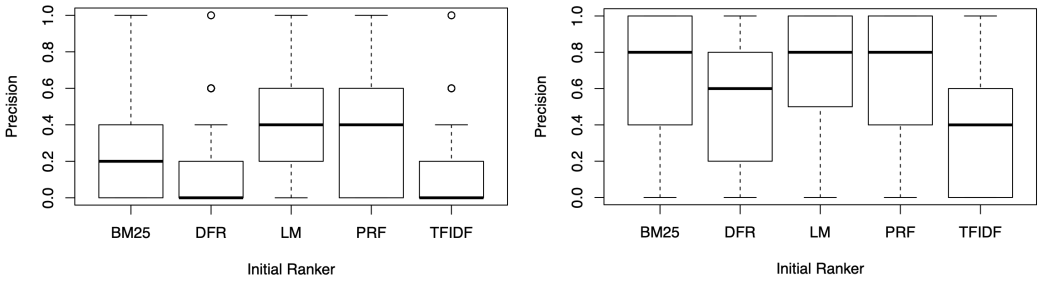
[12]It also explains most of the system variance as indicated by the Sum Squared measure reported in the Appendix.

Table 5. Effects of factors of Equation 19, on TREC DD collections. Each cell reports the effect size ω^2 for each factor, and within parentheses, the p-value for the effect. Statistically insignificant effects, p-value $\geq$ 0.05, are in gray; trivial effects are in white; small effects are in light blue; medium effects are in blue; and large effects are in dark blue.

Iter	Effects	prec	st-rec	α-nDCG	nCT
		TREC DD 2016			
2	IR	0.2065 (0.00)	0.1513 (0.00)	0.2230 (0.00)	0.1455 (0.00)
2	DR	-0.0002 (0.81)	0.0000 (0.32)	-0.0002 (0.76)	0.0001 (0.24)
2	UFG	0.0058 (0.00)	-0.0008 (0.98)	-0.0002 (0.58)	-0.0001 (0.47)
2	IR x DR	-0.0007 (0.87)	-0.0010 (0.98)	-0.0011 (1.00)	-0.0010 (0.99)
2	IR x UFG	-0.0040 (1.00)	-0.0043 (1.00)	-0.0043 (1.00)	-0.0041 (1.00)
2	DR x UFG	-0.0012 (0.95)	-0.0014 (0.98)	-0.0020 (1.00)	-0.0019 (1.00)
2	IR x DR x UFG	-0.0059 (1.00)	-0.0072 (1.00)	-0.0066 (1.00)	-0.0074 (1.00)
5	IR	0.1553 (0.00)	0.1219 (0.00)	0.2490 (0.00)	0.1089 (0.00)
5	DR	0.0029 (0.00)	0.0085 (0.00)	0.0025 (0.00)	0.0087 (0.00)
5	UFG	0.0199 (0.00)	0.0049 (0.00)	0.0019 (0.01)	0.0059 (0.00)
5	IR x DR	0.0002 (0.30)	-0.0004 (0.73)	-0.0009 (0.97)	-0.0007 (0.88)
5	IR x UFG	-0.0043 (1.00)	-0.0036 (1.00)	-0.0043 (1.00)	-0.0037 (1.00)
5	DR x UFG	-0.0001 (0.49)	-0.0013 (0.98)	-0.0017 (1.00)	-0.0010 (0.92)
5	IR x DR x UFG	-0.0058 (1.00)	-0.0046 (1.00)	-0.0072 (1.00)	-0.0063 (1.00)
10	IR	0.0897 (0.00)	0.0656 (0.00)	0.2522 (0.00)	0.0832 (0.00)
10	DR	0.0059 (0.00)	0.0191 (0.00)	0.0052 (0.00)	0.0189 (0.00)
10	UFG	0.0266 (0.00)	0.0021 (0.00)	0.0033 (0.00)	0.0078 (0.00)
10	IR x DR	0.0000 (0.42)	-0.0008 (0.93)	-0.0009 (0.96)	-0.0008 (0.95)
10	IR x UFG	-0.0043 (1.00)	-0.0042 (1.00)	-0.0043 (1.00)	-0.0042 (1.00)
10	DR x UFG	-0.0011 (0.93)	-0.0011 (0.93)	-0.0016 (1.00)	-0.0013 (0.98)
10	IR x DR x UFG	-0.0068 (1.00)	-0.0044 (1.00)	-0.0067 (1.00)	-0.0052 (1.00)
		TREC DD 2017			
2	IR	0.2053 (0.00)	0.0889 (0.00)	0.2247 (0.00)	0.1586 (0.00)
2	DR	0.0078 (0.00)	0.0061 (0.00)	0.0011 (0.01)	0.0071 (0.00)
2	UFG	0.0014 (0.02)	-0.0005 (0.84)	-0.0007 (0.96)	-0.0005 (0.87)
2	IR x DR	0.0007 (0.12)	-0.0007 (0.91)	-0.0010 (1.00)	-0.0007 (0.96)
2	IR x UFG	-0.0033 (1.00)	-0.0034 (1.00)	-0.0038 (1.00)	-0.0038 (1.00)
2	DR x UFG	-0.0002 (0.56)	-0.0016 (1.00)	-0.0017 (1.00)	-0.0015 (1.00)
2	IR x DR x UFG	-0.0061 (1.00)	-0.0060 (1.00)	-0.0068 (1.00)	-0.0056 (1.00)
5	IR	0.1220 (0.00)	0.0590 (0.00)	0.2470 (0.00)	0.1331 (0.00)
5	DR	0.0217 (0.00)	0.0034 (0.00)	0.0009 (0.01)	0.0079 (0.00)
5	UFG	0.0039 (0.00)	-0.0008 (0.98)	-0.0007 (0.96)	-0.0006 (0.90)
5	IR x DR	0.0035 (0.00)	0.0006 (0.14)	-0.0009 (1.00)	-0.0009 (0.99)
5	IR x UFG	-0.0029 (1.00)	-0.0032 (1.00)	-0.0038 (1.00)	-0.0037 (1.00)
5	DR x UFG	-0.0002 (0.58)	-0.0004 (0.67)	-0.0003 (0.60)	-0.0006 (0.80)
5	IR x DR x UFG	-0.0042 (1.00)	-0.0050 (1.00)	-0.0056 (1.00)	-0.0055 (1.00)
10	IR	0.0998 (0.00)	0.0224 (0.00)	0.2564 (0.00)	0.1185 (0.00)
10	DR	0.0228 (0.00)	0.0042 (0.00)	0.0009 (0.01)	0.0091 (0.00)
10	UFG	0.0077 (0.00)	-0.0007 (0.95)	-0.0006 (0.93)	0.0000 (0.39)
10	IR x DR	0.0038 (0.00)	0.0010 (0.06)	-0.0009 (0.99)	-0.0006 (0.90)
10	IR x UFG	-0.0032 (1.00)	-0.0028 (1.00)	-0.0038 (1.00)	-0.0036 (1.00)
10	DR x UFG	-0.0004 (0.66)	-0.0004 (0.69)	-0.0015 (1.00)	-0.0002 (0.54)
10	IR x DR x UFG	-0.0054 (1.00)	-0.0028 (0.98)	-0.0049 (1.00)	-0.0054 (1.00)

relevant documents, the dynamic reranker could start to affect performance early and propagate that effect to later iterations.

Concerning the effects of dynamic rerankers on discovering new aspects of the user queries as measured by subtopic recall (st-rec), we notice there is a statistically significant difference between



(a) TREC DD 2016 (b) TREC DD 2017

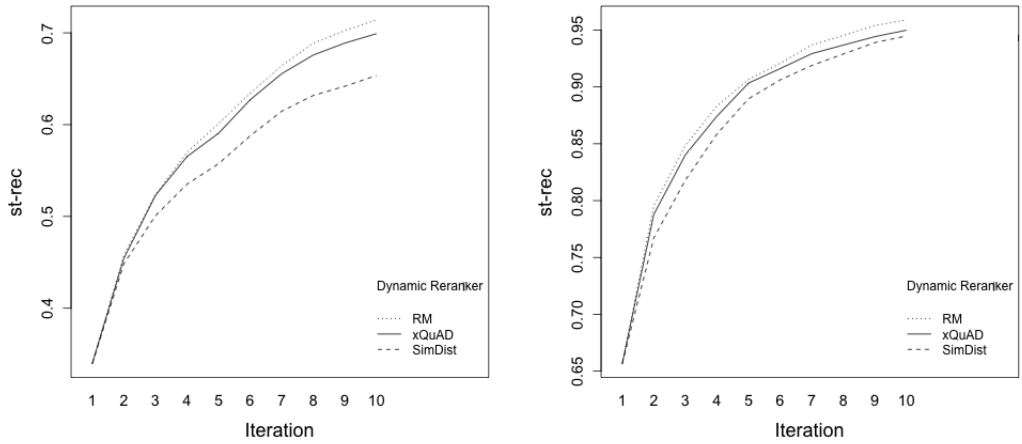
Fig. 3. Precision (prec) at first iteration

the dynamic rerankers across iterations. This might be surprising, given that dynamic rerankers discussed in Section 3.2 do not have a specific component to explore new subtopics. They operate on the user feedback; hence we would expect to see more of the same aspects discovered. The significant difference might be due to the query expansion used in some dynamic rerankers such as the Relevance Model (RM). As documents could discuss multiple subtopics, the dynamic rerankers might be able to implicitly pull documents relevant to unseen subtopics as the expanded or new query representation might contain terms related to unseen subtopics. We could see in Figure 4 that the Relevance Model (RM) reranker, a query expansion method, performs consistently better than the other rerankers. In addition, the subtopic recall at the first iteration on TREC DD 2017, which relies on the initial ranker only, is higher than that of TREC DD 2016. Therefore, the marginal subtopic recall scores added might not be substantial if the initial rankers have high subtopic recall as is the case with TREC DD 2017.

In terms of the overall system performance, the choice of the dynamic reranker does with statistical significance affect performance. However, the effect size might differ based on how the overall performance is measured. Using α-nDCG, which is a binary relevance-, and diversity-based metric, we see the effect size is trivial. When measuring performance using the nCT metric, which models graded-relevance, diversity, and effort, we see the dynamic reranker component has a larger effect as compared to α-nDCG. The statistically significant difference in the overall performance, as measured by nCT, indicates that the choice of dynamic reranker affects the ability of interactive search systems to find documents with higher gains.

Effects of the user feedback granularity. We can make a few observations about the effect of user feedback granularity. First, the granularity of user feedback might affect the ability of search systems to retrieve relevant documents. The effects are larger in TREC DD 2016 as compared to the effects in TREC DD 2017. Second, in terms of the effects of UFG in other qualities of search performance, we see that it has significant effects in TREC DD 2016; The statistically significant effects differ across iterations. However, in TREC DD 2017, it has no significant effects. We suspect this might be due to the performance of the systems in precision. That is when there is a lower performance in precision, it is helpful to have more granularity as it reduces the noise.

5.2.2 Effects of interaction between system components. Table 5 shows the effects of the interaction between the system components for the TREC DD 2016 and TREC DD 2017 collections. In TREC DD 2016, we could see that no interaction has a statistically significant impact on search performance. This is consistent across iterations and metrics. In TREC DD 2017, we observe similar trends



(a) TREC DD 2016 (b) TREC DD 2017

Fig. 4. Subtopic recall (st-rec) of various dynamic rerankers as the search session progresses

except for the precision metric. The **IR x DR** interaction has a statistically significant effect on the performance in mid to late iterations.

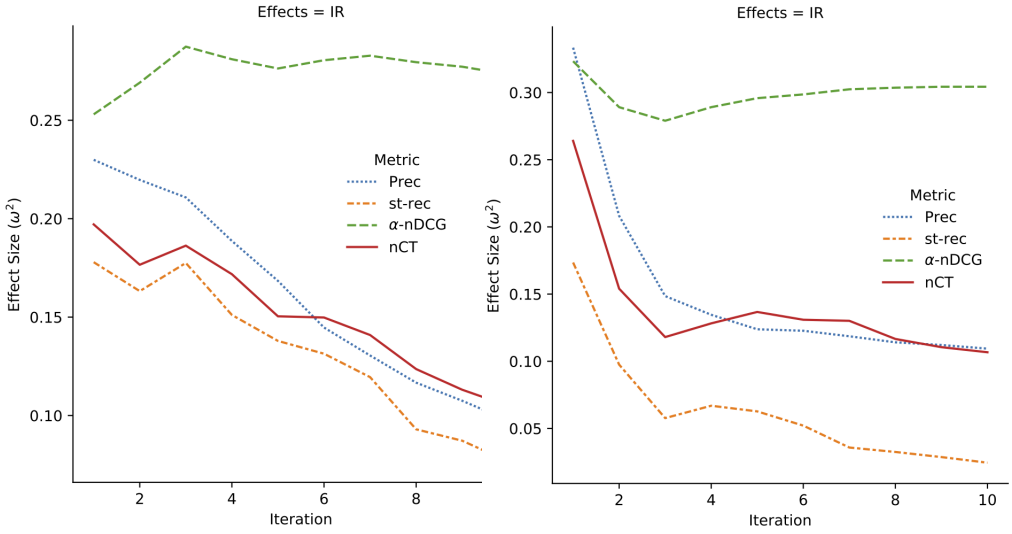
5.3 Effects of the System Components and the Search Session Length

A general trend we observed in Table 5 is the effects of the system components on performance differ at different stages of the search session. In this section, we discuss this and seek to answer the third research sub-question **RQ 3**.

Figure 5 depicts the effect of the Initial Ranker (IR) component on performance as users perform multiple search iterations. We could see that the effect size of the initial ranker factor tends to decrease as we progress through the search session except for α-nDCG where it fluctuates but still has a large effect.

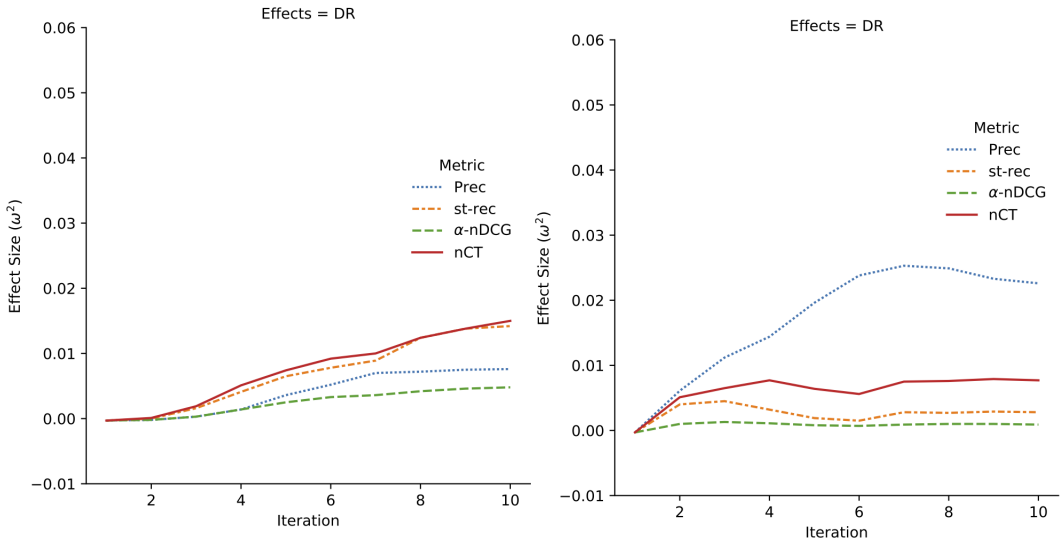
Concerning the effect of dynamic derankers (DR) (Figure 6), there are trends that could be observed. In terms of precision, in both collections, the effect size gradually increases from early to late iterations, and then it tends to decrease in the last two iterations. In TREC DD 2016, we notice that it starts trivial and gradually increases as we progress in the search session. Concerning the effects of dynamic rerankers on discovering new aspects of the user queries as measured by subtopic recall (st-rec), the effect size gradually increases as we move across iterations, we notice the effect size is generally larger than the effect size for precision in TREC DD 2016, and vice versa in TREC DD 2017.

Concerning the effect of User Feedback Granularity (UFG), from Figure 7, we can make a few observations about its effects. In both collections, the effect size gradually increases from early to late iterations. Nevertheless, the effects are larger in TREC DD 2016 as compared to the effects in TREC DD 2017. Second, in terms of the effects of UFG in other qualities of search performance, we see that it has a significant effect in TREC DD 2016; where it has a statistically significant effect from iteration 4 onward. However, in TREC DD 2017, it has no significant effects.



(a) TREC DD 2016 (b) TREC DD 2017

Fig. 5. Effect size of the Initial Ranker (IR) factor as the search session progress

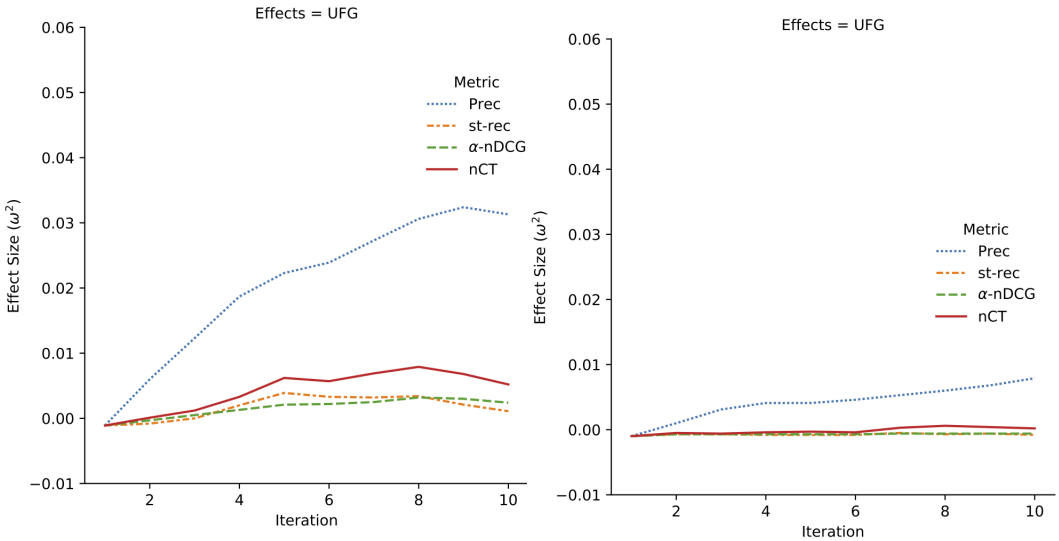


(a) TREC DD 2016 (b) TREC DD 2017

Fig. 6. Effect size of the Dynamic Reranker (DR) factor as the search session progress

5.4 Further Analysis

5.4.1 Effects of the initial ranker on precision. To investigate whether the precision at the first iteration affects search performance at different iterations, we model the precision of the first iteration in our ANOVA analysis. In particular, we perform the analysis using two factors: topic and



(a) TREC DD 2016

(b) TREC DD 2017

Fig. 7. Effect size of the User Feedback Granularity (UFG) factor as the search session progress

search start, shown in Equation 20.

$$\text{Effectivness}_{ij} = \mu_{..} + \text{topic}_i + \text{Search Start}_j + \varepsilon_{ij.} \tag{20}$$

The Search Start factor borrows the concept of "cold start" [1] in recommendation systems where the systems need to recommend items to users with no prior history. Similarly, we classify each system based on its initial ranker performance at the first iteration: cold start and warm start. A cold start means the initial ranker does not have relevant documents in the first iteration. A warm start means the initial ranker has at least a relevant document in the first iteration. Note that with the Search Start factor, the ANOVA analysis is not balanced since not all topics or systems will have an equal number of cold or warm starts.

Tables 6 and 7 show the new ANOVA analysis; for presentation purposes, we do not report the values for the topic factor as it is the same as the previous models. We can see from tables that the Search Start factor has a statistically significant effect across metrics, iterations, and collections. Furthermore, the effect size of the Search Start factor decreases as we progress through the search session. This is expected as initial rankers might have relevant documents in iteration 2, 3 or 4, hence the dynamic reranker might start to affect performance.

5.4.2 Effects of user feedback granularity dimensions. In the previous section, we found that user feedback granularity has a statistically significant impact on performance. Recall that in Section 3, we discuss three granularity dimensions: topic level granularity (topic or aspect), unit granularity (document or passage), relevance scale granularity (binary or graded). Nevertheless, the question is how do these dimensions contribute to the effect? To answer this research question, we construct

Table 6. Effects of factors of Equation 20, on TREC DD 2016 collection. Redundant factors in previous models are omitted for brevity.

Iter	Effects	prec	st-rec	α-nDCG	nCT
2	Search Start	0.4364 (0.00)	0.3206 (0.00)	0.4590 (0.00)	0.3506 (0.00)
3	Search Start	0.3470 (0.00)	0.2220 (0.00)	0.4165 (0.00)	0.2637 (0.00)
4	Search Start	0.2803 (0.00)	0.1643 (0.00)	0.3888 (0.00)	0.2189 (0.00)
5	Search Start	0.2367 (0.00)	0.1489 (0.00)	0.3813 (0.00)	0.1932 (0.00)
6	Search Start	0.2009 (0.00)	0.1032 (0.00)	0.3677 (0.00)	0.1620 (0.00)
7	Search Start	0.1743 (0.00)	0.0728 (0.00)	0.3615 (0.00)	0.1279 (0.00)
8	Search Start	0.1520 (0.00)	0.0552 (0.00)	0.3554 (0.00)	0.1094 (0.00)
9	Search Start	0.1364 (0.00)	0.0545 (0.00)	0.3539 (0.00)	0.1040 (0.00)
10	Search Start	0.1215 (0.00)	0.0457 (0.00)	0.3530 (0.00)	0.0978 (0.00)

Table 7. Interaction effects of factors of Equation 20, on TREC DD 2017 collection. Redundant factors in previous models are omitted for brevity.

Iter	Effects	prec	st-rec	α-nDCG	nCT
1	Search Start	0.3149 (0.00)	0.4413 (0.00)	0.2966 (0.00)	0.2378 (0.00)
2	Search Start	0.2920 (0.00)	0.4102 (0.00)	0.3648 (0.00)	0.3188 (0.00)
3	Search Start	0.2387 (0.00)	0.2805 (0.00)	0.3503 (0.00)	0.2606 (0.00)
4	Search Start	0.1970 (0.00)	0.2336 (0.00)	0.3350 (0.00)	0.2281 (0.00)
5	Search Start	0.1520 (0.00)	0.1379 (0.00)	0.3109 (0.00)	0.1653 (0.00)
6	Search Start	0.1230 (0.00)	0.1157 (0.00)	0.2992 (0.00)	0.1307 (0.00)
7	Search Start	0.1048 (0.00)	0.0704 (0.00)	0.2931 (0.00)	0.1056 (0.00)
8	Search Start	0.0928 (0.00)	0.0608 (0.00)	0.2873 (0.00)	0.0779 (0.00)
9	Search Start	0.0871 (0.00)	0.0423 (0.00)	0.2817 (0.00)	0.0610 (0.00)
10	Search Start	0.0825 (0.00)	0.0364 (0.00)	0.2784 (0.00)	0.0522 (0.00)

an ANOVA model based on the topic and these dimensions as follows:

$$\begin{aligned}\text{Effectiveness}_{ijkl} = \mu_{....} + \overbrace{\text{topic}_i + \text{LG}_j + \text{UG}_k + \text{RSG}_l}^{\text{Main effects}} \\ + \overbrace{\text{LG}_j \times \text{UG}_k}^{\text{Interaction effect}} + \overbrace{\text{LG}_j \times \text{RSG}_l}^{\text{Interaction effect}} + \overbrace{\text{UG}_k \times \text{RSG}_l}^{\text{Interaction effect}} + \overbrace{\text{LG}_j \times \text{UG}_k \times \text{RSG}_l}^{\text{Interaction effect}} \\ + \varepsilon_{ijkl},\end{aligned} \tag{21}$$

where is $\mu_{....}$ is the grand mean, topic_i is the topic effect, LG_j is topic level granularity effect, UG_k is the unit granularity effect, and RSG_l is the relevance scale granularity effect.[13]

Table 8 shows the effect size for the various dimensions of the user feedback granularity and their interactions. From the table, we could see that the unit granularity dimension has a statistically significant effect on the ability of systems to retrieve relevant documents (see the prec column). The effect tends to increase from early to late iterations. In the TREC DD 2016 collection, the effect size is larger. The UG factor also has a statistically significant impact on all other search qualities metrics. The LG and RSG factors occasionally have statistically significance, scattered across iterations in both collections. In particular, LG seems to affect performance at the early iterations, but RSG tends to affect performance at late iterations.

In terms of the interaction between the dimensions, only the interaction between all dimensions is statistically significant, at late iterations in TREC DD 2017. In addition, the effect size is trivial.

5.5 Power Analysis

In our experiments, we utilized existing resources to estimate effects. As reported in the previous sections, we observed a mix of different behaviors of the metrics and components. As a result, in this section, we investigate the power of our experiments by performing a post-hoc statistical power of the probability that the (ANOVA) test will correctly reject the null hypothesis. Using the ω^2 we reported in the previous sections, we utilize G*Power[14] to calculate the power of the experiments. For the topic and system factors, in both collections, we found the power to be high; thus, we omit reporting them here. In Table 9, we report the power analysis for the different factors. From the table, we see that, for the initial ranker, we generally have a high power, which aligns with its consistent larger effect sizes. For the other components, there is a high probability that we might have committed a Type II error. In future work, we plan to investigate designing experiment setups with more queries.

6 CONCLUSION AND FUTURE WORK

Dynamic search methods are complex information retrieval systems. Multiple factors, among them the system components, have been found to affect their performance. However, research has mainly followed a black-box evaluation that obscures estimating the effects of various factors on performance. This study sought to address this research gap and answer the following question *What are the effects of the dynamic search components on performance?*

To answer this question, we conducted an ANOVA-based analysis of various components: topics, the initial ranker, the dynamic reranker, and user feedback granularity. We also implemented and extended many dynamic reranking approaches to utilize various user feedback granularity conditions. In summary, we built 100 systems and evaluated their performance in the TREC Dynamic Domain collections using many metrics. We further divided the question above into three sub-research questions. Our findings answer these questions as follows:

***RQ 1**: What are the effects of the topics and system factors on dynamic search performance?* Both the topics and the system factors generally have large and statistically significant effects. The topics factor has larger effects than the system factor.

***RQ 2**: What are the effects of the dynamic search components such as the initial ranker, the dynamic reranker, and user feedback granularity?* The choice of instantiating any of the components has

[13] Ideally, we would include the initial reranker and dynamic reranker factors in the model as well. However, this would make the model more complex and hard to interpret. As a result, we focus on the topic and user feedback granularity dimensions.

[14] https://www.psychologie.hhu.de/arbeitsgruppen/allgemeine-psychologie-und-arbeitspsychologie/gpower

Table 8. Effects of factors of Equation 21, on TREC DD 2016 and 2017 collections. Each cell reports the effect size ω^2 for each factor, and within parentheses, the p-value for the effect. Statistically insignificant effects, p-value ≥ 0.05, are in gray; trivial effects are in white; small effects are in light blue; medium effects are in blue; and large effects are in dark blue.

Iter	Effects	prec	st-rec	α-nDCG	nCT
		TREC DD 2016			
2	LG	0.0016 (0.00)	-0.0001 (0.62)	0.0000 (0.29)	0.0001 (0.25)
2	UG	0.0043 (0.00)	0.0000 (0.30)	0.0003 (0.10)	0.0006 (0.03)
2	RSG	-0.0001 (0.80)	-0.0001 (0.80)	-0.0002 (0.88)	-0.0001 (0.83)
2	LG x UG	-0.0000 (0.38)	-0.0002 (0.89)	-0.0001 (0.81)	-0.0001 (0.73)
2	LG x RSG	-0.0002 (0.91)	-0.0002 (0.92)	-0.0002 (0.89)	-0.0002 (0.90)
2	UG x RSG	-0.0001 (0.80)	-0.0002 (0.91)	-0.0002 (0.93)	-0.0002 (0.88)
2	LG x UG x RSG	-0.0002 (0.85)	-0.0002 (0.97)	-0.0002 (0.96)	-0.0002 (0.93)
5	LG	0.0003 (0.08)	-0.0001 (0.49)	-0.0000 (0.32)	0.0001 (0.21)
5	UG	0.0189 (0.00)	0.0034 (0.00)	0.0018 (0.00)	0.0050 (0.00)
5	RSG	-0.0000 (0.33)	-0.0000 (0.38)	-0.0001 (0.68)	-0.0001 (0.47)
5	LG x UG	-0.0001 (0.73)	-0.0001 (0.66)	-0.0001 (0.77)	-0.0001 (0.59)
5	LG x RSG	-0.0001 (0.56)	-0.0001 (0.53)	-0.0001 (0.71)	-0.0001 (0.51)
5	UG x RSG	-0.0001 (0.58)	-0.0002 (0.92)	-0.0002 (0.98)	-0.0002 (0.91)
5	LG x UG x RSG	-0.0001 (0.46)	-0.0002 (0.96)	-0.0002 (0.88)	-0.0002 (0.97)
10	LG	-0.0001 (0.77)	-0.0001 (0.80)	-0.0000 (0.41)	-0.0001 (0.55)
10	UG	0.0291 (0.00)	0.0009 (0.01)	0.0017 (0.00)	0.0035 (0.00)
10	RSG	0.0005 (0.04)	-0.0001 (0.49)	-0.0001 (0.65)	-0.0001 (0.51)
10	LG x UG	-0.0002 (0.98)	0.0001 (0.18)	-0.0000 (0.36)	0.0004 (0.05)
10	LG x RSG	0.0001 (0.23)	0.0001 (0.21)	-0.0001 (0.52)	0.0000 (0.26)
10	UG x RSG	0.0002 (0.13)	-0.0002 (0.89)	-0.0002 (0.91)	-0.0001 (0.67)
10	LG x UG x RSG	0.0003 (0.10)	-0.0001 (0.66)	-0.0001 (0.77)	-0.0001 (0.59)
		TREC DD 2017			
2	LG	0.0027 (0.00)	0.0000 (0.30)	0.0001 (0.23)	0.0004 (0.05)
2	UG	0.0004 (0.06)	-0.0000 (0.34)	-0.0001 (0.50)	0.0000 (0.28)
2	RSG	-0.0001 (0.47)	-0.0001 (0.79)	-0.0001 (0.87)	-0.0001 (0.98)
2	LG x UG	0.0002 (0.12)	-0.0001 (0.60)	-0.0001 (0.53)	-0.0000 (0.42)
2	LG x RSG	-0.0001 (0.65)	-0.0001 (0.60)	-0.0001 (0.81)	-0.0001 (0.62)
2	UG x RSG	-0.0001 (0.85)	-0.0001 (0.93)	-0.0001 (0.95)	-0.0001 (0.72)
2	LG x UG x RSG	-0.0000 (0.41)	-0.0001 (0.78)	-0.0001 (0.89)	-0.0001 (0.90)
5	LG	-0.0001 (0.51)	-0.0001 (0.51)	-0.0001 (0.73)	-0.0001 (0.85)
5	UG	0.0023 (0.00)	-0.0001 (0.45)	-0.0001 (0.45)	0.0002 (0.11)
5	RSG	-0.0000 (0.33)	-0.0001 (1.00)	-0.0001 (0.88)	-0.0001 (0.95)
5	LG x UG	-0.0000 (0.32)	-0.0001 (0.83)	-0.0001 (0.57)	-0.0001 (0.50)
5	LG x RSG	0.0002 (0.12)	-0.0001 (0.96)	-0.0001 (0.80)	-0.0001 (0.74)
5	UG x RSG	-0.0001 (0.47)	-0.0001 (0.48)	-0.0001 (0.96)	-0.0001 (0.79)
5	LG x UG x RSG	0.0004 (0.05)	-0.0001 (0.81)	-0.0001 (0.87)	-0.0001 (0.80)
10	LG	0.0002 (0.11)	0.0001 (0.17)	-0.0001 (0.76)	-0.0001 (0.73)
10	UG	0.0043 (0.00)	-0.0000 (0.37)	-0.0000 (0.36)	0.0004 (0.06)
10	RSG	0.0005 (0.03)	-0.0001 (0.88)	-0.0001 (0.90)	-0.0001 (0.99)
10	LG x UG	0.0002 (0.11)	-0.0001 (0.63)	-0.0001 (0.52)	-0.0000 (0.42)
10	LG x RSG	-0.0000 (0.35)	-0.0001 (0.94)	-0.0001 (0.77)	-0.0001 (0.62)
10	UG x RSG	0.0000 (0.26)	-0.0001 (0.58)	-0.0001 (0.95)	-0.0001 (0.76)
10	LG x UG x RSG	0.0007 (0.01)	-0.0001 (0.93)	-0.0001 (0.93)	-0.0001 (0.97)

Table 9. Power analysis of the experiments for estimating the system components. Each cell reports the experiment statistical power for each factor, within parentheses, and the effect size ω^2 of the factor. Insufficient power are in gray (below 0.70); low power are in light blue (0.70-0.80); good power are in blue (0.80-0.90); and high power are in dark blue (0.90-1.0).

Iter	Effects	prec	st-rec	α-nDCG	nCT
2	IR	1.0000 (0.2065)	1.0000 (0.1513)	1.0000 (0.2230)	1.0000 (0.1455)
2	DR	NA (-0.0002)	0.0500 (0.0000)	NA (-0.0002)	0.0657 (0.0001)
2	UFG	0.05684 (0.0058)	NA (-0.0008)	NA (-0.0002)	NA (-0.0001)
5	IR	1.0000 (0.1553)	1.0000 (0.1219)	1.0000 (0.2490)	1.0000 (0.1089)
5	DR	0.0538 (0.0029)	0.0840 (0.0085)	0.0525 (0.0025)	0.0780 (0.0087)
5	UFG	0.2349 (0.0199)	0.0545 (0.0049)	0.0507 (0.0019)	0.0566 (0.0059)
10	IR	1.0000 (0.0897)	0.9815 (0.0656)	1.0000 (0.2522)	1.000 (0.0832)
10	DR	0.3171 (0.0059)	0.5347 (0.0191)	0.0641 (0.0052)	0.2153 (0.0189)
10	UFG	0.2366 (0.0266)	0.7761 (0.0508)	0.0520 (0.0033)	0.0618 (0.0078)
			TREC DD 2017		
2	IR	1.0000 (0.2053)	1.0000 (0.0889)	1.0000 (0.2247)	1.0000 (0.1586)
2	DR	0.0784912 (0.0078)	0.0672 (0.0061)	0.0506 (0.0011)	0.0735 (0.0071)
2	UFG	0.05042 (0.0014)	NA (-0.0005)	NA (-0.0007)	NA (-0.0005)
5	IR	1.0000 (0.1220)	0.9707 (0.0590)	1.0000 (0.2470)	1.0000 (0.1331)
5	DR	0.3047 (0.0217)	0.0552 (0.0034)	0.0504 (0.0009)	0.0792 (0.0079)
5	UFG	0.0532 (0.0039)	NA (-0.0008)	NA (-0.0007)	NA (-0.0006)
10	IR	1.0000 (0.0998)	0.2447750 (0.0224)	1.0000 (0.2564)	1.0000 (0.1185)
10	DR	0.3327 (0.0228)	0.0580 (0.0042)	0.0504 (0.0009)	0.0893 (0.0091)
10	UFG	0.0630 (0.0077)	NA (-0.0007)	NA (-0.0006)	0.0500 (0.0000)

statistically significant effects on performance. However, their effects might depend on the session length and the assessed search quality. The initial ranker has the most prominent and largest effect size on performance on all metrics. We found that these effects might be due to the ability of the initial rankers to retrieve relevant documents and to cover different aspects at the first or early iterations. The dynamic reranker component generally affects performance, particularly in mid to late iterations. The user feedback granularity also affects performance. Within the user feedback granularity dimensions, the unit of information granularity has the most prominent effects.

The effects of the components also differ across search qualities such as relevance and diversity. The initial ranker choice tends to affect all of them. The effects of the dynamic reranker and user feedback granularity components tend to impact on the system's ability to find relevant documents. The dynamic reranker also might impact on the ability of the system to cover more subtopics and find documents with more gains.

***RQ 3**: What are the effects of the factors as we progress in the search session?* The effects of the initial rankers generally start larger in early iterations and gradually decrease. The effects of the dynamic reranker generally might start from the second iteration and gradually increase. Similarly, the feedback granularity tends to increase gradually as the search session progresses and then stabilizes at the session's late stages.

In summary, it is essential to study the characteristics of the components used in dynamic search systems. In particular, the quality of the initial ranker in terms of precision and subtopic recall. As a result, we recommend improving the quality of the initial ranker used to generate the initial ranked list of documents and the dynamic approach that utilizes the user feedback before undertaking a process to elicit more details from users, an option that might be expensive or invasive.

There several lines of work to extend this study. The ANOVA analysis can be performed to estimate the effects of the quality of the initial ranker component in search effectiveness. Our analysis showed that the initial ranker component's precision at the first iteration contributes to the statistically significant variance of effectiveness between dynamic search systems. Therefore, it might be beneficial to perform the analysis using initial rankers with pre-determined effectiveness levels of precision and subtopic recall. For instance, three levels for each metric can be defined: high, medium, and low effectiveness scores. Then, one could have nine initial rankers with various combinations of these levels. Another line of work is to analyze the effects of the system components in other dynamic tasks. In our analysis, we used static topics and document collections. Recently, the TREC COVID-19 track has focused on advancing dynamic search on scientific literature that focuses on medical studies of the COVID-19 pandemic. Although the track uses a collection topically similar to the Ebola epidemic dataset used in TREC DD 2016, it differs from the TREC Dynamic Domain track in that the documents are dynamic. Medical researchers are adding new or removing old documents daily. As a result, systems need to discover new subtopics that might emerge. Therefore, we would expect the effect size of the dynamic reranker component to be larger. Future work also includes the analysis of more sophisticated supervised learning approaches for dynamic search, including neural-based ranking methods [20, 44, 51], diversity-aware learning-to-rank [57, 77], and reinforcement learning [85]. Future work also includes experiments with larger test collections in terms of the number of queries and the size of the document collection, such as the TREC Session Track 2014 test collection [13]. Using larger datasets gives more statistical power to the experiments and increases the reliability of the results. These experiments will require more human resources and time to collect fine-grained explicit relevance judgments. Datasets with implicit feedback introduce a few challenges, such as determining which aspect the user is looking at. Another challenge is how we would induce or infer implicit feedback at the passage level. These questions are interesting to explore. This is also applied to recently released test collections such as the TianGong-ST dataset [16].

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their valuable comments, that helped improve the quality of the paper. This research was partially supported by Australian Research Council (projects LP130100563, DP190101113, and DE200100064), and Real Thing Entertainment Pty Ltd.

REFERENCES

[1] Charu C. Aggarwal. 2016. *Recommender Systems: The Textbook* (1st ed.). Springer Publishing Company, Incorporated, Cham.

[2] Rakesh Agrawal, Sreenivas Gollapudi, Alan Halverson, and Samuel Ieong. 2009. Diversifying Search Results. In *Proc. WSDM*. ACM, New York, NY, USA, 5–14.

[3] Mehmet Akcay, Ismail Sengor Altingovde, Craig Macdonald, and Iadh Ounis. 2017. On the Additivity and Weak Baselines for Search Result Diversification Research. In *Proc. ICTIR*. ACM, New York, NY, USA, 109–116.

[4] Ameer Albahem, Damiano Spina, Falk Scholer, and Lawrence Cavedon. 2019. Meta-Evaluation of Dynamic Search: How Do Metrics Capture Topical Relevance, Diversity and User Effort?. In *Proc. ECIR*. Springer International Publishing, Cham, 607–620.

[5] Ameer Albahem, Damiano Spina, Falk Scholer, Alistair Moffat, and Lawrence Cavedon. 2018. Desirable Properties for Diversity and Truncated Effectiveness Metrics. In *Proc. Aust. Doc. Comp. Symp.* ACM, New York, NY, USA, 9:1–9:7.

[6] Gianni Amati and Cornelis Joost Van Rijsbergen. 2002. Probabilistic Models of Information Retrieval Based on Measuring the Divergence from Randomness. *ACM Trans. Inf. Sys.* 20, 4 (Oct. 2002), 357–389.

[7] Timothy G. Armstrong, Alistair Moffat, William Webber, and Justin Zobel. 2009. Improvements That Don't Add up: Ad-Hoc Retrieval Results since 1998. In *Proc. CIKM*. ACM, New York, NY, USA, 601–610.

[8] David Banks, Paul Over, and Nien-Fan Zhang. 1999. Blind Men and Elephants: Six Approaches to TREC Data. *Inf. Retr.* 1, 1 (1999), 7–34.

[9] Zhou Bo, Qi Fang, Rongwei Cen, Min Zhang, Yiqun Liu, and Shaoping Ma. 2008. THUIR at TREC2008: Relevance Feedback Track1. In *Proc. TREC*, Vol. 500-277. National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, United States.

[10] Elinor Brondwine, Anna Shtok, and Oren Khurland. 2016. Utilizing Focused Relevance Feedback. In *Proc. SIGIR*. ACM, New York, NY, USA, 1061–1064.

[11] Chris Buckley and Robertson Robertson. 2008. Relevance Feedback Track Overview: TREC 2008. In *Proc. TREC*. National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, United States.

[12] Jaime Carbonell and Jade Goldstein. 1998. The Use of MMR, Diversity-based Reranking for Reordering Documents and Producing Summaries. In *Proc. SIGIR*. ACM, New York, NY, USA, 335–336.

[13] Ben Carterette, Evangelos Kanoulas, Mark Hall, and Paul Clough. 2014. Overview of the TREC 2014 Session Track. In *Proc. TREC*. National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, United States.

[14] Benjamin A. Carterette. 2012. Multiple Testing in Statistical Analysis of Systems-Based Information Retrieval Experiments. *ACM Trans. Inf. Syst.* 30, 1, Article 4 (March 2012), 34 pages.

[15] Olivier Chapelle, Donald Metlzer, Ya Zhang, and Pierre Grinspan. 2009. Expected Reciprocal Rank for Graded Relevance. In *Proc. CIKM*. ACM, New York, NY, USA, 621–630.

[16] Jia Chen, Jiaxin Mao, Yiqun Liu, Min Zhang, and Shaoping Ma. 2019. TianGong-ST: A New Dataset with Large-scale Refined Real-world Web Search Sessions. In *Proc. CIKM*. ACM, New York, NY, USA.

[17] Limin Chen, Zhiwen Tang, and Grace Hui Yang. 2020. Balancing Reinforcement Learning Training Experiences in Interactive Information Retrieval. In *Proc. SIGIR*. ACM, New York, NY, USA, 1525–1528.

[18] 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 *Proc. SIGIR* (Singapore, Singapore). ACM, New York, NY, USA, 659–666.

[19] Charles L. A. Clarke, Nick Craswell, and Ellen M. Voorhees. 2012. Overview of the TREC 2012 Web Track. In *Proc. TREC*. National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, United States.

[20] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Jimmy Lin. 2021. MS MARCO: Benchmarking Ranking Models in the Large-Data Regime. In *Proc. SIGIR*. ACM, New York, NY, USA, 1566–1576.

[21] Van Dang and W. Bruce Croft. 2012. Diversity by Proportionality: An Election-based Approach to Search Result Diversification. In *Proc. SIGIR*. ACM, New York, NY, USA, 65–74.

[22] Van Dang and W. Bruce Croft. 2013. Term Level Search Result Diversification. In *Proc. SIGIR*. ACM, New York, NY, USA, 603–612.

[23] Guglielmo Faggioli and Nicola Ferro. 2021. System Effect Estimation by Sharding: A Comparison Between ANOVA Approaches to Detect Significant Differences. In *Proc. ECIR*. Springer International Publishing, Cham, 33–46.

[24] Marco Ferrante, Nicola Ferro, and Norbert Fuhr. 2021. Towards Meaningful Statements in IR Evaluation. Mapping Evaluation Measures to Interval Scales. arXiv:2101.02668 [cs.IR]

[25] Nicola Ferro and Mark Sanderson. 2019. Improving the Accuracy of System Performance Estimation by Using Shards. In *Proc. SIGIR*. ACM, New York, NY, USA, 805–814.

[26] Nicola Ferro and Gianmaria Silvello. 2016. A General Linear Mixed Models Approach to Study System Component Effects. In *Proc. SIGIR*. ACM, New York, NY, USA, 25–34.

[27] Nicola Ferro and Gianmaria Silvello. 2018. Toward an Anatomy of IR System Component Performances. *JASIST* 69, 2 (2018), 187–200.

[28] Ning Gao, Zhi-Hong Deng, Jia-Jian Jiang, Sheng-Long Lv, and Hang Yu. 2011. Combining Strategies for XML Retrieval. In *Comparative Evaluation of Focused Retrieval*, Shlomo Geva, Jaap Kamps, Ralf Schenkel, and Andrew Trotman (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 319–331.

[29] Gene Golovchinsky, Morgan N. Price, and Bill N. Schilit. 1999. From Reading to Retrieval: Freeform Ink Annotations as Queries. In *Proc. SIGIR*. ACM, New York, NY, USA, 19–25.

[30] Donna Harman and Chris Buckley. 2009. Overview of the Reliable Information Access Workshop. *Inf. Retr.* 12, 6 (2009), 615–641.

[31] Sha Hu, Zhicheng Dou, Xiaojie Wang, Tetsuya Sakai, and Ji-Rong Wen. 2015. Search Result Diversification Based on Hierarchical Intents. In *Proc. CIKM*. ACM, New York, NY, USA, 63–72.

[32] P.K. Ito. 1980. 7 Robustness of ANOVA and MANOVA test procedures. In *Analysis of Variance*. Handbook of Statistics, Vol. 1. Elsevier, Amsterdam, 199–236.
[33] Jiepu Jiang, Daqing He, and James Allan. 2017. Comparing In Situ and Multidimensional Relevance Judgments. In *Proc. SIGIR*. ACM, New York, NY, USA, 405–414.
[34] Zhengbao Jiang, Ji-Rong Wen, Zhicheng Dou, Wayne Xin Zhao, Jian-Yun Nie, and Ming Yue. 2017. Learning to Diversify Search Results via Subtopic Attention. In *Proc. SIGIR*. ACM, New York, NY, USA, 545–554.
[35] Xiaoran Jin, Marc Sloan, and Jun Wang. 2013. Interactive Exploratory Search for Multi Page Search Results. In *Proc. WWW*. ACM, New York, NY, USA, 655–666.
[36] Luo Jiyun and Hui Yang. 2015. Re-ranking via User Feedback: Georgetown University at TREC 2015 DD Track. In *Proc. TREC*. National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, United States.
[37] Robin Joganah, Richard Khoury, and Luc Lamontagne. 2015. Laval University and Lakehead University at TREC Dynamic Domain 2015 : Combination of Techniques for Subtopics Coverage. In *Proc. TREC*. National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, United States.
[38] Evangelos Kanoulas, Ben Carterette, Paul D. Clough, and Mark Sanderson. 2011. Evaluating Multi-query Sessions. In *Proc. SIGIR* (Beijing, China). ACM, New York, NY, USA, 1053–1062.
[39] Sadegh Kharazmi, Falk Scholer, David Vallet, and Mark Sanderson. 2016. Examining Additivity and Weak Baselines. *ACM Trans. Inf. Sys.* 34, 4, Article 23 (June 2016), 18 pages.
[40] Victor Lavrenko and W. Bruce Croft. 2001. Relevance Based Language Models. In *Proc. SIGIR*. ACM, New York, NY, USA, 120–127.
[41] Anton Leuski. 2001. Evaluating Document Clustering for Interactive Information Retrieval. In *Proc. CIKM*. ACM, New York, NY, USA, 33–40.
[42] Nir Levine, Haggai Roitman, and Doron Cohen. 2017. An Extended Relevance Model for Session Search. In *Proc. SIGIR*. ACM, New York, NY, USA, 865–868.
[43] Cheng Li, Paul Resnick, and Qiaozhu Mei. 2016. Multiple Queries As Bandit Arms. In *Proc. CIKM*. ACM, New York, NY, USA, 1089–1098.
[44] Jimmy Lin, Daniel Campos, Nick Craswell, Bhaskar Mitra, and Emine Yilmaz. 2021. Significant Improvements over the State of the Art? A Case Study of the MS MARCO Document Ranking Leaderboard. In *Proc. SIGIR*. ACM, New York, NY, USA, 2283–2287.
[45] Jiongnan Liu, Zhicheng Dou, Xiaojie Wang, Shuqi Lu, and Ji-Rong Wen. 2020. DVGAN: A Minimax Game for Search Result Diversification Combining Explicit and Implicit Features. In *Proc. SIGIR*. ACM, New York, NY, USA, 479–488.
[46] Xiaolu Lu, Oren Kurland, J. Shane Culpepper, Nick Craswell, and Ofri Rom. 2019. Relevance Modeling with Multiple Query Variations. In *Proc. SIGIR*. ACM, New York, NY, USA, 27–34.
[47] Jiyun Luo, Christopher Wing, Hui Yang, and Marti Hearst. 2013. The Water Filling Model and the Cube Test: Multi-dimensional Evaluation for Professional Search. In *Proc. CIKM*. ACM, New York, NY, USA, 709–714.
[48] Jiyun Luo, Sicong Zhang, and Hui Yang. 2014. Win-win Search: Dual-agent Stochastic Game in Session Search. In *Proc. SIGIR*. ACM, New York, NY, USA, 587–596.
[49] Angelini Marco, Vanessa Fazzini, Nicola Ferro, Giuseppe Santucci, and Gianmaria Silvello. 2018. CLAIRE: A Combinatorial Visual Analytics System for Information Retrieval Evaluation. *Inf. Proc. & Man.* 54, 6 (2018), 1077 – 1100.
[50] David Maxwell, Leif Azzopardi, and Yashar Moshfeghi. 2019. The Impact of Result diversification on Search Behaviour and Performance. *Inf. Retr.* 22 (16 May 2019), 422–446. Issue 5.
[51] Bhaskar Mitra, Nick Craswell, et al. 2018. An introduction to neural information retrieval. *Found. & Trends in IR* 13, 1 (2018), 1–126.
[52] Felipe Moraes. 2017. *On Effective Dynamic Search Systems.* Master thesis. University of Minas Gerais.
[53] Felipe Moraes, Rodrygo L.T. Santos, and Nivio Ziviani. 2017. On Effective Dynamic Search in Specialized Domains. In *Proc. ICTIR*. ACM, New York, NY, USA, 177–184.
[54] Felipe Moraes, Rodrygo L T Santos, and Nivio Ziviani. 2016. UFMG at the TREC 2016 Dynamic Domain Track. In *Proc. TREC*. National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, United States.
[55] Paul Over. 1999. TREC-7 Interactive Track Report. In *Proc. TREC* (Gaithersburg, Maryland). National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, United States.
[56] Jay M. Ponte and W. Bruce Croft. 1998. A Language Modeling Approach to Information Retrieval. In *Proc. SIGIR*. ACM, New York, NY, USA, 275–281.
[57] Xubo Qin, Zhicheng Dou, and Ji-Rong Wen. 2020. Diversifying Search Results Using Self-Attention Network. In *Proc. CIKM* (Virtual Event, Ireland). ACM, New York, NY, USA, 1265–1274.
[58] Razieh Rahimi and Grace Hui Yang. 2016. An Investigation of Basic Retrieval Models for the Dynamic Domain Task. In *Proc. TREC*. National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, United States.
[59] Fiana Raiber and Oren Khurland. 2019. Relevance Feedback: The Whole Is Inferior to the Sum of Its Parts. *ACM Trans. Inf. Sys.* 37, 4, Article 44 (Oct. 2019), 28 pages.

[60] Karthik Raman, Paul N. Bennett, and Kevyn Collins-Thompson. 2013. Toward Whole-session Relevance: Exploring Intrinsic Diversity in Web Search. In *Proc. SIGIR*. ACM, New York, NY, USA, 463–472.
[61] Karthik Raman, Thorsten Joachims, and Pannaga Shivaswamy. 2011. Structured Learning of Two-level Dynamic Rankings. In *Proc. CIKM*. ACM, New York, NY, USA, 291–296.
[62] J.J Rocchio. 1971. *Relevance Feedback in Information Retrieval.* Prentice Hall, USA.
[63] Kevin Roitero, Ben Carterrete, Rishabh Mehrotra, and Mounia Lalmas. 2020. Leveraging Behavioral Heterogeneity Across Markets for Cross-Market Training of Recommender Systems. In *Proc. WWW*. ACM, New York, NY, USA, 694–702.
[64] Rodrygo L.T. Santos, Craig Macdonald, and Iadh Ounis. 2010. Exploiting Query Reformulations for Web Search Result Diversification. In *Proc. WWW*. ACM, New York, NY, USA, 881–890.
[65] Rodrygo LT Santos, Craig Macdonald, and Iadh Ounis. 2015. Search Result Diversification. *Found. & Trends in IR* 9, 1 (2015), 1–90.
[66] Xuehua Shen and ChengXiang Zhai. 2005. Active Feedback in Ad Hoc Information Retrieval. In *Proc. SIGIR*. ACM, New York, NY, USA, 59–66.
[67] Marc Sloan and Jun Wang. 2015. Dynamic Information Retrieval: Theoretical Framework and Application. In *Proc. ICTIR*. ACM, New York, NY, USA, 61–70.
[68] Esben Sørig, Nicolas Collignon, Rebecca Fiebrink, and Noriko Kando. 2019. Evaluation of Rich and Explicit Feedback for Exploratory Search. In *Proc. CHIIR*. ACM, New York, NY, USA.
[69] Robertson Stephen, S. Walker, S. Jones, M.M. Hancock-Beaulieu, and M. Gatford. 1996. Okapi at TREC-3. In *Proc. TREC*. National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, United States, 109–126.
[70] J. Tague-Sutcliffe and J. Blustein. 1994. A Statistical Analysis of the TREC-3 Data. In *Proc. TREC*. National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, United States.
[71] Zhiwen Tang and Grace Hui Yang. 2017. Georgetown University at TREC 2017 Dynamic Domain Track. In *Proc. TREC*. National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, United States.
[72] Ellen M. Voorhees, Daniel Samarov, and Ian Soboroff. 2017. Using Replicates in Information Retrieval Evaluation. *ACM Trans. Inf. Sys.* 36, 2, Article 12 (Aug. 2017), 21 pages.
[73] Zhijing Wu, Jiaxin Mao, Yiqun Liu, Min Zhang, and Shaoping Ma. 2019. Investigating Passage-Level Relevance and Its Role in Document-Level Relevance Judgment. In *Proc. SIGIR*. ACM, New York, NY, USA, 605–614.
[74] Long Xia, Jun Xu, Yanyan Lan, Jiafeng Guo, and Xueqi Cheng. 2015. Learning maximal marginal relevance model via directly optimizing diversity evaluation measures. In *Proc. SIGIR*. ACM, New York, NY, USA, 113–122.
[75] Long Xia, Jun Xu, Yanyan Lan, Jiafeng Guo, and Xueqi Cheng. 2016. Modeling document novelty with neural tensor network for search result diversification. In *Proc. SIGIR*. ACM, New York, NY, USA, 395–404.
[76] Jun Xu, Long Xia, Yanyan Lan, Jiafeng Guo, and Xueqi Cheng. 2017. Directly optimize diversity evaluation measures: a new approach to search result diversification. *ACM Trans. Int. Sys. Tech.* 8, 3 (2017), 1–26.
[77] Le Yan, Zhen Qin, Rama Kumar Pasumarthi, Xuanhui Wang, and Michael Bendersky. 2021. Diversification-Aware Learning to Rank Using Distributed Representation. In *Proc. The Web Conference 2021* (Ljubljana, Slovenia) *(WWW '21)*. ACM, New York, NY, USA, 10 pages.
[78] Angela Yang and Grace Hui Yang. 2017. A Contextual Bandit Approach to Dynamic Search. In *Proc. ICTIR*. ACM, New York, NY, USA, 301–304.
[79] Hui Yang, John Frank, and Ian Soboroff. 2015. TREC 2015 Dynamic Domain Track Overview. In *Proc. TREC*. National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, United States.
[80] Hui Yang and Ian Soboroff. 2016. TREC 2016 Dynamic Domain Track Overview. In *Proc. TREC*. National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, United States.
[81] Hui Yang, Zhiwen Tang, and Ian Soboroff. 2017. TREC 2017 Dynamic Domain Track Overview. In *Proc. TREC*, Vol. 500-324. National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, United States.
[82] Fabio Zampieri, Kevin Roitero, J. Shane Culpepper, Oren Kurland, and Stefano Mizzaro. 2019. On Topic Difficulty in IR Evaluation: The Effect of Systems, Corpora, and System Components. In *Proc. SIGIR*. ACM, New York, NY, USA, 909–912.
[83] Chengxiang Zhai and John Lafferty. 2004. A Study of Smoothing Methods for Language Models Applied to Information Retrieval. *ACM Trans. Inf. Sys.* 22, 2 (April 2004), 179–214.
[84] Weimin Zhang, Yong-Cheng Hu, Rongqian Jia, Xianfa Wang, Le Zhang, Yue Feng, Simon Chun Ho Yu, Yuanhai Xue, Xiaoming Yu, Yue Liu, and Xueqi Cheng. 2017. ICTNET at TREC 2017 Dynamic Domain Track. In *Proc. TREC*. National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, United States.
[85] Weinan Zhang, Xiangyu Zhao, Li Zhao, Dawei Yin, Grace Hui Yang, and Alex Beutel. 2020. Deep reinforcement learning for information retrieval: Fundamentals and advances. In *Proc. SIGIR*. ACM, New York, NY, USA, 2468–2471.
[86] Le Zhao, Chenmin Liang, and Jamie Callan. 2008. Extending Relevance Model for Relevance Feedback. In *Proc. TREC*. National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, United States, 500–277.

[87] Tang Zhiwen and Grace Hui Yang. 2020. Corpus-Level End-to-End Exploration for Interactive Systems. In *Proc. AAAI*. AAAI Press, USA, 2527–2534.
[88] Jianghong Zhou and Eugene Agichtein. 2020. RLIRank: Learning to Rank with Reinforcement Learning for Dynamic Search. In *Proc. WWW* (Taipei, Taiwan). ACM, New York, NY, USA, 2842–2848.
[89] Yadong Zhu, Yanyan Lan, Jiafeng Guo, Xueqi Cheng, and Shuzi Niu. 2014. Learning for search result diversification. In *Proc. SIGIR*. ACM, New York, NY, USA, 293–302.

Appendix A DETAILED ANOVA TABLES

Table 10. ANOVA analysis based on the nCT metric for Equation 18, on TREC DD 2016 collection. .

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	Topic	18701.2561	52	359.6395	194.8446	0.000	0.6131
1	System	2738.4695	99	27.6613	14.9863	0.000	0.1788
1	Residual	9502.0562	5148	1.8458			
2	Topic	124769.6474	52	2399.4163	198.6063	0.000	0.6177
2	System	13247.2718	99	133.8108	11.0759	0.000	0.1356
2	Residual	62194.3824	5148	12.0813			
5	Topic	306185.1625	52	5888.1762	207.5622	0.000	0.6281
5	System	25771.1377	99	260.3145	9.1763	0.000	0.1129
5	Residual	146039.7728	5148	28.3683			
10	Topic	535424.4085	52	10296.6232	345.1706	0.000	0.7378
10	System	23697.6350	99	239.3701	8.0243	0.000	0.0986
10	Residual	153567.5774	5148	29.8305			

Table 11. ANOVA analysis based on the α-nDCG metric for Equation 18, on TREC DD 2016 collection. .

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	Topic	24772.9312	52	476.4025	194.7508	0.000	0.6130
1	System	5075.6619	99	51.2693	20.9586	0.000	0.2370
1	Residual	12593.1208	5148	2.4462			
2	Topic	205474.5960	52	3951.4345	221.3304	0.000	0.6430
2	System	32413.0444	99	327.4045	18.3388	0.000	0.2125
2	Residual	91907.7593	5148	17.8531			
5	Topic	485470.2953	52	9335.9672	244.2164	0.000	0.6654
5	System	81595.9533	99	824.2015	21.5600	0.000	0.2424
5	Residual	196799.0387	5148	38.2283			
10	Topic	657795.6751	52	12649.9168	251.4264	0.000	0.6719
10	System	110565.5974	99	1116.8242	22.1977	0.000	0.2481
10	Residual	259009.2328	5148	50.3126			

Table 12. ANOVA analysis based on the prec metric for Equation 18, on TREC DD 2016 collection. .

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	Topic	400.2617	52	7.6973	177.7641	0.000	0.5910
1	System	79.2796	99	0.8008	18.4940	0.000	0.2140
1	Residual	222.9129	5148	0.0433			
2	Topic	1050.1612	52	20.1954	189.9892	0.000	0.6071
2	System	181.5572	99	1.8339	17.2526	0.000	0.2019
2	Residual	547.2203	5148	0.1063			
5	Topic	3468.9782	52	66.7111	310.7376	0.000	0.7169
5	System	300.5359	99	3.0357	14.1402	0.000	0.1698
5	Residual	1105.2055	5148	0.2147			
10	Topic	7364.7351	52	141.6295	489.3870	0.000	0.7997
10	System	274.5746	99	2.7735	9.5835	0.000	0.1179
10	Residual	1489.8408	5148	0.2894			

Table 13. ANOVA analysis based on the st-rec metric for Equation 18, on TREC DD 2016 collection. .

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	Topic	5166.8770	52	99.3630	191.3617	0.000	0.6088
1	System	719.8913	99	7.2716	14.0043	0.000	0.1683
1	Residual	2673.0569	5148	0.5192			
2	Topic	9287.6145	52	178.6080	200.6282	0.000	0.6201
2	System	1011.4528	99	10.2167	11.4763	0.000	0.1402
2	Residual	4582.9731	5148	0.8902			
5	Topic	10473.2820	52	201.4093	163.5000	0.000	0.5706
5	System	1244.6295	99	12.5720	10.2057	0.000	0.1253
5	Residual	6341.6201	5148	1.2319			
10	Topic	10739.5059	52	206.5290	182.6049	0.000	0.5976
10	System	704.2180	99	7.1133	6.2893	0.000	0.0761
10	Residual	5822.4668	5148	1.1310			

Table 14. ANOVA analysis based on the nCT metric for Equation 18, on TREC DD 2017 collection.

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	Topic	519.2073	59	8.8001	535.4561	0.000	0.8141
1	System	50.3584	119	0.4232	25.7490	0.000	0.2903
1	Residual	115.3889	7021	0.0164			
2	Topic	479.2006	59	8.1220	505.5517	0.000	0.8052
2	System	28.3055	119	0.2379	14.8056	0.000	0.1858
2	Residual	112.7973	7021	0.0161			
5	Topic	352.7752	59	5.9792	613.1415	0.000	0.8338
5	System	15.2910	119	0.1285	13.1766	0.000	0.1675
5	Residual	68.4675	7021	0.0098			
10	Topic	270.3234	59	4.5818	725.2077	0.000	0.8558
10	System	8.3586	119	0.0702	11.1178	0.000	0.1433
10	Residual	44.3576	7021	0.0063			

Table 15. ANOVA analysis based on the α-nDCG metric for Equation 18, on TREC DD 2017 collection. .

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	Topic	67065.2127	59	1136.6985	311.1331	0.000	0.7176
1	System	9581.4928	99	96.7828	26.4910	0.000	0.2595
1	Residual	21339.5997	5841	3.6534			
2	Topic	854735.2180	59	14487.0376	285.6686	0.000	0.6999
2	System	105681.0863	99	1067.4857	21.0497	0.000	0.2161
2	Residual	296213.1479	5841	50.7127			
5	Topic	1301893.1359	59	22065.9854	291.1080	0.000	0.7039
5	System	178722.5232	99	1805.2780	23.8163	0.000	0.2388
5	Residual	442747.7991	5841	75.8000			
10	Topic	1336732.1441	59	22656.4770	279.5890	0.000	0.6954
10	System	200869.7997	99	2028.9879	25.0384	0.000	0.2484
10	Residual	473325.0474	5841	81.0349			

Table 16. ANOVA analysis based on the prec metric for Equation 18, on TREC DD 2017 collection.

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	Topic	546.6208	59	9.2648	357.9661	0.000	0.7452
1	System	110.5088	119	0.9286	35.8804	0.000	0.3657
1	Residual	181.7152	7021	0.0259			
2	Topic	449.3662	59	7.6164	407.2487	0.000	0.7690
2	System	52.2485	119	0.4391	23.4767	0.000	0.2709
2	Residual	131.3070	7021	0.0187			
5	Topic	284.9547	59	4.8297	442.1653	0.000	0.7833
5	System	21.7121	119	0.1825	16.7038	0.000	0.2061
5	Residual	76.6899	7021	0.0109			
10	Topic	225.4060	59	3.8204	604.0597	0.000	0.8317
10	System	12.8657	119	0.1081	17.0943	0.000	0.2101
10	Residual	44.4051	7021	0.0063			

Table 17. ANOVA analysis based on the st-rec metric for Equation 18, on TREC DD 2017 collection.

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	Topic	608.8919	59	10.3202	299.6996	0.000	0.7099
1	System	61.8281	119	0.5196	15.0882	0.000	0.1889
1	Residual	241.7692	7021	0.0344			
2	Topic	423.2458	59	7.1737	205.5996	0.000	0.6264
2	System	36.0280	119	0.3028	8.6771	0.000	0.1126
2	Residual	244.9725	7021	0.0349			
5	Topic	189.4492	59	3.2110	147.8690	0.000	0.5462
5	System	14.8394	119	0.1247	5.7426	0.000	0.0727
5	Residual	152.4624	7021	0.0217			
10	Topic	88.8848	59	1.5065	105.6290	0.000	0.4616
10	System	4.7340	119	0.0398	2.7893	0.000	0.0287
10	Residual	100.1362	7021	0.0143			

Table 18. ANOVA analysis based on the nCT metric for Equation 19, on TREC DD 2016 collection. Redundant factors in previous models are omitted for brevity. .

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	IR	2738.4695	4	684.6174	377.1065	0.000	0.1913
1	DR	0.0000	2	0.0000	0.0000	1.000	-0.0003
1	UFG	0.0000	7	0.0000	0.0000	1.000	-0.0011
1	Residual	9502.0562	5234	1.8154			
2	IR	12955.9912	4	3238.9978	271.7987	0.000	0.1455
2	DR	33.9100	2	16.9550	1.4228	0.241	0.0001
2	UFG	78.6926	7	11.2418	0.9433	0.471	-0.0001
2	Residual	62373.0604	5234	11.9169			
5	IR	21948.0417	4	5487.0104	195.3916	0.000	0.1089
5	DR	1618.1144	2	809.0572	28.8104	0.000	0.0087
5	UFG	1262.9059	7	180.4151	6.4246	0.000	0.0059
5	Residual	146981.8486	5234	28.0821			
10	IR	17188.7081	4	4297.1770	145.3810	0.000	0.0832
10	DR	3690.2699	2	1845.1350	62.4241	0.000	0.0189
10	UFG	1679.4511	7	239.9216	8.1170	0.000	0.0078
10	Residual	154706.7833	5234	29.5580			

Table 19. ANOVA analysis based on the α-nDCG metric for Equation 19, on TREC DD 2016 collection. Redundant factors in previous models are omitted for brevity. .

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	IR	5075.6619	4	1268.9155	527.3914	0.000	0.2487
1	DR	0.0000	2	0.0000	0.0000	1.000	-0.0003
1	UFG	0.0000	7	0.0000	0.0000	1.000	-0.0011
1	Residual	12593.1208	5234	2.4060			
2	IR	32164.9860	4	8041.2465	457.2448	0.000	0.2230
2	DR	9.7155	2	4.8578	0.2762	0.759	-0.0002
2	UFG	99.3871	7	14.1982	0.8073	0.581	-0.0002
2	Residual	92046.7151	5234	17.5863			
5	IR	79668.8546	4	19917.2137	528.2495	0.000	0.2490
5	DR	672.5204	2	336.2602	8.9184	0.000	0.0025
5	UFG	709.9519	7	101.4217	2.6899	0.009	0.0019
5	Residual	197343.6650	5234	37.7042			
10	IR	106650.9807	4	26662.7452	537.1524	0.000	0.2522
10	DR	1738.0579	2	869.0289	17.5076	0.000	0.0052
10	UFG	1384.6621	7	197.8089	3.9851	0.000	0.0033
10	Residual	259801.1296	5234	49.6372			

Table 20. ANOVA analysis based on the prec metric for Equation 19, on TREC DD 2016 collection. Redundant factors in previous models are omitted for brevity. .

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	IR	79.2796	4	19.8199	465.3720	0.000	0.2260
1	DR	0.0000	2	0.0000	0.0000	1.000	-0.0003
1	UFG	0.0000	7	0.0000	0.0000	1.000	-0.0011
1	Residual	222.9129	5234	0.0426			
2	IR	174.3072	4	43.5768	414.8594	0.000	0.2065
2	DR	0.0455	2	0.0228	0.2166	0.805	-0.0002
2	UFG	4.6460	7	0.6637	6.3187	0.000	0.0058
2	Residual	549.7789	5234	0.1050			
5	IR	251.3506	4	62.8377	293.4308	0.000	0.1553
5	DR	4.3404	2	2.1702	10.1341	0.000	0.0029
5	UFG	29.1989	7	4.1713	19.4785	0.000	0.0199
5	Residual	1120.8514	5234	0.2141			
10	IR	182.9411	4	45.7353	157.7269	0.000	0.0897
10	DR	11.4525	2	5.7263	19.7481	0.000	0.0059
10	UFG	52.3450	7	7.4779	25.7888	0.000	0.0266
10	Residual	1517.6768	5234	0.2900			

Table 21. ANOVA analysis based on the st-rec metric for Equation 19, on TREC DD 2016 collection. Redundant factors in previous models are omitted for brevity. .

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	IR	719.8913	4	179.9728	352.3972	0.000	0.1810
1	DR	0.0000	2	0.0000	0.0000	1.000	-0.0003
1	UFG	0.0000	7	0.0000	0.0000	1.000	-0.0011
1	Residual	2673.0569	5234	0.5107			
2	IR	998.4197	4	249.6049	284.4666	0.000	0.1513
2	DR	2.0049	2	1.0025	1.1425	0.319	0.0000
2	UFG	1.4330	7	0.2047	0.2333	0.977	-0.0008
2	Residual	4592.5683	5234	0.8774			
5	IR	1082.4113	4	270.6028	221.6998	0.000	0.1219
5	DR	68.7576	2	34.3788	28.1659	0.000	0.0085
5	UFG	46.5552	7	6.6507	5.4488	0.000	0.0049
5	Residual	6388.5256	5234	1.2206			
10	IR	503.9750	4	125.9937	112.5606	0.000	0.0656
10	DR	141.0247	2	70.5124	62.9945	0.000	0.0191
10	UFG	23.0542	7	3.2935	2.9423	0.004	0.0021
10	Residual	5858.6308	5234	1.1193			

Table 22. ANOVA analysis based on the nCT metric for Equation 19, on TREC DD 2017 collection. Redundant factors in previous models are omitted for brevity. .

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	IR	4275.1925	4	1068.7981	610.0734	0.000	0.2528
1	DR	0.0000	2	0.0000	0.0000	1.000	-0.0003
1	UFG	0.0000	7	0.0000	0.0000	1.000	-0.0010
1	Residual	10383.6138	5927	1.7519			
2	IR	17897.3963	4	4474.3491	340.3333	0.000	0.1586
2	DR	704.9115	2	352.4558	26.8089	0.000	0.0071
2	UFG	41.2324	7	5.8903	0.4480	0.872	-0.0005
2	Residual	77922.0553	5927	13.1470			
5	IR	31503.2420	4	7875.8105	277.4187	0.000	0.1331
5	DR	1682.1200	2	841.0600	29.6256	0.000	0.0079
5	UFG	81.3201	7	11.6172	0.4092	0.897	-0.0006
5	Residual	168265.2730	5927	28.3896			
10	IR	29852.1425	4	7463.0356	243.0811	0.000	0.1185
10	DR	2089.9938	2	1044.9969	34.0370	0.000	0.0091
10	UFG	225.6535	7	32.2362	1.0500	0.394	0.0000
10	Residual	181969.7571	5927	30.7018			

Table 23. ANOVA analysis based on the α-nDCG metric for Equation 19, on TREC DD 2017 collection. Redundant factors in previous models are omitted for brevity. .

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	IR	9581.4928	4	2395.3732	665.3066	0.000	0.2696
1	DR	0.0000	2	0.0000	0.0000	1.000	-0.0003
1	UFG	0.0000	7	0.0000	0.0000	1.000	-0.0010
1	Residual	21339.5997	5927	3.6004			
2	IR	104646.5821	4	26161.6455	522.7022	0.000	0.2247
2	DR	494.8749	2	247.4375	4.9437	0.007	0.0011
2	UFG	101.8780	7	14.5540	0.2908	0.958	-0.0007
2	Residual	296650.8991	5927	50.0508			
5	IR	177047.2750	4	44261.8187	591.3432	0.000	0.2470
5	DR	635.5982	2	317.7991	4.2458	0.014	0.0009
5	UFG	153.6862	7	21.9552	0.2933	0.957	-0.0007
5	Residual	443633.7629	5927	74.8496			
10	IR	198987.8938	4	49746.9734	621.6263	0.000	0.2564
10	DR	688.0816	2	344.0408	4.2991	0.014	0.0009
10	UFG	197.9946	7	28.2849	0.3534	0.929	-0.0006
10	Residual	474320.8771	5927	80.0271			

Table 24. ANOVA analysis based on the prec metric for Equation 19, on TREC DD 2017 collection. Redundant factors in previous models are omitted for brevity.

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	IR	110.5088	4	27.6272	1083.5585	0.000	0.3756
1	DR	0.0000	2	0.0000	0.0000	1.000	-0.0003
1	UFG	0.0000	7	0.0000	0.0000	1.000	-0.0010
1	Residual	181.7152	7127	0.0255			
2	IR	47.3003	4	11.8251	628.0455	0.000	0.2584
2	DR	1.3519	2	0.6759	35.9003	0.000	0.0096
2	UFG	0.7135	7	0.1019	5.4136	0.000	0.0043
2	Residual	134.1897	7127	0.0188			
5	IR	16.1582	4	4.0395	358.6825	0.000	0.1658
5	DR	1.6047	2	0.8024	71.2445	0.000	0.0191
5	UFG	0.3737	7	0.0534	4.7398	0.000	0.0036
5	Residual	80.2654	7127	0.0113			
10	IR	8.5949	4	2.1487	324.0873	0.000	0.1522
10	DR	1.0238	2	0.5119	77.2085	0.000	0.0207
10	UFG	0.3994	7	0.0571	8.6065	0.000	0.0073
10	Residual	47.2526	7127	0.0066			

Table 25. ANOVA analysis based on the st-rec metric for Equation 19, on TREC DD 2017 collection. Redundant factors in previous models are omitted for brevity.

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	IR	61.8281	4	15.4570	455.6506	0.000	0.2017
1	DR	0.0000	2	0.0000	0.0000	1.000	-0.0003
1	UFG	0.0000	7	0.0000	0.0000	1.000	-0.0010
1	Residual	241.7692	7127	0.0339			
2	IR	33.7234	4	8.4309	244.1209	0.000	0.1190
2	DR	1.0365	2	0.5183	15.0066	0.000	0.0039
2	UFG	0.1055	7	0.0151	0.4366	0.880	-0.0005
2	Residual	246.1351	7127	0.0345			
5	IR	13.3154	4	3.3289	154.4992	0.000	0.0786
5	DR	0.3894	2	0.1947	9.0375	0.000	0.0022
5	UFG	0.0379	7	0.0054	0.2510	0.972	-0.0007
5	Residual	153.5591	7127	0.0215			
10	IR	3.5864	4	0.8966	63.2740	0.000	0.0334
10	DR	0.2468	2	0.1234	8.7072	0.000	0.0021
10	UFG	0.0481	7	0.0069	0.4850	0.846	-0.0005
10	Residual	100.9890	7127	0.0142			

Table 26. ANOVA analysis based on the nCT metric for Equation 19, on TREC DD 2016 collection. Redundant factors in previous models are omitted for brevity. .

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	IR x DR	0.0000	8	0.0000	0.0000	1.000	-0.0013
1	IR x UFG	0.0000	28	0.0000	0.0000	1.000	-0.0044
1	DR x UFG	44.7197	14	3.1943	0.5889	0.876	-0.0009
1	IR x DR x UFG	93.2696	56	1.6655	0.3071	1.000	-0.0061
1	Residual	28203.3123	5200	5.4237			
2	IR x DR	63.1071	8	7.8884	0.2194	0.988	-0.0010
2	IR x UFG	71.2536	28	2.5448	0.0708	1.000	-0.0041
2	DR x UFG	69.4534	14	4.9610	0.1380	1.000	-0.0019
2	IR x DR x UFG	323.6378	56	5.7792	0.1607	1.000	-0.0074
2	Residual	186964.0297	5200	35.9546			
5	IR x DR	326.6369	8	40.8296	0.4695	0.878	-0.0007
5	IR x UFG	409.5047	28	14.6252	0.1682	1.000	-0.0037
5	DR x UFG	643.0939	14	45.9353	0.5282	0.918	-0.0010
5	IR x DR x UFG	1427.2082	56	25.4859	0.2931	1.000	-0.0063
5	Residual	452224.9353	5200	86.9663			
10	IR x DR	353.9503	8	44.2438	0.3339	0.953	-0.0008
10	IR x UFG	160.2934	28	5.7248	0.0432	1.000	-0.0042
10	DR x UFG	737.3077	14	52.6648	0.3975	0.976	-0.0013
10	IR x DR x UFG	3062.7471	56	54.6919	0.4128	1.000	-0.0052
10	Residual	688991.9859	5200	132.4985			

Table 27. ANOVA analysis based on the α-nDCG metric for Equation 19, on TREC DD 2016 collection. Redundant factors in previous models are omitted for brevity. .

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	IR x DR	0.0000	8	0.0000	0.0000	1.000	-0.0013
1	IR x UFG	0.0000	28	0.0000	0.0000	1.000	-0.0044
1	DR x UFG	31.5837	14	2.2560	0.3140	0.993	-0.0015
1	IR x DR x UFG	118.3238	56	2.1129	0.2940	1.000	-0.0063
1	Residual	37366.0520	5200	7.1858			
2	IR x DR	65.6145	8	8.2018	0.1434	0.997	-0.0011
2	IR x UFG	47.6570	28	1.7020	0.0298	1.000	-0.0043
2	DR x UFG	59.1255	14	4.2233	0.0738	1.000	-0.0020
2	IR x DR x UFG	822.3314	56	14.6845	0.2568	1.000	-0.0066
2	Residual	297382.3553	5200	57.1889			
5	IR x DR	311.5450	8	38.9431	0.2968	0.967	-0.0009
5	IR x UFG	117.6373	28	4.2013	0.0320	1.000	-0.0043
5	DR x UFG	388.3127	14	27.7366	0.2114	0.999	-0.0017
5	IR x DR x UFG	1390.2427	56	24.8258	0.1892	1.000	-0.0072
5	Residual	682269.3340	5200	131.2056			
10	IR x DR	437.5402	8	54.6925	0.3102	0.963	-0.0009
10	IR x UFG	102.2912	28	3.6533	0.0207	1.000	-0.0043
10	DR x UFG	645.5579	14	46.1113	0.2615	0.997	-0.0016
10	IR x DR x UFG	2441.8690	56	43.6048	0.2473	1.000	-0.0067
10	Residual	916804.9079	5200	176.3086			

Table 28. ANOVA analysis based on the prec metric for Equation 19, on TREC DD 2016 collection. Redundant factors in previous models are omitted for brevity. .

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	IR x DR	0.0000	8	0.0000	0.0000	1.000	-0.0013
1	IR x UFG	0.0000	28	0.0000	0.0000	1.000	-0.0044
1	DR x UFG	0.5610	14	0.0401	0.3344	0.990	-0.0015
1	IR x DR x UFG	1.6205	56	0.0289	0.2415	1.000	-0.0067
1	Residual	623.1746	5200	0.1198			
2	IR x DR	1.1713	8	0.1464	0.4766	0.874	-0.0007
2	IR x UFG	0.8754	28	0.0313	0.1018	1.000	-0.0040
2	DR x UFG	1.9711	14	0.1408	0.4583	0.955	-0.0012
2	IR x DR x UFG	5.6501	56	0.1009	0.3284	1.000	-0.0059
2	Residual	1597.3815	5200	0.3072			
5	IR x DR	8.4212	8	1.0526	1.1967	0.297	0.0002
5	IR x UFG	0.9287	28	0.0332	0.0377	1.000	-0.0043
5	DR x UFG	11.8830	14	0.8488	0.9649	0.487	-0.0001
5	IR x DR x UFG	17.0248	56	0.3040	0.3456	1.000	-0.0058
5	Residual	4574.1837	5200	0.8797			
10	IR x DR	13.9191	8	1.7399	1.0218	0.417	0.0000
10	IR x UFG	1.1492	28	0.0410	0.0241	1.000	-0.0043
10	DR x UFG	11.9941	14	0.8567	0.5031	0.933	-0.0011
10	IR x DR x UFG	22.4590	56	0.4011	0.2355	1.000	-0.0068
10	Residual	8854.5760	5200	1.7028			

Table 29. ANOVA analysis based on the st-rec metric for Equation 19, on TREC DD 2016 collection. Redundant factors in previous models are omitted for brevity. .

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	IR x DR	0.0000	8	0.0000	0.0000	1.000	-0.0013
1	IR x UFG	0.0000	28	0.0000	0.0000	1.000	-0.0044
1	DR x UFG	7.6901	14	0.5493	0.3643	0.984	-0.0014
1	IR x DR x UFG	32.2662	56	0.5762	0.3822	1.000	-0.0055
1	Residual	7839.9340	5200	1.5077			
2	IR x DR	5.0644	8	0.6330	0.2373	0.984	-0.0010
2	IR x UFG	1.9787	28	0.0707	0.0265	1.000	-0.0043
2	DR x UFG	14.1445	14	1.0103	0.3788	0.981	-0.0014
2	IR x DR x UFG	28.4629	56	0.5083	0.1905	1.000	-0.0072
2	Residual	13870.5876	5200	2.6674			
5	IR x DR	17.0097	8	2.1262	0.6575	0.729	-0.0004
5	IR x UFG	16.6634	28	0.5951	0.1840	1.000	-0.0036
5	DR x UFG	17.9841	14	1.2846	0.3973	0.976	-0.0013
5	IR x DR x UFG	86.4810	56	1.5443	0.4776	1.000	-0.0046
5	Residual	16814.9022	5200	3.2336			
10	IR x DR	9.6834	8	1.2104	0.3800	0.932	-0.0008
10	IR x UFG	3.6348	28	0.1298	0.0408	1.000	-0.0042
10	DR x UFG	22.9221	14	1.6373	0.5141	0.927	-0.0011
10	IR x DR x UFG	89.9015	56	1.6054	0.5040	0.999	-0.0044
10	Residual	16561.9726	5200	3.1850			

Table 30. ANOVA analysis based on the nCT metric for Equation 19, on TREC DD 2017 collection. Redundant factors in previous models are omitted for brevity. .

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	IR x DR	0.0000	8	0.0000	0.0000	1.000	-0.0011
1	IR x UFG	0.0000	28	0.0000	0.0000	1.000	-0.0039
1	DR x UFG	37.5318	14	2.6808	0.2489	0.998	-0.0015
1	IR x DR x UFG	148.9421	56	2.6597	0.2470	1.000	-0.0059
1	Residual	63535.2846	5900	10.7687			
2	IR x DR	237.0307	8	29.6288	0.3272	0.956	-0.0007
2	IR x UFG	77.5135	28	2.7683	0.0306	1.000	-0.0038
2	DR x UFG	281.9676	14	20.1405	0.2224	0.999	-0.0015
2	IR x DR x UFG	1425.6938	56	25.4588	0.2812	1.000	-0.0056
2	Residual	534226.8406	5900	90.5469			
5	IR x DR	384.4718	8	48.0590	0.2257	0.986	-0.0009
5	IR x UFG	330.9131	28	11.8183	0.0555	1.000	-0.0037
5	DR x UFG	2004.7351	14	143.1954	0.6724	0.804	-0.0006
5	IR x DR x UFG	3586.4407	56	64.0436	0.3007	1.000	-0.0055
5	Residual	1256502.5475	5900	212.9665			
10	IR x DR	959.3064	8	119.9133	0.4328	0.902	-0.0006
10	IR x UFG	530.2464	28	18.9374	0.0683	1.000	-0.0036
10	DR x UFG	3574.2195	14	255.3014	0.9214	0.535	-0.0002
10	IR x DR x UFG	4863.9376	56	86.8560	0.3135	1.000	-0.0054
10	Residual	1634757.5464	5900	277.0776			

Table 31. ANOVA analysis based on the α-nDCG metric for Equation 19, on TREC DD 2017 collection. Redundant factors in previous models are omitted for brevity. .

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	IR x DR	0.0000	8	0.0000	0.0000	1.000	-0.0011
1	IR x UFG	0.0000	28	0.0000	0.0000	1.000	-0.0039
1	DR x UFG	62.2907	14	4.4493	0.2969	0.994	-0.0014
1	IR x DR x UFG	214.0923	56	3.8231	0.2551	1.000	-0.0058
1	Residual	88404.8124	5900	14.9839			
2	IR x DR	205.8870	8	25.7359	0.1319	0.998	-0.0010
2	IR x UFG	75.4253	28	2.6938	0.0138	1.000	-0.0038
2	DR x UFG	286.6464	14	20.4747	0.1050	1.000	-0.0017
2	IR x DR x UFG	1401.0110	56	25.0181	0.1282	1.000	-0.0068
2	Residual	1150948.3659	5900	195.0760			
5	IR x DR	366.9551	8	45.8694	0.1551	0.996	-0.0009
5	IR x UFG	242.9576	28	8.6771	0.0293	1.000	-0.0038
5	DR x UFG	3558.8050	14	254.2004	0.8597	0.603	-0.0003
5	IR x DR x UFG	4600.0007	56	82.1429	0.2778	1.000	-0.0056
5	Residual	1744640.9350	5900	295.7019			
10	IR x DR	457.1985	8	57.1498	0.1863	0.993	-0.0009
10	IR x UFG	213.6344	28	7.6298	0.0249	1.000	-0.0038
10	DR x UFG	917.9309	14	65.5665	0.2137	0.999	-0.0015
10	IR x DR x UFG	6419.5493	56	114.6348	0.3737	1.000	-0.0049
10	Residual	1810057.1915	5900	306.7894			

Table 32. ANOVA analysis based on the prec metric for Equation 19, on TREC DD 2017 collection. Redundant factors in previous models are omitted for brevity.

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	IR x DR	0.0000	8	0.0000	0.0000	1.000	-0.0011
1	IR x UFG	0.0000	28	0.0000	0.0000	1.000	-0.0039
1	DR x UFG	0.0000	14	0.0000	0.0000	1.000	-0.0019
1	IR x DR x UFG	0.0000	56	0.0000	0.0000	1.000	-0.0078
1	Residual	728.3360	7080	0.1029			
2	IR x DR	0.8874	8	0.1109	1.3526	0.212	0.0004
2	IR x UFG	0.4892	28	0.0175	0.2130	1.000	-0.0031
2	DR x UFG	0.8967	14	0.0640	0.7809	0.691	-0.0004
2	IR x DR x UFG	0.6094	56	0.0109	0.1327	1.000	-0.0068
2	Residual	580.6732	7080	0.0820			
5	IR x DR	1.4902	8	0.1863	3.6467	0.000	0.0029
5	IR x UFG	0.4834	28	0.0173	0.3380	1.000	-0.0026
5	DR x UFG	0.7888	14	0.0563	1.1031	0.349	0.0002
5	IR x DR x UFG	0.8131	56	0.0145	0.2843	1.000	-0.0056
5	Residual	361.6446	7080	0.0511			
10	IR x DR	1.2944	8	0.1618	4.2458	0.000	0.0036
10	IR x UFG	0.2656	28	0.0095	0.2489	1.000	-0.0029
10	DR x UFG	0.7741	14	0.0553	1.4509	0.121	0.0009
10	IR x DR x UFG	0.5134	56	0.0092	0.2406	1.000	-0.0059
10	Residual	269.8111	7080	0.0381			

Table 33. ANOVA analysis based on the st-rec metric for Equation 19, on TREC DD 2017 collection. Redundant factors in previous models are omitted for brevity.

Iter	Effects	SS	DF	MS	F	p-value	ω^2_{fact}
1	IR x DR	0.0000	8	0.0000	0.0000	1.000	-0.0011
1	IR x UFG	0.0000	28	0.0000	0.0000	1.000	-0.0039
1	DR x UFG	0.0000	14	0.0000	0.0000	1.000	-0.0019
1	IR x DR x UFG	0.0000	56	0.0000	0.0000	1.000	-0.0078
1	Residual	850.6611	7080	0.1201			
2	IR x DR	0.2376	8	0.0297	0.3147	0.961	-0.0008
2	IR x UFG	0.3591	28	0.0128	0.1359	1.000	-0.0034
2	DR x UFG	0.0907	14	0.0065	0.0686	1.000	-0.0018
2	IR x DR x UFG	0.4751	56	0.0085	0.0899	1.000	-0.0071
2	Residual	668.2184	7080	0.0944			
5	IR x DR	0.3961	8	0.0495	1.0252	0.414	0.0000
5	IR x UFG	0.2151	28	0.0077	0.1591	1.000	-0.0033
5	DR x UFG	0.1629	14	0.0116	0.2409	0.998	-0.0015
5	IR x DR x UFG	0.3226	56	0.0058	0.1193	1.000	-0.0069
5	Residual	341.9116	7080	0.0483			
10	IR x DR	0.2528	8	0.0316	1.1836	0.305	0.0002
10	IR x UFG	0.1713	28	0.0061	0.2292	1.000	-0.0030
10	DR x UFG	0.2049	14	0.0146	0.5483	0.905	-0.0009
10	IR x DR x UFG	0.2238	56	0.0040	0.1497	1.000	-0.0067
10	Residual	189.0210	7080	0.0267			