Modelling and Explaining IR System Performance Towards Predictive Evaluation

Guglielmo Faggioli

Supervisor: Prof. Nicola Ferro

Department of Information Engineering
University of Padua

This dissertation is submitted for the degree of
Doctor of Philosophy

September 2022

To my Family.

Acknowledgements

First, I would like to acknowledge my Supervisor Prof. Nicola Ferro without whom nothing of this would have been possible. Nicola guided me e throughout these years, providing me countless opportunities of professional growth. I will treasure his valuable teachings for the time coming.

I thank the Reviewers Prof. Fabio Crestani from Università della Svizzera Italiana and Prof. Salvatore Orlando from Università Ca' Foscari Venezia for their precious suggestions and indications that allowed me to further improve this manuscript.

I thank the senior members of the IMS group Prof. Maristella Agosti, Prof. Giorgio Maria Di Nunzio and Prof. Gianmaria Silvello for being steady guides and constant sources of inspiration.

I would like to acknowledge my colleagues Dr. Dennis Dosso, Dr. Stefano Marchesin, Dr. Federica Vezzani, Dr. Alberto Purpura, Alessandro Fabris, Fabio Giachelle, Ornella Irrera, Laura Menotti for the motivation, suggestions, and all the opportunities for discussion.

I acknowledge my huge extended family for their patience, support and company: to Alessia, Elia, Martina, Ana, Eugenio, Ilenia. A special thanks to my friends, who lighten my life and make difficult days less difficult. I would not be where I am today without the incredible support, encouragement, and love of my parents Valeria and Franco, my brother, Raimondo and all my family.

Thanks to Vi for giving me your hand during this long journey. For your kindness and understanding while being at my side all the time. You made sure I never forgot I was able to make it. This paragraph is not enough to express my gratitude for the support and love I received from all of them.

Abstract

Information Retrieval (IR) systems play a fundamental role in many modern commodities, including Search Engines (SE), digital libraries, recommender systems and social networks. The IR task is particularly challenging because of the volatility of IR systems performance: users' information needs change daily, and so do the documents to be retrieved and the concept of what is relevant to a given information need. Nevertheless, the empirical evaluation of an IR system is a costly and slow post-hoc procedure, that happens after the system deployment. Given the challenges linked to empirical IR evaluation, predicting a system's performance before its deployment, would add significant value to the development of an IR system. In this manuscript, we place the cornerstone for the prediction of IR performance, by considering two closely related areas: the modeling of IR systems performance and the Query Performance Prediction (QPP). The former area allows us to identify those features that impact the most on the performance and that can be used as predictors, while the latter provides us with a starting point to instantiate the predictive task in IR.

Concerning the modeling of IR performance, we first investigate one of the most popular statistical tools, ANOVA, by comparing the traditional ANOVA with a recent approach, bootstrap ANOVA. Secondly, using ANOVA, we study the concept of topic difficulty and observe that the topic difficulty is not an intrinsic property of the information need, but it stems from the formulation used to represent the topic. Finally, we show how to use Generalized Linear Models as an alternative to the traditional linear modeling of IR performance. We show how GLMs provide more powerful inference, with comparable stability.

Our analyses on the QPP domain start with developing a predictor used to select among a set of reformulations for the same information need, the best performing one for the systematic review task. Secondly, we investigate how to classify queries as either semantic or lexical to predict whether neural models will perform better than lexical ones. Finally, given the challenges shown in the evaluation of the previous approaches, we devise a new evaluation procedure, dubbed sMARE. sMARE allows moving from single point estimation of the performance, to a distributional one, allowing to achieve improved comparisons between QPP models and more precise analyses.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

> Everyone has the right to freedom of opinion and expression; this right includes freedom to hold opinions without interference and to seek, receive and impart information and ideas through any media and regardless of frontiers.
>
> Universal Declaration of Human Rights, *Article 19*

Information Retrieval is the activity of finding and retrieving documents from a corpus to satisfy a user's information need expressed under the form of a query. Techniques developed in the IR environment are essential to several neighbouring areas, such as search engines, digital libraries, web search, product search and recommendation, and social media. Because of the pervasiveness of the IR in modern information access systems, there is a strong economical interest in constantly improving the quality of the IR system. Thanks to the maturity of the field, large part of the research community, both in academia and industry, have understood the importance of a sound, theoretically well founded, and empirically well executed evaluation of the IR systems. The evaluation of an IR system in the offline scenario typically employs an evaluation or test collections that follows the Cranfield paradigm – more on this in Chapter 2.

The main limitations in evaluating IR Systems stem from its empirical nature. The fact that the IR evaluation is so strongly rooted in empirical evaluation, makes this field peculiar with respect to many other engineering fields. In other scenarios it is common to have mathematical models capable of describing the expected performance of a system, prior to its deployment. For example, it is possible to compute the maximum load that a bridge can hold. Similarly, we are capable of estimating the resources that we should allocate for a computer network to handle a certain quantity of traffic.

Several characteristics make the modeling of the performance in IR a challenging task. In particular:

- Topical relevance and user relevance: in many scenarios, the relevance of document depends on the context. For example, if we consider the news domain, the relevance is often coupled with the freshness of news: even though a news is highly on-topic with the information need of the user, if it concerns old events it might lose relevance rapidly. Similarly, in a legal documents retrieval domain, a document regarding a certain sentence might lose relevance if new laws are enacted. In this sense, a document relevant to a topic (topical relevance) might not be relevant to the user as well (user relevance).

- Semantic gap: it represents the difference between the user's and machine's representation of a content. The semantic gap is linked to linguistic features of the terms used in a document or a query, such as synonymy (different terms having the same meaning) and polysemy (same term with multiple meanings, depending on the context).

With the research line proposed in this manuscript, we plan to set the cornerstone for the prediction of performance in the IR evaluation. If we could predict a system's performance before deployment, we could reduce the need for experimental collections.

With such overall objective in mind, we can identify two main subgoals of this manuscript.

- **Modeling IR performance**: Investigate the modeling strategies currently adopted to model the performance of an IR system to identify which tools are the best suited and what is the domain in which a predictive model can be used;

- **Predicting IR performance**: Investigate the currently available tools in the IR performance prediction domain, to extend them.

We list here the main contributions that result as output of the study of each subgoal.

1.1 Modeling Information Retrieval Performance

In investigating the first subgoal, we identify three main relevant aspects of the currently adopted approach to performance modeling that can contribute in understanding how to predict the performance of an IR system: *i)* the study of ANOVA and its reproduciblity; *ii)* the study of the impact of different formulations on the performance of a system and their effect on the concept of topic difficulty; *iii)* the study of the Generalized Linear Models to improve the fitness of our evaluation models to information retrieval data.

The ultimate goal of the evaluation is to understand when two IR systems are (significantly) different. To this end, many comparison procedures have been developed over time.

To investigate the first aspect of the first subgoal, we focus on methods based on ANOVA, which explicitly model the data in terms of different contributing effects, allowing us to obtain a more accurate estimate of significant differences. In this context, recent studies have shown how sharding the corpus can further improve the estimation of the system effect. As the first contribution, we replicated and compared methods based on "traditional" ANOVA (tANOVA) to those based on a bootstrapped version of ANOVA (bANOVA) and those performing multiple comparisons relying on a more conservative Family-wise Error Rate (FWER) controlling approach to those relying on a more lenient False Discovery Rate (FDR) controlling approach. We found that bANOVA shows overall a good degree of reproducibility, with some limitations regarding the confidence intervals. Besides, compared to the tANOVA approaches, bANOVA presents greater statistical power, at the cost of lower stability.

In the TREC-based evaluation paradigm, it is common to consider the information need as represented by a single query. In reality, users routinely reformulate queries to satisfy an information need. In this sense, the notion of "query variations" corresponds to the multiple user formulations for an information need. Like many retrieval models, some queries are highly effective while others are not. This is often an artefact of the collection being searched which might be more or less sensitive to word choice. Users rarely have perfect knowledge about the underlying collection, and so finding queries that work is often a trial-and-error process. To address the second sub-aspect of the first goal, we explore the fundamental problem of system interaction effects between collections, ranking models, and queries. We formalize it using ANOVA models to measure multiple component effects across collections and topics by nesting multiple query variations within each topic. Our contribution is to show that query formulations have a comparable effect size of the topic factor itself, which is known to be the factor with the greatest effect size in prior ANOVA studies. Both topic and formulation have a substantially larger effect size than any other factor, including the ranking algorithms and, surprisingly, even query expansion. This finding reinforces the importance of further research in understanding the role of query rewriting in IR-related tasks.

Linear models (e.g., t-test and ANOVA) play a pivotal role in modelling IR performance. Linear models rely on assumptions that IR experimental observations rarely meet, e.g. about the normality of the data or the linearity itself. Even though linear models are, in general, resilient to violations of their assumptions, departing from them might reduce the effectiveness of the tests. Hence, as a final contribution to the first subgoal, we investigate the use of the Generalized Linear Model (GLM) framework, a generalization of the traditional linear modelling that relaxes assumptions about the distribution and the shape of the models. We discuss how GLMs work and how they can be applied in the context of IR evaluation. In particular, we focus on the *link function* used to build GLMs, which allows for the model to have non-linear shapes. We conduct thorough experimentation using two TREC collections and several evaluation measures. Overall, we show how the log and logit links can identify more and more consistent significant differences, up to 25% more when using 50 topics, than the identity link used today and with a comparable, or slightly better, risk of publication bias.

1.2 Predicting Information Retrieval Performance

Our investigation of the second subgoal, the study of predictive tools, is also divided into three main research paths to which we provided our contribution: *i)* the study of a new QPP model in the systematic review domain that exploits the knowledge we gained when measuring the impact of multiple query formulations on IR systems; *ii)* the development of a machine-learning based approach to predict if either lexical or semantic systems perform better on a given query to carry out model selection; *iii)* the analysis of the current approach used to evaluate QPP models.

Evidence-based healthcare integrates the best research evidence with clinical expertise in order to make decisions based on the best practices available. In this context, the task of collecting all the relevant information, a recall-oriented task, in order to make the right decision within a reasonable time frame has become an important issue. We investigate the problem of building effective Consumer Health Search (CHS) systems that use query variations to achieve high recall and fulfil the information needs of health consumers. In particular, we study an intent-aware gain metric used to estimate the amount of missing information and make a prediction about the achievable recall for each query reformulation during a search session. We evaluate and propose alternative formulations of this metric using standard test collections of the CLEF 2018 eHealth Evaluation Lab CHS.

Traditional Information Retrieval (IR) models, also known as lexical models, are hindered by the semantic gap, which refers to the mismatch between different representations of the same underlying concept. To address this gap, semantic models have been developed. Semantic and lexical models exploit complementary signals that are best suited for different types of queries. For this reason, these model categories should not be used interchangeably, but should rather be properly alternated depending on the query. Therefore, it is important to identify queries where the semantic gap is prominent and thus semantic models prove effective. In this regard, as a contribution to the study of the second subgoal, we quantify the impact of using semantic or lexical models on different queries, and we show that the interaction between queries and model categories is large. Then, we propose a labeling strategy to classify queries into semantically hard or easy, and we deploy a prototype classifier to discriminate between them.

The development of the approaches mentioned above highlighted an important, often underlooked, challenge linked to the QPP scenario: its evaluation. Therefore, as a contribution to the final research path of the second subgoal, we re-examine the existing evaluation methodology commonly used for QPP, and propose a new approach. Our key idea is to model QPP performance as a distribution instead of relying on point estimates. We demonstrate important statistical implications, and show how to overcome key limitations imposed by the currently used correlation-based point-estimate evaluation approaches. We also explore the potential benefits of using multiple query formulations and ANOVA modelling in order to measure interactions between multiple factors. The resulting statistical analysis combined with a novel evaluation framework demonstrates the merits of modelling QPP performance as distributions and enables detailed statistical ANOVA models for comparative analyses to be created.

1.3 Outline

Each chapter of this manuscript follows a self-contained structure to ease its readability and improve its modularity. In each chapter, we detail the rationale and motivations that pushed us in investigating a specific domain, the experimental approach followed, the empirical results attained and the conclusive remarks that highlight how our findings help improve the current state of the art. This thesis is outlined as follows. Chapter 2 details the theoretical context underlying this manuscript. We first introduce linear models and how they are used in the context of the IR evaluation. We detail how the fitness of linear models to the data can be further improved via Generalized Linear Models (GLMs). We then describe the main strategies used to model different aspects related to the IR scenario. Namely, we detail the modeling of topic difficulty, the simulation of IR experimental data and the query representation. Finally, we describe what is the role of predictive models in the context of the IR evaluation. In particular, we focus on the QPP framework and the model selection task. The first part of this book, Part I, contains our investigation in the domain of the IR performance modeling. In Chapter 3 we analyze one of the most commonly used tools in IR evaluation: ANOVA. We compare two well-known approaches to ANOVA, with the objective of understanding which desirable properties ANOVA enjoys in our specific experimental settings. After that, in Chapter 4, we exploit ANOVA to increase our knowledge of the concept of "Topic difficulty". Finally, in Chapter 5, we investigate what are the benefits of employing GLMs in the context of the IR performance modeling to define models that better fit our data to improve how we compare systems and, in general, improve how we model IR empirical data. Part II illustrates our analysis in the domain of predictive models in the IR scenario. We initiate our analysis in the domain of predictive models, by considering in Chapter 6, the role played by query formulations in predicting the recall achieved by different systems in the systematic reviews domain. We prosecute in Chapter 7, where we investigate the possibility of understanding whether lexical models or semantic ones will perform better on a given query. Motivated by the challenges represented by the evaluation of QPP models in the previous two chapters, we conclude our analysis in Chapter 8, where we analyze sMARE, a new evaluation approach to QPP.

Finally, Chapter 9 reports our conclusive remarks, describing the main findings achieved in each chapter and giving an overall view on how such findings can be used to enable performance prediction in IR. We conclude Chapter 9 by drawing the main research paths enabled by the findings detailed in this manuscript.

Chapter 2

Background

Due to the transversal nature of the performance modelling and prediction tasks in IR, this manuscript draws its background from several areas linked to the IR systems evaluation.

We start our overview of the background considering the statistical modeling tools typically used in IR evaluation, among which ANOVA is the most prominent, from a general standpoint.

We survey the main works concerning the statistical evaluation of IR models, and we consider past endeavours in system component analysis, detailing how previous works intended and addressed the challenge of describing the systems' performance. we further describe some non-linear modelling approaches to the system performance in IR.

Finally, we survey the primary efforts made in the query performance prediction and model selection fields moving toward predictive models.

2.1 Statistical Background and Tools

2.1.1 ANOVA

A General Linear Mixed Model (GLMM) [107, 140] models variations of a dependent variable ("Data") w.r.t. a controlled variation of independent variables ("Model"), in addition to a residual uncontrolled variation ("Error"):

$$Data = Model + Error$$

The most basic example of GLMM is a simple linear regression, where

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$$

Table 2.1 Representation of the typical experimental IR data: a set of systems s_1, ..., s_n is used to retrieve documents in response to a set of information needs t_1,..., t_m. For each pair of system, topic, we can compute a measure of performance $y_{i,j}$. $\mu_{i.}$ represents the average performance observed over the i-th topic, while $\mu_{.j}$ is the mean performance for system j.

System Factor

		s_1	s_2	$\dots$	s_n	
Topic Factor	t_1	$y_{1,1}$	$y_{1,2}$	$\dots$	$y_{1,n}$	$\mu_{1.}$
	t_2	$y_{2,1}$	$\ddots$		$\vdots$	$\mu_{2.}$
	$\vdots$	$\vdots$				$\vdots$
	t_m	$y_{m,1}$	$\dots$		$y_{m,n}$	$\mu_{m.}$
		$\mu_{.1}$	$\mu_{.2}$	$\dots$	$\mu_{.n}$	$\mu_{..}$

The dependent variable Y_i, representing the score of the i-th subject, is explained (predicted) in terms of an intercept β_0 and an independent variable X_i (predictor) times the regression coefficient β_1, the slope of the regression line, plus a residual error ε_i, not explained by the model, which follows a zero-mean Gaussian distribution.

ANalysis Of VAriance (ANOVA), when viewed as a General Linear Mixed Model (GLMM), attempts to explain data (the dependent variable scores) in terms of the experimental conditions (the model) and an error component. Typically, ANOVA is used to determine under which experimental condition the dependent variable score means differ. In other terms, called μ_i the means for the different experimental conditions, ANOVA tests the following system of hypothesis:

$$H_0 : \mu_0 = \mu_1 = ... = \mu_n$$

$$H_1 : \text{Means are not all equal}$$

Additionally, ANOVA allows determining what proportion of the variance observed for a dependent variable can be attributed to differences between specific experimental groups or conditions, as defined by the independent variable(s) being modelled. An ANOVA can be regarded as a type of regression analysis using only categorical predictors.

The regression model described above is expressed in ANOVA terms as:

$$y_{ij} = \mu + \alpha_j + \varepsilon_{ij},$$

where Y_{ij} is the i-th dependent variable subject score in the j-th experimental condition. The parameter μ is the grand mean of the experimental condition population means underlying all of the dependent variable scores of the subjects. The parameter α_j is the effect of the j-th experimental condition, and the random variable ε_{ij} is the error, which reflects any variance caused by an undefined source. The above regression model corresponds to the ANOVA version once you add as many X_{ij} predictors and as many levels as there are in the experimental condition α_j.

Parameters Estimation

Assume to have a set $\mathscr{S}$ of systems such that $|\mathscr{S}| = n$ and a set of information needs $\mathscr{T}$ with cardinality $|\mathscr{T}| = m$. Table 2.1 illustrates the typical shape of experimental IR data, organized in a tabular form, with one row for each topic and a column for each system. In the aforementioned scenario, the most common evaluation model used in IR is the two-ways ANOVA that models separately the effect of the topic and the system and can be represented using the following equation:

$$y_{ij} = \mu_{..} + \tau_i + \alpha_j + \varepsilon_{ij}$$

The grand mean $\mu_{..}$ corresponds to the intercept of the linear model, and its Ordinary Least Squares (OLS) estimator is the following:

$$\mu_{..} = \frac{1}{n \cdot m} \sum_{i=1}^{n} \sum_{j=1}^{m} y_{ij}$$

Called $\mu_{i.}$ and $\mu_{.j}$ the average performance over all systems for topic i and the mean performance over all topics for system j, the OLS estimator for the effect of the i-th topic is computed as:

$$\tau_i = \mu_{i.} - \mu_{..} = \frac{i}{n} \sum_{j=1}^{n} y_{ij} - \hat{\mu}_{..}$$

Similarly, the OLS estimator for the j-th system si given by:

$$\tau_j = \mu_{.j} - \mu_{..} = \frac{i}{m} \sum_{i=1}^{m} y_{ij} - \hat{\mu}_{..}$$

Finally, the error is computed as:

$$\varepsilon_{ij} = y_{ij} - \hat{y}_{ij} = y_{ij} - \mu_{..} - \tau_i - \alpha_j$$

Using aforementioned estimators, we can compute the Sum of Squares (SS) for the different factors. In the case of the topics, the SS$_t opic$ is defined as follows:

$$SS_{topic} = \sum_{j=1}^{n} \sum_{i=1}^{m} (\mu_{i.} - \mu_{..})^2 = \sum_{i=1}^{m} n(\mu_{i.} - \mu_{..})^2 = \sum_{i=1}^{m} n\tau_i^2$$

For what concerns the the systems we have that:

$$SS_{system} = \sum_{j=1}^{n} \sum_{i=1}^{m} (\mu_{.j} - \mu_{..})^2 = \sum_{j=1}^{n} m(\mu_{.j} - \mu_{..})^2 = \sum_{j=1}^{n} m\alpha_j^2$$

The SS of the error is computed using the following equation:

$$SS_{error} = \sum_{j=1}^{n} \sum_{i=1}^{m} (y_{ij} - \hat{y}_{ij})^2$$

In this context, the Degrees of Freedom (DF) are defined as the number of values that are free to vary, without changing the parameters estimation. Concerning topics and systems, the DF are respectively $DF_{topic} = m - 1$ and $DF_{system} = n - 1$. DF for the error are computed as $DF_{error} = (n-1) \cdot (m-1) \cdot k$, where k represents the number of replicates for each experimental condition (one in this case where each query is issued against each system once).

The Mean Squares (MS) for each factor – and for the error – is computed as:

$$MS_{factor} = \frac{SS_{factor}}{DF_{factor}}$$

Finally, the F statistics is computed as:

$$F_{factor} = \frac{MS_{factor}}{MS_{error}}$$

The F statistic is then compared with reference tables and used to carry out the test of hypotheses.

Interactions

Interactions are an additional focal element in many experimental settings since different experimental components of an might interact with each other. This is particularly true in IR where different system components often interact with each other. Observing an interaction means that a system's part might perform better (or worse) than its average, depending on the other components included in the pipeline. To be able to mathematically compute the interaction effect, we need multiple replicates of the same experiment. Even though replicates are common in other scenarios – such as social, biological or environmental sciences, they are not so common in classical IR evaluation based on the Cranfield paradigm. In fact, in the most common situation, the single experiment – trying to answer a query using a system – can be repeated only once. Therefore, several approaches have been proposed to obtain the required replicates. Among the most prominent efforts, we list corpus sharding, the use of multiple assessors, query formulations or the use of simulations. If the replicates are available then we can fit the following ANOVA model:

$$y_{ijk} = \mu_{...} + \tau_i + \alpha_j + \tau\alpha_{ij} + \varepsilon_{ijk},$$

where k indicates the k-th replicate of the experiment involving i-th topic and j-th system and $\tau\alpha_{ij}$ is the interaction between topic i and system j. The magnitude of the interaction indicates how much the two elements synergize (or discord). A highly positive interaction suggests that the topic is particularly suited for the system (and vice versa). Conversely, observing negative interaction highlights that the system works particularly poorly on that topic. Finally, close-to-zero interaction indicates no particular advantage (or disadvantage) in using system j to retrieve documents for topic i. Calling $\mu_{ij.}$ the average performance observed for the i-th topic, using the j-th system, the OLS estimate of the interaction is:

$$\tau\alpha_{ij} = \mu_{ij.} - (\mu_{...} + \tau_i + \alpha_j)$$

DF for the interaction factor are computed as $DF_{topic,system} = (n-1) \cdot (m-1)$.

For a given model, the ANOVA table summarizes the outcomes of the ANOVA test indicating, for each factor, the SS, the DF, the MS, the F statistics, and the p-value of that factor, which allows us to determine the significance of that factor. For a detailed description of how to estimate GLMM model parameters and assess their statistical significance using ANOVA, refer to Maxwell and Delaney [107], Rutherford [140].

Design of Experiments

Independent variables can be *fixed effects* or *random effects*. A fixed effect has precisely defined levels, and inferences about its effect apply only to those levels. Random effects describe a random and independently drawn set of levels representing variation for a wider population. The latter case corresponds to a more sophisticated model that, in estimating the variance attributed to the different factors, accounts for the additional randomness due to sampling of effect levels. The experimental design determines how to compute the model and estimate the parameters. In particular, it is possible to have an *independent measures* design where different subjects participate under different experimental conditions (factors) or a *repeated measures* design, where each subject participates in all of the experimental conditions (factors).

A final distinction is between *crossed/factorial* designs, where every level of one factor is measured in combination with every level of the other factors, and *nested* designs, where levels of a factor are grouped within each level of another nesting factor. In the IR experimental scenario, several aspects need to be modelled as nested factors. For example, Culpepper et al. [43] treat different formulations as a nested factor inside the topic they represent, while Faggioli et al. [52] use a nested design to represent random permutations of a given conversation in the conversational search domain.

Effect Size

Since the F statistic tends to increase and the p-value tends to decrease as the sample size increases, the *effect size* of a factor is used, and quantifies the magnitude of the variance observed in the model using an unbiased estimator [120, 142]:

$$\hat{\omega}^2_{\langle fact \rangle} = \frac{\mathrm{df}_{fact}(\mathrm{F}_{fact} - 1)}{\mathrm{df}_{fact}(\mathrm{F}_{fact} - 1) + N} \tag{2.1}$$

where F_{fact} is the F-statistic and df_{fact} are the degrees of freedom for the factor while N is the total number of samples. In this way, we are able to assess not only if a factor is significant but its contribution as well. The common rule of thumb [140] when classifying $\hat{\omega}^2_{\langle fact \rangle}$ effect size is: 0.14 and above is a *large-size effect*, 0.06–0.14 is a *medium-size effect*, and 0.01–0.06 is a *small-size effect*. Note, $\hat{\omega}^2_{\langle fact \rangle}$ can be negative; in such cases it has no contribution.

Multiple Comparison Correction Procedures

When conducting tests of hypotheses, there are two main classes of errors that might verify. The *Type I error*, also called *False Positive*, represents the scenario in which we falsely reject a null hypothesis. The second class of errors, known as *Type II errors* or *False Negatives*, indicates the situation in which a false null hypothesis is deemed true. Traditionally, *Type I errors* are regarded as more dangerous since they underlay wrong statistical inference. On the other hand, *False Negatives* are often underlooked since they can be removed via further experimentation. The probability of committing a *Type I error* is usually controlled by fixing the confidence level α of the statistical test to 5% – or even lower if needed. Called β the probability of observing a false negative, the *Type II errors* are addressed by choosing a statistical test with high power, indicated with $1 - beta$.

The multiple comparison problem arises when several independent tests of hypotheses are carried out at once – e.g., we are interested in determining how many systems are better than a given baseline. Called α the probability of committing a *Type I error*, the probability of committing an error over c independent experiments is computed as:

$$p(|FP| > 0) = 1 - (1 - \alpha)^c$$

Such probability is equal to α only if $c = 1$ and it rapidly decreases toward 0 as c grows.

Being able to address the multiple comparisons problem and correct for the false positives is often of uttermost importance in IR evaluation settings. The large dimension of the experiments (high c) determines several false positives. To avoid reaching spurious and wrong conclusions, it is necessary to *i)* use a procedure with a global test of hypotheses, such as ANOVA; *ii)* use suitable multiple comparison correction procedures.

Several correction procedures have been adopted in IR experimental scenarios. For example, Tague-Sutcliffe and Blustein [168] used ANOVA to decompose performance into a topic and a system factor and adopted the Scheffe tests [152] to compensate for multiple comparisons.

Multiple comparisons procedures aim at controlling Family-wise Error Rate (FWER) [97] or False Discovery Rate (FDR) [20]. FWER is the probability of having at most one false positive among all rejected null hypotheses, and FWER-controlling procedures aim at keeping it equal to $1 - \alpha$.

One of the most popular FWER correction approaches is the Honestly Significant Difference (HSD) by Tukey [173]. Given $\hat{\mu}_{.u}$ and $\hat{\mu}_{.v}$ the marginal means for two different systems, the test value for the HSD is computed as:

$$|tk| = \frac{|\hat{\mu}_{.u} - \hat{\mu}_{.v}|}{\sqrt{\frac{MS_{error}}{m}}},$$

where MS_{error} is the mean square error according to the ANOVA model and m is the number of topics. This test value is compared against the critical value, obtained from $Q^{\alpha}_{n,df_{error}}$, the studentized range distribution [119], where n is the number of systems. Conversely, FDR-controlling procedures aim at keeping the false discovery rate (the number of false findings overall findings) at level α: this corresponds to allowing the number of false positives to increase, as long as the number of true discoveries increases. One of the most prominent FDR-controlling procedures is the Benjamini Hochberg (BH) [20] procedure. It sorts in ascending order the p-values associated with N tested hypotheses. The greatest value of k for which $p_{(k)} \leq \alpha \frac{k}{N}$ is then found: null hypotheses associated to p-values in ranks from 0 to k are rejected.

Assumptions

Linear models are based on the following assumptions [93]:

- Normality of the error terms;
- Equal variance (homoskedasticity) of the error terms;
- Independence of the observations, i.e. they independent and identically distributed observation.

A fourth often under-looked assumption concerns the relation between the explanatory variables and the explained one, which is assumed to be linear. Such assumptions are fundamental to be able to treat the model fitting computationally.

In general, linear models tend to be resilient to departures from assumptions, especially for what concern normality and homoskedasticity [80, 150, 168, 79, 37]. Nevertheless, the more the data comply with assumptions, the better the model is, with increased predictive and descriptive power.

Among Linear Models assumptions, ANOVA is known to be quite robust to violation of normality and homoskedasticity assumptions. Ito [80, p. 205] observes that the F-test is remarkably insensitive to non-normality. Similarly, Mendenhall and Sincich [108] note that, for relatively large samples (e.g. 20 or more observations per factor) ANOVA is robust to violations of the normality assumption and that it is also robust to differences in variances when using a balanced design. In commonly occurring cases where the group sample sizes are equal, it is insensitive to the heterogeneity of variance across groups. On the other hand, violation of the independence assumption can severely impact the F-test and any subsequent conclusions. This issue is discussed in detail, for example, by Scariano and Davenport [150].

IR performance scores are known to violate both normality and homoskedasticity assumptions [168, 31]. Tague-Sutcliffe and Blustein [168] noted that performance scores did not satisfy the homoskedasticity assumption and applied a transformation, which is typically used in the case of ratio data, consisting of taking the arcsine of the square root of the original scores. However, they only observed minor differences in the analysis conducted on the transformed data and ultimately decided to continue using the untransformed scores, which are easier to interpret. Carterette [31] observed that both of the first two assumptions are commonly violated, and that performance scores are typically bounded between [0,1]. However, Carterette [31] concluded that ANOVA is nevertheless resilient to the kind of violations of normality encountered in IR performance scores and that also the violations of homoskedasticity have a limited impact, which supports the previous findings of Tague-Sutcliffe and Blustein [168].

Throughout the remainder of this work, we always select as many different experimental conditions (topics, queries, systems, etc.) as needed to induce the necessary sample size to ensure that the model is robust w.r.t. violations of the normality assumption. We also adopt always a balanced design where group sample sizes are equal, limiting the impact of any violations of homoskedasticity.

Finally, when considering violations of the independence assumption in topics that can derive from multiple formulations of the same topic, query variations can be regarded as independent samples from a universe of possible queries representing an information need. Indeed, queries were gathered independently using several hundred test subjects.

2.1.2 Generalized Linear Models (GLMs)

The concept of GLMs was formulated by Nelder and Wedderburn [118] in 1972. Since then, this framework has been studied and documented extensively [105], becoming a widely accepted standard in the statistics community. GLMs have been applied successfully in several scenarios, such as engineering, biology and medicine [117]. We describe more in detail the characteristics of this modelling tool in Section 5.2. A subclass of GLMs is the General Linear Models (GLiMs), which includes the most used and studied tools for IR evaluation, such as t-tests and ANOVAs [143, 33]. As their name suggests, GLMs are a *generalization* of the linear model, able to handle far more data-sets than its traditional counterpart with increased accuracy. More in detail, GLMs relax the assumptions underlying linear models. First of all, they relax the normality assumption by allowing the data to distribute following any distribution drawn from the exponential distributions family – of which the normal is just a representative. This also softens the homoskedasticity requirement: some exponential distributions allow for the variance to change with the mean. GLMs also relax the linearity assumption by linking the response variable to the experimental conditions using a non-linear function called *link*.

Indeed, GLMs are not an alternative to linear modelling but rather a generalization of such a statistical tool.

2.2 Information Retrieval Systems and Their Evaluation

A widely acknowledged [39] definition of Information Retrieval (IR) was provided by Gerard Salton, one of the pioneers in this field, in 1968 [145]:

> Information retrieval is a field concerned with the structure analysis, organization, storage, searching, and retrieval of information.
>
> Salton, 1968

Regardless of the huge technological advances since 1968 and the changes on how we do IR, the general definition remains true today.

As the definition suggests, IR investigates many different aspects of how we handle information. The most prominent of such aspects is the study of how we model the relevance of a document to an information need, starting from the experiments on Term Frequency (TF) from Salton itself [146], to the Inverse Document Frequency (IDF) definition proposed by Spärck Jones [166], to the probabilistic model by Robertson [132] and up to modern days solutions, based on dense indexes [193] and neural networks [111].

The advances achieved in the IR field have certainly been enabled by a sound and scientifically well-founded evaluation methodology that, in turn, allowed researchers and practitioners to make models better and better, understanding their weaknesses and strengths.

2.2.1 IR Evaluation

The need for scientifically sound and generalizable evaluation procedures of the information retrieval process is clear because to improve a system quality it necessary to measure its performance. Cyril W. Cleverdon developed his renowned Cranfield paradigm [186, 187] to address, from a scientific standpoint, the evaluation of the indexing procedures. None of the Cranfield experiments was computerized. Since, and thanks to, the Cranfield experiments that proved vital also to evaluate computerized IR systems, the Information retrieval field has grown beyond any expectation.

The evaluation of an IR system according to the Cranfield paradigm is based on an experimental collection that contains three elements:

- a corpus of documents;
- a sample of topics that represent the information needs of the users of the systems that we are evaluating;
- the relevance judgments describing what documents are relevant to each topic.

Nowadays, almost every offline IR evaluation framework is based on evaluation collections that follow the Cranfield paradigm. A further push toward the adoption of this common empirical approach to the evaluation of IR systems was given in 1992 by the first Text REtrieval Conference (TREC) [70]. Since then, TREC has evolved and, in its current form is organized into several evaluation campaigns that provide practitioners with tools and infrastructure for large-scale empirical comparable evaluation of IR systems. The objective of TREC is the development of text collections to foster IR research and guide under a shared evaluation framework. The TREC conference is sponsored by the National Institute of Standards and Technology (NIST), with an investment estimated around 30 million $ in 2010. The magnitude of such investment highlights two important aspects of the *i)* the importance given to the evaluation of the systems by the research community *ii)* its high economical cost. Such huge monetary and time expense highlights the importance of predicting the performance of a system before its full deployment – with the secondary objective of depending less on traditional collections.

Evaluating an IR system The traditional IR evaluation based on the Cranfield paradigm follows a well-defined pipeline. The IR system retrieves a ranked list of corpus documents, called run, for each topic in the collection.

Using the relevance judgments for the topic at hand, we map each document of the run into its relevance score. If the document was judged relevant, its score will be a positive value. Conversely, if the document was judged irrelevant or unjudged, then the associated score will be 0. In some cases, we might assign negative relevance scores to sensitive documents (i.e., private documents or adult contents). The sorted list of relevance judgements obtained after this process is called “judged run”.

The judged run allows computing a measure of performance. Depending on the aspects that we consider relevant for the system, the kind of topics, the users that will use it, or the domain, we might decide to include different measures to measure multiple system features.

Given a specific topic of the collection we define D the list of documents retrieved for that topic, RB the set of relevant documents, (also called the *recall base*) r_i the relevance of the i-th document of the run, $k = |D|$ the length of the run, and I the indicator function that outputs 1 if its argument is true. The main IR measures are:

precision : proportion of relevant documents retrieved over all retrieved documents:

$$p = \frac{\sum_{i=1}^{k} I(r_i > 0)}{k}$$

recall : proportion of relevant documents retrieved over all relevant documents

$$r = \frac{\sum_{i=1}^{k} I(r_i > 0)}{|RB|}$$

F1-score : harmonic mean of precision and recall

$$\text{F1-score} = \frac{2 \cdot p \cdot r}{p + r}$$

Average Precision (AP) [25]: mean of the precision, measured at different recall levels:

$$AP = \frac{\sum_{i=1}^{k} p@i}{RB},$$

where $p@i$ is the precision measured considering only the first i documents. The rationale behind AP is that it does not only consider how many relevant documents are retrieved but also how soon they are retrieved;

Normalized Discounted Cumulated Gain (nDCG) [81]: the proportion of ideal discounted cumulative gain attained by the run at hand.

$$\text{nDCG} = \frac{\text{DCG}}{\text{iDCG}},$$

where DCG is the Discounted Cumulative Gain and is measured as:

$$\text{DCG} = \sum_{i=1}^{k} \frac{r_i}{\min(1, log_b(i))},$$

while the iDCG is the *ideal* DCG: the DCG measured for a perfectly ranked run (the ideal run where the documents are sorted by relevance). Similarly to the AP, the nDCG aims at weighting the value (gain) produced by each relevant document, with the rationale that the later such documents have been retrieved, the more discounted the gain they produce.

Rank-Biased Precision (RBP) [115]: the utility accrued by a user that scans the ranked list depending on its persistence parameter $\rho \in (0,1)$. The persistence indicates how likely is for the user to continue scanning the result list with probability ρ – patient users will have higher ρ.

$$RBP = (1-\rho) \sum_{i=1}^{k} r_i * \rho^{i-1}$$

Many other measures have been proposed thorough time, depending on the task at hand and what are the most relevant aspects for the system evaluated.

Once a measure of performance has been computed for each topic, it is possible to evaluate the overall performance of a system, by averaging the performance achieved over the different information needs.

Comparing multiple IR systems Being able to assess the performance of a single system has limited utility – the real importance of the IR evaluation is linked to the capability of comparing two or more systems. By comparing multiple systems – or multiple different versions of the same system – we can understand what components or characteristics of a system allow it to perform better and how we can improve it.

One of the major challenges in this context is that the selected topics are a (limited) sample of the whole topic population – which includes all the information needs (past, preseent, and future) that we can express in a given domain.

When we carry out our experiments and observe that a system is better than another, we need to be aware that this might be simply due to the topics included in the collection: other topics might have produced opposite results. We are therefore required to use statistical inference to understand whether our empirical observations generalize to the whole population of topics. This allows us to determine if the (mean performance of) two systems are indeed Statistically Significantly Different (SSD) or the difference we observed is simply due to chance.

When it comes to carrying out statistical inference in IR to determine if two (or more) systems are SSD, several possible approaches have been proposed through time. In this regard, the Wilcoxon test [188] was one of the first to be adopted in IR. Kempthorne and Doerfler [85] showed that, while being more robust than the signed test [163], the Wilcoxon signed rank test is worse than the randomization test [51], suggesting to use of the last one. Their analysis concerned statistical aspects in general, without having the IR setting in mind. Hull [79] was one of the first authors experimenting with statistical tests in the IR setting. He argued that the t-test [167][1], being sufficiently robust to the violation of its assumption, should be the preferred test when comparing the mean for multiple IR systems. One year later, Wilbur [192] compared the non-parametric tests – Wilcoxon signed rank, sign, permutation, and bootstrap tests. Wilbur [192] observed that, in line with what was indicated by Kempthorne and Doerfler [85], the Wilcoxon and sign tests should be avoided in favour of randomization tests with bootstrap being the best. A few years later, Savoy

[1] In this regard, ANOVA, having a similar underlying rationale as the t-test, should be considered equivalent when compared to other statistical tests.

[149], by comparing the bootstrap test against the t-test, confirmed the observation of Wilbur [192], favouring the former. It was Zobel [207] who, differently from what was observed in literature until then, argued that the Wilcoxon test is more reliable and powerful than the t-test (and ANOVA). Sanderson and Zobel [148] further experimented on this topic, reaching similar conclusions as Hull [79]: the t-test should be favoured over both Wilcoxon and sign tests. Sakai [141] further arguments the advantages of the bootstrap procedure. Nevertheless, he does not compare it with other approaches. From that point on, the literature substantially agrees in considering the t-test superior to other procedures, either considering multiple topic splits such as Cormack and Lynam [38], Smucker et al. [165, 164] or via simulation-based approaches, such as the one proposed by Urbano et al. [176]. A recent exception to this is proposed by Parapar et al. [121] that, using a simulation approach, argued in favour of the Wilcoxon signed rank test over the others, similarly to Zobel [207]. A summary of what done concerning statistical testing and what conclusions have been drawn is available in Table 2.2.

With few noticeable exceptions, a large part of the literature highlights the superiority of the t-test in terms of robustness. This aspect, combined with the fact that almost all statistical packages allow for its computation, contributed to making it the most popular approach [143, 33] to the statistical inference in IR.

Table 2.2 Literature comparisons between main statistical tests used to determine if a pair of IR systems are SSD. The × symbol indicates that the test was considered by the authors but deemed unsuited to IR, the ✓ symbol indicates that the test was considered and deemed the most suited or best performing. * indicates that the comparison was done in general, without considering specifically the IR setting.

	Wilcoxon [188]	Sign [163]	Permutation [51]	Bootstrap [51]	t-test [167]
Kempthorne and Doerfler [85]*	×	×	✓		
Hull [79]	×	×			✓
Wilbur [192]	×	×	×	✓	
Savoy [149]				✓	×
Zobel [207]	✓				×
Sanderson and Zobel [148]	×	×			✓
Sakai [141]				✓	
Cormack and Lynam [38]	×	×			✓
Smucker et al. [165]	×	×	×	×	✓
Smucker et al. [164]			×	×	✓
Urbano et al. [177]	×	×	×	×	✓
Urbano et al. [176]	×	×	×	×	✓
Parapar et al. [121]	✓	×	×	×	×

Regardless of the test taken into account, it is important to compensate for the multiple comparison problem (See section 2.1.1).

2.3 Modeling IR Systems Performance: Linear Models

Linear Models are, to various extents of awareness, the *de facto* standard when it comes to model, evaluating and comparing the performance of IR systems. Indeed, linear models, of which the linear regression is just a representative, including the two well-known tests of hypotheses: t-tests and ANOVA. These evaluation techniques are the most popular evaluation approaches according to two recent surveys carried out by Sakai [143] and Carterette [33]. In this regard, a rich body of related work has explored component analysis via linear modelling to measure the effects of complex systems on retrieval effectiveness, including factors like topics, collections, and system components.

One of the first applications of the linear models to the IR evaluation is the one by Tague-Sutcliffe and Blustein [168] which adopted a two-way ANOVA model to decompose the overall performance into a topic and system effect. Tague-Sutcliffe and Blustein [168] also hypothesized that the topic*system interaction should be an important factor but were unable to estimate the effect size as they did not have a sufficient number of replicates available in their experiments. Banks et al. [12] provided an overview of methods available to analyze the performance of IR systems and reexamined the two-way ANOVA model of Tague-Sutcliffe and Blustein to compare and contrast each of the methods. Banks et al. also performed an indirect estimation of the size of the topic*system interaction effect, providing further evidence that it has a significant effect size.

Since then, much effort has been devoted to increasing the statistical power of the linear models in IR by adding factors of different nature. Bodoff and Li [22] used multiple relevance judgments to induce the replicates needed to estimate the topic*system interaction effect. They relied on *Generalizability Theory* [23, 157], a generalization of ANOVA, to estimate the topic*system, topic*assessor, and assessor*system interaction to improve model accuracy when assessing differences between systems. Similarly, Robertson and Kanoulas [130] used simulated data to show that an ANOVA model including the topic*system interaction is better equipped to detect significant differences between systems and reaffirm that topic*system interactions consistently have large effect sizes.

Bailey et al. [7] presented an ANOVA model comprised of topics, systems, and query variations factors and found that query variations also have a large-sized effect that can be even bigger than the topic*system effect. However, Bailey et al. [7] treated query variations and topics as separate factors, which may be at odds with independence assumptions made by the standard ANOVA model. Similarly, Culpepper et al. [43] use topic reformulations and multiple collections in their ANOVA models.

Ferro and Silvello [64, 65] decomposed system factors into component effects and their respective interactions using a Grid of Points (GoP) [58], which is the set of all system configurations possible when permuting every targeted system components. Their work found that stop lists have a medium-size effect, stemmers have a small-size effect and IR models have a small to medium-size effect. Only the stop list*IR model interaction was significant, with a medium effect size. Ferro and Silvello used Terrier[2] to generate the GoP, and more recently Ferro [57] used Lucene[3] to generate a GoP using (almost) identical components as Ferro and Silvello to perform similar experiments. Ferro found that the stemmer has a large effect size while the stop list has a small effect size, and only the stemmer*IR model interaction was significant, with a small effect size. This suggests that the subtle implementation differences between the two systems may influence findings in empirical studies that measure system component effects. Frameworks such as those proposed by Ferro and Silvello [64, 64] and Carterette [31] are meant to unify the evaluation techniques. In particular, such works study how to model an arbitrary number of factors, obtaining more powerful statistical analyses. These works recognize the connection between our current evaluation approach and the GLMs framework without actually using Generalized Models.

Sanderson et al. [147] and Jones et al. [83] studied how sub-corpora or *shards* of a given collection impact IR effectiveness and how collection size and the choice of documents influenced the way that evaluation exercises using a single test collection might influence comparisons between retrieval systems. These studies emphasized the impact of sub-corpora/shards on system performance. However, they did not rely on a comprehensive ANOVA model that integrates all of the possible factors. Ferro and Sanderson [61] developed such an ANOVA model and found that the shard factor has a medium-size effect while the shard*system interaction was not significant.

In similar work, Voorhees et al. [184] used sharding to produce the replicates necessary for estimating topic*system interaction effect sizes. Ferro et al. [59] developed the first exhaustive model of topics, systems, shards, as well as all their interactions, and ran extensive experiments using several sharding schemes, including a selection of randomized and deterministic methods. They found that the topic factor has a large effect, the system factor is a small to medium-size effect, and the shard factor is a medium-to-large effect (roughly half of the topic factor). Concerning interactions, the topic*system is a large-size effect (roughly one-third of the topic factor), the system*shard is a small-size effect, and the topic*shard interaction is a large-size effect – and could be as large as the topic effect in specific scenarios. Ferro and Sanderson [62] went on to provide a formal demonstration of why topic*shard interactions are crucial when determining which systems are significantly

[2] http://terrier.org/

[3] https://lucene.apache.org/

different from others. Faggioli and Ferro [53] compare and analyze how various ANOVA approaches behave under different conditions. Finally, Zampieri et al. [198] used ANOVA to model topics, systems and collections (and not sub-corpora or shards as in previous work) and found results consistent with those mentioned above.

2.4 Beyond Linear Modeling

Linear modelling, although fundamental in our typical experimental setup, has the limitation of not adhering perfectly to IR experimental data. This impairs both the descriptive and statistical power of our evaluation frameworks. Therefore, several approaches have been proposed to tackle this limitation in order to design models that better fit IR experimental data. We report in the following section the primary efforts in this direction, which can be organized in two macro areas: non-linear transformations of the IR measures and new modelling strategies.

2.4.1 Response transformation

One of the most relevant strategies to bring the IR data closer to the assumptions relies on transforming the data. The idea consists in applying a function to the observations to change their distribution. Transformations help obtain more normal and homoskedastic distributions of performance.

One of the first efforts to obtain more normal data is by Tague-Sutcliffe and Blustein [168] who proposed to use the arcsin of the square root or the rank transformations of the performance scores when the linear modelling assumptions are not satisfied. One of the most frequently used transformations is the logit transformation. The underlying logit function, by mapping scores in the (0, 1) interval to $\mathbb{R}$, brings the distribution of the IR performance scores closer to a normal distribution. Such transformation was originally studied by Cormack and Lynam [37] and later employed in practical experiments by Robertson and Kanoulas [130] and Berto et al. [21]. Robertson [129] explores a further smoothed version of the logit transformation of the AP, which exhibits higher normality, dubbed “yaAP”. The logit transformation has the limitation of not being defined for values equal to 0 and 1, requiring to either ignore them or resort or a smoothing-based solution, as proposed by [37, 129]. Robertson [128] compares the advantages - and disadvantages - of several transformation strategies, including log and logit ones.

While the efforts mentioned above focus mainly on the non-normality of the measurements, Sakai [143] and later Urbano et al. [176] explore a standardization-based approach aimed at increasing the homoscedasticity of the data. In [143, 176], authors propose to transform the performance measures into z-scores, as proposed by Webber et al. [191], and apply a linear transformation to such scores to reduce the inter-topic heteroscedasticity.

Both GLMs and response transformation exploit non-linear functions to improve the fitness of linear models to the data. Nevertheless, they represent two completely different modelling strategies regarding assumptions, computation and interpretation of the predictions.

2.4.2 Modeling approach

A final approach to deal with the departure of IR data from the assumptions underlying linear models consists in changing the modelling tool completely, going beyond the linear approach. Urbano [175] describes one of the first approaches to apply GLMs in the IR domain. In [175], logistic regression is employed to model and predict the performance over different topic splits. Another effort to define a different modelling strategy relies on the usage of the Geometric Mean of the AP (GMAP). This approach was employed initially for the TREC 13 Robust 04 evaluation campaign [180]. The objective of the Robust 04 track was to evaluate the systems on topics that were particularly challenging in previous TREC editions. GMAP has the advantage of weighing more the performance of the systems on low performing topics, allowing to better compare systems on hard queries. The approach was explored further by Robertson [129] and Berto et al. [21]. For small AP values, the mean of the scores using log and logit transformations and the GMAP almost coincide.

Finally, another approach to developing a comprehensive framework for the IR evaluation, an alternative to the pure linear modelling, was proposed by Carterette [30] and further developed by Carterette [32]. [30, 32] propose a solution that on the idea of using a Bayesian framework for hypothesis testing to compare IR systems.

2.5 Modeling Different Aspects

Besides modelling the performance of a system, several endeavours have been devoted to modelling different aspects of the information retrieval experimental pipelines, such as the topic difficulty or the query representation. We detail some of such efforts in the remainder of this section.

2.5.1 Topic difficulty

Query performance, query hardness, query quality and query ambiguity explore different aspects of topic difficulty. The difficulty of a topic can be based on system performance

[161, 125, 34, 196, 6, 40] or human perception [74]. Topic difficulty affects systems and users. It can also play an important role in user agreement during relevance assessment [154, 45] and QPP [28]. It is important to emphasize that most prior work uses the terminology query difficulty and topic difficulty interchangeably, which is not problematic when a single query represents a topic. This is the most common scenario in IR, but not true in this work and other recent work on query variants. The topic difficulty is generally defined as the average effectiveness of a set of systems for a topic, more specifically the "average" average precision (AAP) originally [113]. A similar approach was also used by Carterette et al. [34] to classify topics into "easy", "medium" and "hard" for the 2009 million query track.

The topic difficulty has also proven to be a relevant factor in IR evaluation. For example, in Mizzaro [112] the query difficulty is used to develop a "Normalized" version of the well-known AP measure that penalizes and rewards systems whether they perform well or poorly on easy or challenging topics. Note that as in other studies in this field, Mizzaro [112] use the concept of topic difficulty and query difficulty interchangeably. Additionally, they do not provide any generalized notion of "topic difficulty"; it is defined based on the performance achieved by most of the systems using a specific formulation of the topic and on a specific corpus. Roitero et al. [135] provide a deeper analysis of the relationship between query versus topic difficulty and IR evaluation. The ability to predict the difficulty of a topic allows adapting the system to the input query, as done by Amati et al. [4], and later by Pehcevski et al. [123]. More in detail, Pehcevski et al. [123] develop a topic difficulty classifier that uses textual features of the topic (such as the length of the title formulation or the narrative representing the topic) to predict the topic difficulty. This classifier allows the selection of the parameters used by the subsequent entity-ranking model. As in other work, the topic difficulty is defined based on the AP achieved by a set of systems on a specific corpus. There are additional studies on estimating query difficulty based on linguistic features of the collection [76, 94].

2.5.2 Simulating IR Data

One challenge impairing the evaluation of evaluation frameworks is the absence of ground truth. Several efforts have been made to simulate IR systems to obtain data for which we know underlying distributions and thus can act as ground truth. Wilbur [192] exploits simulations of IR data to compare non-parametric and parametric tests, finding the former to be superior. Robertson and Kanoulas [130] develop a bootstrap based simulation approach to model the intra-topic variance, showing its effect when modelling IR performance. To define a ground truth and compare different test strategies, Urbano et al. [176] adopt a simulation process, previously defined by Urbano and Nagler [178], capable of jointly modelling both the system's internal variance and its covariance with another system via copulas. More recently, Parapar et al. [122] model new runs as a stochastic process capable of simulating significantly different runs. Nevertheless, the proposed simulation process does not include the topic-system interaction. We argue that every simulation procedure, regardless of its accuracy, relies on underlying assumptions that might interact with the assumptions of GLMs considered in this work. We do not use simulated data in our study to avoid possible biases in the evaluation procedure due to the simulation process, leaving it for future work.

2.5.3 Query Representation

For decades, it has been known that short keyword queries often result in vocabulary mismatches, as the terms used in the query must exactly match the terms found in relevant documents in the collection being searched.

The vast majority of statistical retrieval models have an implicit term independence assumption, which means that small query reformulations can significantly impact retrieval performance. Stemming and lemmatization can help alleviate this problem for commonly used terms but not eliminate it. The classic approach to address the vocabulary mismatch challenge is *query expansion* [133, 24, 126, 96], which is the process of finding terms in the collection that the user did not initially select, but that are presumably relevant to the information need; actual relevance or pseudo-relevance feedback is the most common way to operationalize query expansion in practice [194]. This is analogous to typical user behaviour, where a user reformulates a query for a search engine by adding additional terms to the original query when the retrieved results are not satisfactory.

Several early studies investigate how users independently express an information need as a query. These "query variations" were explored in the context of TREC retrieval experiments and were shown to be highly effective when combined through fusion [13, 14, 158]. This line of work has seen a revival of interest in recent years as more powerful hardware and retrieval models provide new opportunities to leverage multiple sources of information simultaneously [9, 15, 17, 18, 16]. For example, fusing the result lists retrieved in response to query variations was shown to be of much merit [9, 15]. Using multiple query variations can also improve query expansion, as recently demonstrated by Lu et al. [104]. Interestingly, a recent study shows that, on average, query variations automatically selected from a query log of a commercial search engine can be as effective as human-created variations [101].

2.6 Towards Predictive Models

The cost in terms of money and time of online and offline IR evaluation has fostered the research toward the development of predictive techniques to estimate the performance of a system on a given query. In particular, in this regard, researchers followed two main directions: Query Performance Prediction (QPP) and Model selection. The former allows ranking queries based on the expected performance: this, in turn, should permit rewriting the query to make it perform better or suggest better formulations of the same information need. The objective of the latter, on the other hand, is to predict which model will accomplish better results on a specific query to adapt it at query issuing time. The remainder of this section will detail some of the most relevant efforts in these two areas.

2.6.1 Query Performance Prediction

Retrieval performance can vary widely across different systems, even for a single query [28, 43]. This has resulted in a large body of work on Query Performance Prediction (QPP). The QPP task is defined as "estimating retrieval effectiveness without relevance judgments" [28]. In practice, all previous work on QPP focused on the task of estimating *topic* difficulty concerning a given fixed retrieval method, where each topic was represented using a *single* query [127, 200].

Pre-retrieval predictors analyze query and corpus statistics prior to retrieval [73, 76, 116, 155, 204] and *post-retrieval predictors* that also analyze the retrieval results [6, 138, 160, 197, 205, 29, 44, 49, 4].

Pre-retrieval techniques exploit the distribution of the query terms within the collection, providing coarse-grained information on the expected performance of a given query. On the other hand, post-retrieval techniques leverage the information on the retrieval scores assigned by the retrieval model. Such techniques tend to perform better compared to pre-retrieval QPP [55] but are dependent on the considered models.

Predictors are typically evaluated by measuring the correlation coefficient between the AP values attained with relevance judgments and the values assigned by the predictor. Such evaluation methodologies are based on a *point estimate* and have been shown unreliable when comparing multiple systems, corpora and predictors [72, 153]. Hauff et al. [72] demonstrate that higher correlation does not necessarily attest to better prediction, and used Root Mean Square Error (RMSE) in their evaluation. Hauff et al. applied methods from Meng et al. [109] to compare 2 or more correlation coefficients and argued that to test the significance of differences in correlation between the predictors, Fisher's z transformation should be used and the Confidence Interval (CI) should be reported. When computing the CI for Pearson's linear correlation in the evaluation using multiple previously reported pre-retrieval predictors, they found that many of the predictors had overlapping CI and concluded that they were not significantly different from the best performing predictor. Hauff et al. focused on the prediction of normalized scores that can be compared to AP using linear correlation as measured with a parametric statistic. In this work, we focus on ranking the queries based on the retrieval effectiveness, which is analogous to a rank-based correlation given by Kendall's τ as our reference for the existing evaluation framework, but many other alternatives are possible. We chose to use a rank-based correlation as it is a non-parametric statistical method and makes no assumptions about the underlying distributions of the data.

Also of interest, recent work using query variations for QPP [170, 200, 199, 47] has demonstrated that the relative prediction quality of predictors can vary compared to the effectiveness of the queries used to represent the topics, and we explore such observation further using advanced statistical instrumentation. However, both Zendel et al. [200] and Culpepper et al. [43] show that the relative prediction quality of existing predictors can significantly change when varying the queries used to represent the information needs. Indeed, topic difficulty and query difficulty are, in essence, two different concepts that should be carefully distinguished since a topic (information need) can be represented using various queries, and retrieval effectiveness is sensitive to the query terms selected by the user.

QPP models

Table 2.3 reports the QPP models used throughout this work, we describe each of them more in detail in the following paragraphs.

Pre-retrieval models The traditional pre-retrieval models are roughly based on computing for each query term how much it can contribute to the overall performance.

SCQ [204]: in line with works of Salton and McGill [146], Spärck Jones [166], if a term is frequent in the corpus and present in a few documents, it is likely that such documents will be easily retrieved, while also hinting at their relevance. Given a query term, the SCQ is based on computing the collection frequency (CF) of a term – the total number of occurrences of the word on the collection – and multiplying it by the IDF. The SQC score w_t of the term t is defined as:

$$w_t = (1 + ln(f_{c,t})) \times ln\left(1 + \frac{N}{f_t}\right)$$

where $f_{c,t}$ is the collection frequency of the term t, N is the total number of documents and f_t the number of documents containing the term t (document frequency).

VAR [204]: The “VAR” predictor relies on measuring how variable is the frequency of a term in the documents. The rationale is that if the number of occurrences of a term is particularly variable, then it is easy for a system to rank higher documents containing a lot of occurrences of that term. Vice versa, if the token distributes equally over all documents, it is likely to be uninformative and unhelpful during the rank phase. Called D_t the set of documents containing t and $w_{d,t} = (1 + ln(f_{c,t})) \times ln\left(1 + \frac{N}{f_t}\right)$ the TF*IDF weight of the term t with respect to document d the var score w_t is defined as:

$$w_t = \sqrt{\frac{1}{f_t} \sum_{d \in D_t} (w_{d,t} - \bar{w}_t)^2}$$

where:

$$\bar{w}_t = \frac{\sum_{d \in D_T} w_{d,t}}{|D_t|}$$

IDF [42, 155]: Following the works from [166], the IDF can be used as an indicator of “relevance” if a term is contained in a small subset of documents (high IDF), it is likely such term is a strong indication of topical relevance of the document.

$$w_t = ln\left(1 + \frac{N}{f_t}\right)$$

Each of the scores mentioned above is computed singularly for each term of the query. After that, the scores for all the query terms can be arbitrarily combined into a unique prediction score. For example, the default SCQ predictor [204] is based on summing the SCQ score for each word of the query. Similarly, it is possible to compute the meanSCQ score [204] normalizing the SCQ predictor by the length of the query. For what concerns the VAR predictor, according to Zhao et al. [204] the best performing aggregations are again the sum, the mean and the maximum VAR over the query terms. Finally, Cronen-Townsend et al. [42] argued that the best combination for the IDF predictor is the mean, while Scholer et al. [155] showed that the best aggregation function was the maximum over the query terms.

Post-retrieval models Post retrieval methods assume to also have at hand the results of the retrieval using the query and are typically more elaborate.

Clarity [41]: the clarity predictor requires computing a relevance model using the first k documents retrieved (typically around 100-150) and computing the KL divergence between this model and the one induced by the entire corpus. The underlying idea is that, if the documents retrieved present a language model which differs particularly from the language model of the entire collection, then it is more likely that the system effectively ranked relevant documents on top.

Normalized Query Commitment (NQC) [162]: The NQC score is based on the score distribution of the retrieved documents. A disperse score distribution indicates that the ranker separated sharply top documents from the rest. This is evidence in favour of the fact that the query performed particularly good. The NQC score is formulated as follows:

$$NQC(q) = \frac{1}{|D|}\sqrt{\frac{1}{k}\sum_{d\in D^{[k]}}(s_d - \hat{\mu})^2}$$

Where k is a free parameter (typically around 50), s_d is the score given to document d by the ranker in response to the query q, $\hat{\mu}$ is the mean score given to the documents.

Even though NQC – as for WIG and SVM – formulation involves the scores of the documents, the approach requires to compute such scores using the Language Model approach, even if used to compute the prediction for other IR models. Changing this, typically lead to decreased performance.

Weighted Information Gain (WIG) [206]: Similarly to NQC, also WIG relies on the idea that if the top documents have widely different scores from a reference score, then it is more likely that the IR system retrieved correctly. Differently from NQC where the average score is used as a proxy, in WIG the reference score is the score of the entire corpus.

$$WIG(q) = \frac{1}{\sqrt{|q|}\cdot k}\sum_{d\in D^{[k]}} s_d - s_D$$

where s_D is the score of the entire corpus.

Score Magnitude and Variance (SMV) [169]: The SMV predictor tries to extend both NQC and WIG, by taking into account both the magnitude of the scores (WIG), but also their variance (NQC).

$$SMV(q) = \frac{\sum_{d\in D^{[k]}} s_d \cdot ln(\frac{s_d}{\hat{\mu}})}{k \cdot s_D}$$

Utility Estimation Framework (UEF) [159]: The UEF approach differs from the previous ones since it is a more general framework that can be instantiated with any of the predictors mentioned before, pre-retrieval included. The UEF approach assumes retrieving a list of documents using the relevance model constructed from the ranked list retrieved in the first round. After that, UEF computes the correlation between the original ranked list and the one obtained using the relevance model. The rationale is that if the two ranked lists are close, then we can assume that all and only the relevant documents have been retrieved. Vice versa, if the two ranked lists are dissimilar, it is more likely that the retriever did not correctly rank the relevant documents. The correlation is then used to re-weight the score associated with the query by one of the QPP models mentioned above.

Table 2.3 A summary of QPP models used in this work.

QPP model	**Description**
	Pre-retrieval
SCQ by [204]	Measures similarity based on $CF \cdot IDF$ to the corpus, summed over the query terms.
AvgSCQ by [204]	SCQ normalized by the query length.
MaxSCQ by [204]	The query term with maximal SCQ score.
SumVAR by [204]	Measures the variability of the query terms in the corpus.
AvgVAR by [204]	Variability normalized with the query length.
MaxVAR by [204]	The query term with maximal variability.
AvgIDF by [42]	The mean IDF value of the query terms.
MaxIDF by [155]	The query term with maximal IDF value.
	Post-retrieval
Clarity by [41]	Measures the divergence between the Language Model (LM) constructed over top documents in the result list to the LM of the entire corpus.
NQC by [162]	Measures the standard deviation of the top documents scores in the retrieval list.
WIG by [206]	Measures the difference between the mean retrieval score of the top retrieved documents and the score of the entire corpus.
SMV by [169]	Scores the queries based on a combination of the scores standard deviation and magnitude.
UEF by [159]	Prediction framework that is based on the similarity of the initial result list with the list re-ranked using a Relevance Model (RM), scaled by an estimator of the RM quality. In this work we scale the RM with the existing post-retrieval predictors: UEF(Clarity), UEF(NQC), UEF(WIG) and UEF(SMV).

2.6.2 Model Selection

One of the first approaches to model selection in IR was developed by He and Ounis [75], who proposed a query-based pre-retrieval approach. He and Ounis [75] cluster queries according to pre-retrieval features and link the best performing model to each cluster. Then, given a new query, they assign it to the closest cluster and use the model associated with that cluster to perform retrieval. Balasubramanian and Allan [11] proposed a learning approach for query-dependent model selection. The selection framework relies on rank-time features – available to retrieval models during ranking – to select between two models. Model selection approaches based on rank-time features have been further explored by Balasubramanian in [10]. Beyond model selection, Levi et al. [98] addressed the problem of selective cluster retrieval [68, 102, 171], where the objective is to decide, on a per-query basis, whether to apply cluster-based retrieval or standard document retrieval. Levi et al. [98] proposed different sets of features based on cluster-based rankers, query performance predictors, and cluster properties. The different features sets are used to decide between cluster-based and standard document retrieval. The objective of Levi et al. [98] is to select the most effective approach between cluster-based and document-based retrieval given the query. Even though QPP approaches and model selection strategies have many commonalities, QPP techniques cannot be directly applied to model selection since, in most cases, they rank queries and scores cannot be compared across systems. Nevertheless, the signals provided by QPP models can be used as input features for the model selection task. For example, He and Ounis [76] explore the possibility to use the distribution of the IDF over query terms to determine the ability of lexical models to retrieve relevant documents. Similarly, Zhao et al. [204] propose a re-weighting schema based on IDF, called SCQ, while Mothe and Tanguy [116] considers linguistic aspects – such as synonymy and polysemy – linked to the query terms.

Part I

Performance Modeling

Evaluation in Information Retrieval (IR) plays the fundamental role of allowing researchers and practitioners to study and compare their systems and understand how to improve them [79, 149, 31]. The importance of the evaluation determines the relevance of statistical inference techniques used to compare the systems. Indeed, sound statistical inference methods allow us to obtain robust and generalizable insights and predict what happens when systems run in a real-world scenario.

Statistical analyses, such as bootstrap, randomization tests [141, 165], t-tests and ANOVA [140, 168, 12] have been successfully employed in IR evaluation. Besides allowing us to compare systems, statistical analyses can also be used to model the performance and its relation to experimental conditions. They can prove vital to determining what features and aspects of a scenario – systems considered, corpus, queries and their formulation – correlate with the performance.

In Part I of this manuscript, we explore how to use statistical analyses, especially linear models and ANOVA, as modelling tools to describe the performance of an IR system, and what intuition they can provide on the concept of topic difficulty, and how we can improve them. To this end, we will organize our analyses as follows:

- We deepen our understanding of ANOVA in the IR setting;
- Using ANOVA, we determine how different aspects of an IR experimental pipeline influence the system performance;
- We exploit GLMs to generalize ANOVA to increase the explanatory power of our statistical tools.

The remainder of this part of the manuscript is organized as follows. In Chapter 3 we present our investigation concerning different ANOVA approaches. Subsequently, in Chapter 4 we illustrate how ANOVA models can be used to model IR systems' performance and to gain deeper insights into the concept of topic difficulty. Finally, Chapter 5 describes an approach to further enhance the descriptive power of our statistical tools via the usage of GLMs.

Chapter 3

Statistical Tools to Model IR Data: Replicating Bootstrap ANOVA

In this chapter we explore different versions of ANOVA used to model IR data to understand how they behave under different experimental setup. More in detail, we focus on determine the degree of reproducibility of `bANOVA`, originally proposed by Voorhees et al. [184] and compare it with the traditional ANOVA approach, as proposed by Ferro and Sanderson [62].

3.1 Introduction

Using reproducible – and thus trustworthy – statistical tools is crucial to drawing robust inferences and conclusions. In recent years, many fields have devoted a lot of effort to reproducing and generalizing their systems and algorithms [60, 106, 56, 36]. Yet, the literature still lacks reproducibility studies on the statistical tools used to compare the performance of such systems and algorithms. Intending to deepen our understanding of ANOVA models, we investigate the reproducibility and compare two recently developed ANOVA models:

- Voorhees et al. [184] used sharding of the document corpus to obtain the replicates of the performance score for every (topic, system) pair needed to develop a model accounting not only for the main effects but also for the interaction between topics and systems; Voorhees et al. also used an ANOVA version based on residuals bootstrapping [51], which we call `bANOVA`.

- Ferro and Sanderson [62] used document sharding as well, but they developed a more comprehensive model, based on traditional ANOVA, which also accounts for the shards factor, the shard*system interaction, and the topic*shard interaction; we call this approach tANOVA.

Another fundamental aspect to consider when comparing several IR systems is the need to adjust for *multiple comparisons* [66, 144]. Indeed, when comparing just two systems, significance tests control the *Type-I error* at the significance level α. A Type-I error (also called *false positive*) corresponds to falsely rejecting a null hypothesis. Concerning the IR evaluation setting, it means to find a statistically significant difference between a pair of systems when they are not. The risk of committing a Type-I error is controlled by setting a low significance level α when carrying out the test of hypotheses. However, when c simultaneous tests are carried out, the probability of committing at least one Type-I error increases up to $1-(1-\alpha)^c$.

Several procedures have been developed for controlling Type-I errors when multiple comparisons are performed [78]. Voorhees et al. adopted a lenient False Discovery Rate (FDR) correction by Benjamini and Hochberg [20]; Ferro and Sanderson used a conservative Family-wise Error Rate (FWER) correction, using the Honestly Significant Difference (HSD) method by Tukey [173].

We identified three aspects that can impact the reproducibility of the ANOVA approaches above-mentioned: *i)* the strategy used to obtain replicates, *ii)* the kind of ANOVA used, and *iii)* the control procedure for the pair-wise comparisons problem. We articulate the study of the ANOVA tool into two research questions:

RQ 3.1 Determining the degree of replicability of the evaluation methodology proposed in Voorhees et al. [184];

RQ 3.2 Studying the behaviour of tANOVA and bANOVA under different experimental settings – with respect to the above-mentioned focal points – and the generalizability of their results.

3.2 Bootstrap ANOVA (bANOVA)

The bootstrap based version of ANOVA is the focus of our reproducibility anaylsis. It relies on bootstrap sampling of the residuals produced by a tradional ANOVA linear model. The use of bootstrap is motivated by the fact that, since it does not rely on the traditional F statistics, it allows for minimizing the assumptions imposed on the distribution of the data. To compute the bootstrap ANOVA, it is necessary to fit a traditional ANOVA linear model. Once the model is estimated, we can use it to compute the estimated performance $\hat{y}_{ijk}$, for the i-th topic, using the j-th system on the k-th shard. Note that estimated performance values can be organized in an estimated performance tensor $\hat{\mathbf{Y}}$, where $\hat{Y}_{ijk} = \hat{y}_{ijk}$. Afterwards, residuals are computed as $r_{ijk} = y_{ijk} - \hat{y}_{ijk}$, where y_{ijk} is the observed performance value. Called $\mathscr{R}$ the set of all residuals, B different perturbation tensors $\mathbf{R}^{(b)}$ are sampled, with $b \in \{0,...,B-1\}$. In particular, $R^{(b)}_{ijk} = r^{(b)}_{ijk}$ where $r^{(b)}_{ijk}$ is sampled uniformly with replacement among all possible original ANOVA residuals $\mathscr{R}$. These perturbation tensors are then added to $\hat{\mathbf{Y}}$, producing B perturbed observation tensors $\tilde{\mathbf{Y}}^{(b)}$. Each perturbed observation tensor is then used to fit an ANOVA model, providing B new bootstrap sampled estimations for the effect of each system. Using these estimations, it is possible to fit a Probability Density Function (PDF) of the effect of the system. Note that, Voorhees et al. do not specify the approach to fit the PDF, and thus we used the Kernel Density Estimation (KDE) technique [189], using a Maximum Likelihood Estimation (MLE) approach. The average MLE bandwidth is 0.0016 and ranges between 0.0005 and 0.0033, according to the system, the number of shards, and model considered. Such distribution is used to compute the p-value associated with the null hypothesis that the system with greater effect is not statistically significantly better then the other (one-tail hypothesis). Once a p-value for each pairwise comparison is available, Voorhees et al. propose to apply Benjamini-Hochberg correction procedure to correct for multiple comparisons. Finally, using the information on the number of significant differences found, Voorhees et al. propose a strategy to compute an interval of confidence around the system effect, by trimming the vector of the bootstrap sampled estimations of the system effects. In particular, the proportion of samples removed from each side is $\alpha\frac{k}{2N}$, where N is the total number of pairwise comparisons between systems and k is the number of pairs of systems for which one of the two system has statistically larger effect size, according to the Benjamini-Hochberg procedure.

As a final remark, using either bootstrap ANOVA or the traditional ANOVA, does not change the expressiveness of ANOVA itself: the ANOVA models that can be used to describe the data remain the same. The only difference between the two approaches relies on how such models are fitted – either using the bootstrap procedure or without it.

3.3 Experimental Setup

Akin Voorhees et al. [184], we used two collections: the TREC 3 Adhoc track [71] and TREC 8 Adhoc track [183]. TREC 3 contains 50 topics and 40 runs for a total of 820 pairwise run comparisons. TREC 8 consists of 50 topics and 129 runs for a total of 8,256 pairwise run comparisons.

We use Average Precision (AP) and Precision (P) with the cutoff at 10 documents (P@10) as performance measure. The document corpus has been split in $2,3,5,10$ even-sized random shards and we repeated the sampling 5 times.

3.3.1 ANOVA Models

One of the main advantages of using ANOVA, for both `tANOVA`and `bANOVA`, is linked to its capability of modelling an arbitrary number of different experimental conditions to connect them to the performance observed. Throughout this work, we exploit six main categories of ANOVA models, depending on the experimental approach followed to collect the performance measurements. We consider ANOVAs based on either a single formulation for each topic or that employ multiple formulations to describe each information need. Furthermore, we divide the considered ANOVA models between those that rely on a single shard or corpus and those that include multiple shards or corpora.

We describe here the ANOVA models used in the reminder of this chapter.

Single Shard ANOVA Models

The simplest experimental setting involves using a single shard – or corpus. In this case, the performance of a IR can be described using the following ANOVA model:

$$y_{ij} = \mu_{\cdot\cdot} + \tau_i + \alpha_j + \varepsilon_{ij} \tag{MD0}$$

- $\mu_{\cdot\cdot}$ is the grand mean;
- τ_i is the effect of the i-th topic;
- α_j is the effect of the j-th system;

As pointed out in Subsection 2.1.1 MD0 is the oldest and most straightforward effort to use ANOVA to model the performance of an IR system. Originally proposed by Tague-Sutcliffe and Blustein [168] and later further explored by Banks et al. [12], this model describes the performance as a linear combination of the effect of the topic at hand and the system used to answer it.

Sharded Corpus ANOVA Models

Empirically, it is well-known that different shards (or corpora) can have a huge impact in determining the performance of a system. To better quantify from a mathematical standpoint the impact of changing the underlying set of documents, we can exploit the following ANOVA models:

$$y_{ijk} = \mu_{\cdots} + \tau_i + \alpha_j + \varepsilon_{ijk} \tag{MD1}$$

$$y_{ijk} = \mu_{\cdots} + \tau_i + \alpha_j + (\tau\alpha)_{ij} + \varepsilon_{ijk} \tag{MD2}$$

$$y_{ijk} = \mu_{\cdots} + \tau_i + \alpha_j + \beta_k + (\tau\alpha)_{ij} + (\tau\beta)_{ik} + (\alpha\beta)_{jk} + \varepsilon_{ijk} \tag{MD3}$$

where:

- β_k is the effect of the k-th shard;
- $(\tau\alpha)_{ij}$, $(\tau\beta)_{ik}$, and $(\alpha\beta)_{jk}$ are respectively interactions between topics and systems, topics and shards, and systems and shards;
- ε is the error committed by the model in predicting y.

Notice that the same models can be employed to model the scenario in which, instead of having multiple shards, we have multiple corpora. In that case, β_k corresponds to the effect of the k-th corpus, and similarly, the interactions concern different corpora.

Our MD1 is the generalization of the model originally used by Tague-Sutcliffe and Blustein [168] (MD0) when multiple corpora are considered. It corresponds to the model in equation (2) of Voorhees et al. [184] and to (MD2) of Ferro and Sanderson [62]. Our MD2 corresponds to the model in equation (3) of Voorhees et al. [184] and to (MD3) of Ferro and Sanderson [62]. Finally, our MD3 corresponds to the model (MD6) of Ferro and Sanderson [62]. Voorhees et al. [184] did not experimented with the latter model; so, its usage represents an aspect of generalizability.

3.4 Replicability of bANOVA

We tried to replicate the widths of the confidence intervals of the system effect and the number of SSD pairs, i.e. systems for which one is significantly better than the other. Table 3.1 reports the results of our replicability analysis. Confidence intervals are much smaller, approximately halved, than those reported in the original paper. On the other hand, the number of SSD pairs is slightly higher for both AP and P@10; however, this could be still considered within the bounds of the variability due to the random sharding, observed also by Voorhees et al.. To further investigate the interval size, we hypothesized that, even if the original paper describes a single-tailed test, its implementation might have used a more-strict two-tailed one, which is often the default in many statistical software libraries. Table 3.2 shows the results when using such a two-tailed test. We can note that the confidence intervals are still very similar to

Table 3.1 Confidence interval widths on systems effects and number of SSD system pairs using one-tailed bANOVA on TREC 3. Between parentheses, values originally reported by Voorhees et al.; dashed values were not reported in the original paper.

		no interactions (MD1)				**interactions** (MD2)			
sample	**measure**	**mean**	**min**	**max**	**SSD**	**mean**	**min**	**max**	**SSD**
2 shards	AP	0.045	0.044	0.045	683.80	0.016	0.016	0.017	749.00
		(0.075)	(0.071)	(0.082)	(—)	(0.029)	(0.026)	(0.031)	(743)
	P@10	0.078	0.076	0.080	666.00	0.038	0.037	0.039	728.00
		(0.130)	(0.122)	(0.140)	(—)	(0.065)	(0.061)	(0.069)	(712)
3 shards	AP	0.038	0.037	0.039	699.40	0.018	0.018	0.019	746.20
		(0.064)	(0.060)	(0.069)	(—)	(0.032)	(0.030)	(0.034)	(741)
	P@10	0.062	0.061	0.063	682.20	0.037	0.036	0.037	727.00
		(0.106)	(0.099)	(0.112)	(—)	(0.065)	(0.061)	(0.071)	(712)
5 shards	AP	0.033	0.032	0.033	714.40	0.020	0.020	0.021	742.20
		(0.055)	(0.052)	(0.058)	(—)	(0.033)	(0.031)	(0.034)	(—)
	P@10	0.046	0.045	0.047	697.00	0.031	0.030	0.032	723.00
		(0.081)	(0.076)	(0.086)	(—)	(0.055)	(0.052)	(0.060)	(—)

the case of Table 3.1 and, thus, the difference between one-tailed and two-tailed test is not the cause of the observed discrepancy. On the other hand, the number of SSD pairs is getting even closer to those of Voorhees et al.; a little bit less close in the case of P@10 but, as also observed by Voorhees et al., it is a less stable measure.

To understand the issue with confidence interval sizes, we modified how they are computed. Instead of removing a percentage of the total number of samples, as described by Voorhees et al., we treated that number as an integer value, representing the actual number of samples to discard. Basically, this milder cut-off allows for removing just the most extreme values. Table 3.3 reports the result for such modification and we can now see that these modified confidence intervals are closer to those of Voorhees et al.. To double-check the confidence intervals, we also tried the vice-versa, i.e., we used the intervals reported in Voorhees et al. to determine the number of SSD pairs. Note that Voorhees et al. use the BH correction to determine the SSD pairs and not the confidence intervals; in their case, they estimate confidence intervals in such a way that they should be consistent with the number of SSD pairs obtained by the BH correction. Since we do not have the sizes of the original intervals, we use, for all the systems, in turn, the mean, minimum, and maximum interval widths reported by Voorhees et al.. Table 3.4 reports the results of such analysis. The number of SSD pairs is still lower compared to the expected one, in the range of 30 to 70 less, on average (cf. Tab. 3.2). This suggests that the original intervals are still a bit large to

Table 3.2 Confidence intervals width on systems effects and number of SSD system pairs using two-tailed bANOVA on TREC 3.

		no interactions (MD1)				**interactions** (MD2)			
sample	**measure**	**mean**	**min**	**max**	**SSD**	**mean**	**min**	**max**	**SSD**
2 shards	AP	0.045	0.044	0.046	661.40	0.016	0.016	0.017	743.20
	P@10	0.078	0.076	0.080	639.60	0.038	0.037	0.039	717.40
3 shards	AP	0.038	0.038	0.039	678.80	0.019	0.018	0.019	739.60
	P@10	0.062	0.061	0.064	662.40	0.037	0.036	0.038	717.80
5 shards	AP	0.033	0.032	0.034	696.00	0.020	0.020	0.021	734.80
	P@10	0.047	0.046	0.048	677.60	0.031	0.030	0.032	712.00

obtain the reported number of SSD pairs; this might be due to the intrinsic accuracy of the estimation procedure or to some differences in the implementation, as we hypothesized in Table 3.3. Overall, we can conclude that it is possible to fully replicate the bANOVA with BH correction and the resulting number of SSD system pairs which, to us, is the core contribution of Voorhees et al. and what is used in actual analyses. On the other hand, we were not able to replicate the derived estimation of the confidence intervals and remains an open issue.

3.5 Impact of the multiple comparison strategies and bootstrapping

The following experiments are based on TREC 3: this allows us having comparable results with respect to Voorhees et al. [184], which mostly focuses on the collection mentioned above. Notice that, akin to Voorhees et al. [184], we repeat the sampling until all the shards contain at least one relevant document for each topic.

To investigate the differences between ANOVA approaches, our first analysis compares the number of SSD system pairs found by them. We consider the following multiple comparison procedures: HSD for tANOVA, as originally proposed by Ferro and Sanderson, indicated with tANOVA(HSD); BH for bANOVA, as originally proposed by Voorhees et al., indicated with bANOVA(BH); and, BH for tANOVA, indicated with tANOVA(BH). tANOVA with Benjamini-Hochberg correction is here employed and analyzed for the first time, representing a generalizability aspect. It takes the p-values on the difference between levels of the factors produced by the traditional ANOVA, but corrects them using the BH correction. The rationale behind it is that it enjoys the statistical properties provided by the ANOVA while granting a higher discriminative power, due to the BH correction procedure. Finally, in this specific

Table 3.3 Mean, Min and Max modified confidence intervals widths of systems effects on TREC-3, using 3 shards. Highlighted values are the closest to the original ones by Voorhees et al. (* for AP and ‡ for P@10).

		no interactions (MD1)			**interactions**(MD2)		
sample	**measure**	**mean**	**min**	**max**	**mean**	**min**	**max**
original	AP	0.064	0.060	0.069	0.032	0.030	0.034
	P@10	0.106	0.099	0.112	0.065	0.061	0.071
1	AP	0.065^{*}	0.061	0.071	0.033	0.030^{*}	0.035
	P@10	$0.106^{\ddagger}$	0.100	0.113	0.063	0.058	$0.069^{\ddagger}$
2	AP	0.065^{*}	0.061	0.072	0.032^{*}	0.030^{*}	0.034^{*}
	P@10	0.105	$0.099^{\ddagger}$	$0.112^{\ddagger}$	0.063	$0.060^{\ddagger}$	0.068
3	AP	0.068	0.065	0.073	0.037	0.034	0.041
	P@10	0.107	0.101	0.113	$0.066^{\ddagger}$	0.062	0.074
4	AP	0.065^{*}	0.060^{*}	0.070	0.030	0.028	0.033
	P@10	0.105	0.098	$0.112^{\ddagger}$	0.061	0.057	0.064
5	AP	0.065^{*}	0.059	0.069^{*}	0.030	0.026	0.032
	P@10	0.105	$0.099^{\ddagger}$	0.114	0.063	0.059	0.068
avg	AP	0.066	0.061	0.071	0.032	0.030	0.035
	P@10	0.106	0.099	0.113	0.063	0.059	0.069

setting, such correction procedure allows us to investigate whether the differences between the `bANOVA` and `tANOVA` are due to the different ANOVA computation (bootstrap vs direct computation of F-statistics), or are due to the correction procedure applied (BH vs HSD) correction. `zero` has been used as interpolation strategy; in Section 3.6.3 we empirically show that the interpolation strategy has a negligible effect on the results. Finally, we experiment all the models from (MD1) to (MD3) with all the ANOVA approaches; note that (MD3) has not been studied before for `bANOVA` and this represents another generalizability aspect.

Table 3.5 reports the results averaged over the five samples of shards together with their confidence interval. Numbers on the diagonal of Table 3.5 describe how many pairs of systems are considered SSD by a given approach; numbers above the diagonal are the additional SSD pairs found by one method with respect to the other. Table 3.5 shows that, as the complexity of the model increases from (MD1) to (MD3), the pairs of systems deemed significantly different increase as well, confirming previous findings in the literature. `tANOVA`(HSD) controls `tANOVA`(BH) since all the SSD pairs for `tANOVA`(HSD) are significant also for `tANOVA`(BH); this was expected since FWER controls FDR [78]. It is possible see this by considering the differences between approaches (above diagonal): by summing the difference between `tANOVA`(HSD) and `tANOVA`(BH) to the `tANOVA`(HSD) you obtain back

Table 3.4 SSD system pairs as obtained by using the confidence intervals widths reported by Voorhees et al.. Compare them with the ones reported in Table 3.1.

		no interactions (MD1)			**interactions** (MD2)		
sample	**measure**	**mean**	**min**	**max**	**mean**	**min**	**max**
2 shards	AP	577.20	590.00	563.20	711.00	721.60	706.00
	P@10	544.60	558.20	528.80	670.40	678.80	661.40
3 shards	AP	608.80	622.80	592.00	702.80	708.60	695.00
	P@10	573.80	583.20	562.00	659.80	667.60	638.60
5 shards	AP	638.80	645.60	629.00	697.40	704.80	695.00
	P@10	597.00	608.20	586.40	656.80	663.60	644.00

Table 3.5 SSD pairs of systems for different ANOVA approaches, using AP.

Model	**Approach**	bANOVA(BH)	tANOVA(BH)	tANOVA(HSD)
MD1	bANOVA(BH)	6866.60 ± 36.965	329.20 ± 22.027	2275.80 ± 39.844
	tANOVA(BH)	-	6537.40 ± 57.107	1946.60 ± 23.190
	tANOVA(HSD)	-	-	4590.80 ± 75.850
MD2	bANOVA(BH)	7231.80 ± 51.085	375.20 ± 17.436	2133.40 ± 70.456
	tANOVA(BH)	-	6856.60 ± 65.859	1758.20 ± 54.580
	tANOVA(HSD)	-	-	5098.40 ± 113.429
MD3	bANOVA(BH)	7563.40 ± 15.273	262.00 ± 11.681	1655.80 ± 25.377
	tANOVA(BH)	-	7301.40 ± 11.734	1393.80 ± 32.585
	tANOVA(HSD)	-	-	5907.60 ± 37.359

the number of SSD pairs identified by tANOVA(BH). However, this pattern holds also for bANOVA(BH) and tANOVA(BH), i.e. all the SSD pairs of tANOVA(BH) are SSD pairs for bANOVA(BH) too. While the relation between BH and HSD was expected, this finding sheds some light on the difference between using a traditional or a bootstrapped version of ANOVA. In summary, most of the increase in the SSD pairs is due to the correction procedure rather than the use of bootstrap or not. Since bANOVA is more computationally demanding than tANOVA, due to its iterative nature, its use may be not worth if not when you really need to squeeze out all the possible SSD pairs

3.6 A comparison between bANOVA and tANOVA

3.6.1 Comparing Tests

We are now in the position of comparing bANOVAagainst tANOVA. To do so,we follow an empirical strategy based on comparing the number of statistically significantly different pairs of systems found by the two approaches. In particular, to assess the stability of the different tests, we consider the agreement measures defined by Ferro and Sanderson [63], following previous works on the stability of statistical tests [53, 114, 177].

Without loss of generality, we assume to have two tests and a pair of systems - A and B. The two tests might differ, for example, for the topic considered or the inferential approach followed. According to the decisions taken by each test, we have the following possibilities: Active ("A-") decisions – both tests consider the difference between A and B statistically significant; Passive ("P-") decisions – none of the tests has achieved enough evidence to determine if A is statistically better than B; Mixed ("M-") decisions – only one of the test is capable of deeming A is statistically better than B.

Besides that, the parametric tests that we take into account are based on sample summary statistics (the mean in our case) that describes the effect of our experimental conditions on the magnitude of interest. Nevertheless, different tests might agree or not in considering the system A to have a more significant effect than system B - for example, because various topics or collections have been used. It might be possible that the two tests agree (Agreement "-A") or disagree (Disagreement "-D") on considering A to have a larger effect than B. This classification determines six possible scenarios: Active Agreements (AA); Active Disagreements (AD); Passive Agreements (PA); PD; Mixed Agreement (MA); Mixed Disagreements (MD).

- Active Agreements (AA): system A is deemed to be significantly better than B by both evaluation frameworks;

- Active Disagreements (AD): One of the evaluation frameworks deems A to be statistically better than B, while the other considers B to be statistically better than A;

- Passive Agreements (PA): Both evaluation frameworks do not reject the null hypothesis that system A is equal to B in terms of performance, and both tests agree on considering A to have a larger (although not significant) effect than B;

- Passive Disagreements (PD): Both evaluation frameworks do not reject the null hypothesis that system A is equal to B in terms of performance, but one of the tests considers A to have a larger effect than B, while the other considers B to be better than A (in both cases, effects are not significant);

- Mixed Agreement (MA): one of the tests deems A to be statistically better than B, while the other deems A to be better than B, without statistical significance;
- Mixed Disagreements (MD): one of the tests deems A to be statistically better than B, while the other deems B to be better than A, without statistical significance;

Several AA decisions denote the overall stability of the test. Conversely, AD implies that the test provides opposite inference depending on the topic set at hand, and thus it is an index of instability. PA and PD indicate that the test avoids taking decisions: this might not be wrong if the information is not enough, but it also suggests a weak test. The mixed scenario MA entails that a test is more inclined to consider a difference as significant: this reveals more power, but also that the test might be more prone to false positives. Similarly to MA, MD hints at more powerful tests. Nevertheless, MD indicates possible lower stability than MA since, on the two topic sets, the sign of the difference between the systems is opposite.

Agreement indicators can also be further aggregated as follows:

- The Proportion of Active Agreements (PPA), given by $PAA = 2AA/(2AA + MA + MD)$, represents how many times the two tests agree on two systems being SSD over the total number of times at least one of the test considers the pair of systems at hand to be different SSD;
- The Proportion of Passive Agreements (PPA), given by $PPA = 2PA/(2PA + MA + MD)$, shows how often an approach agrees on two systems not being SSD over the total number of times at least one test considers the pair of systems at hand to be not SSD.

PPA and PPA indicate, respectively, the stability of the decisions about which systems are and are not SSD, independently from the shard samples. These two proportions indicate how much you would not change your mind when changing the random shard sample at hand.

3.6.2 Effect of the Random Shards on the Stability of the Approaches

To move further from the reproducibility of `tANOVA`and `bANOVA`, subsequent experiments are reported on TREC 8.

To assess the stability of different approaches against random resharding, we fix the number of shards (5 in the following analysis). We resampled the shards 5 times and we considered all the possible pairs of shard samples – i.e. 10 possible pairs of shards. Then, for each

Table 3.6 Average PAA and PPA.

Model	Approach	Average PAA	Average PPA
MD1	bANOVA(BH)	0.979 ± 0.001	0.903 ± 0.005
	tANOVA(BH)	0.980 ± 0.001	0.924 ± 0.004
	tANOVA(HSD)	0.979 ± 0.002	0.973 ± 0.003
MD2	bANOVA(BH)	0.980 ± 0.001	0.866 ± 0.007
	tANOVA(BH)	0.979 ± 0.001	0.896 ± 0.006
	tANOVA(HSD)	0.977 ± 0.002	0.963 ± 0.004
MD3	bANOVA(BH)	0.982 ± 0.001	0.802 ± 0.012
	tANOVA(BH)	0.980 ± 0.001	0.850 ± 0.006
	tANOVA(HSD)	0.981 ± 0.001	0.953 ± 0.003

pair of shards, we compute PPA and PPA for each type of test – bANOVA(BH), tANOVA(BH) and tANOVA(HSD), following the definitions presented in Section 3.6.1. In this specific case, we are not comparing two different tests, but rather the same test computed on two different document sets.

We did not find any occurrence of AD in any of our experiments: AD would indicate a dependency of an approach on a specific random shard, raising some concerns – their absence indicates that all approaches have a sufficient level of stability.

Table 3.6 shows the PPA and PPA averaged over every possible pair of shards together with their confidence intervals. All the approaches have a very high PPA, suggesting that the conclusion about which systems are to be considered SSD is quite stable. The PPA is also very close for all the approaches, slightly increasing as we adopt the more sophisticated (MD3) model but without notable differences between bootstrap and traditional ANOVA or between HSD and BH correction. On the other hand, tANOVA approaches lead to higher PPA than bANOVA ones. The HSD correction produces notably higher PPA than the BH one. We hypothesize that the additional SSD pairs brought in by bootstrap and BH are "corner cases" and the decision about them depends more on the actual shards at hand. We can also observe as the PPA tends to decrease as the models get more sophisticated from (MD1) to (MD3); also, in this case, a more complex model can identify more SSD pairs, but some of them are "corner" cases subject to change from a random shard to another. Overall, the findings concerning PPA and PPA suggest that tANOVA with HSD correction is the most stable approach against different random shards. It should therefore be used when the goal is not the absolute number of SSD pairs, but the accuracy of the decisions.

Table 3.7 Average number of PD for ANOVA model MD2.

(MD2)		5 Shards			
Approach	Interp.	`zero`	`lq`	`mean`	`one`
tANOVA(HSD)	`zero`	230.60± 21.55	23.00± 15.21	100.20± 74.45	89.80± 82.47
	`lq`	—	239.20± 22.56	77.20± 62.86	85.60± 96.98
	`mean`	—	—	253.20± 32.18	124.40± 92.81
	`one`	—	—	—	265.80± 53.21
bANOVA(BH)	`zero`	282.60± 13.70	5.80 ± 3.45	41.60 ± 24.44	33.20 ± 28.83
	`lq`	—	280.80± 12.99	35.80 ± 21.12	32.60 ± 30.75
	`mean`	—	—	285.00 ± 13.24	49.20 ± 40.73
	`one`	—	—	—	288.40 ± 18.59

3.6.3 Stability of ANOVA Models with respect to Different Interpolation Values

As a final analysis, we remove the constraint that each shard contains a relevant document for each topic. This causes the computation of AP to be undefined for those shards without relevant documents. In such cases we interpolate the missing value using 4 possible strategies: `zero`; `lq`, the value of the lower quartile of the measure scores; `mean`, the average value of the measure scores; and, `one`.

We study the impact of the interpolation strategy, i.e. how to substitute missing values for topics without any relevant document on a given shard, for the different approaches. Here, for space reasons, we report only the results for tANOVA(HSD) and bANOVA(BH), being the tANOVA(BH) midway between these two.

Ferro and Sanderson [62] mathematically proved that model (MD3) is independent of the adopted interpolation values while Voorhees et al. [184] did not experiment with interpolation values and did not consider this model at all.

Tables 3.7 and 3.8 report the average PD counts together with their confidence interval (remember that AD turned out to be zero in our experiments), respectively for models MD2 and MD3. Values on the diagonal are the average PD observed using the same interpolation strategy, but over the pairs of shards samples. The upper triangle of the Table contains the average PD when using two different interpolation values. The PD counts on the diagonal are consistent with the findings of Table 3.6 in terms of PPA, confirming that bANOVA(BH) is more sensitive to the random sampling of shards than tANOVA(HSD).

Table 3.7 shows what happens if, using model (MD2) by Voorhees et al., instead of re-sampling shards we use an interpolation value. We can note that the PD count on the diagonal, compared to the one of Table 3.8, slightly increases for both bANOVA(BH) and

Table 3.8 Average number of PD for ANOVA model MD3.

(MD3) Approach	Interp.	**5 Shards** zero	lq	mean	one
tANOVA(HSD)	zero	222.60± 15.392	0.00± 0.000	0.00± 0.000	0.00± 0.000
	lq	—	222.60± 15.392	0.00± 0.000	0.00± 0.000
	mean	—	—	222.60± 15.392	0.00± 0.000
	one	—	—	—	222.60± 15.392
bANOVA(BH)	zero	279.20± 16.60	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
	lq	-	279.20± 16.60	0.00 ± 0.00	0.00 ± 0.00
	mean	-	-	279.20± 16.60	0.00 ± 0.00
	one	-	-	-	279.20± 16.60

tANOVA(HSD). On the other hand, the values are in the same confidence interval, and thus are not significantly different.We can also note that, as the interpolation value increases, the PD count on the diagonal tends to increase too. When it comes to the upper triangles, we interestingly find that bANOVA(BH) is much less sensitive to the interpolation values than tANOVA(HSD), being the PD counts substantially lower. Thus, Voorhees et al. could have used an interpolation value instead of re-sampling, without drastically changing the conclusions. The bootstrapped version of ANOVA (bANOVA) appears to be less stable with respect to the resharding. This phenomenon is likely due to its greater discriminative power: since a small evidence for bANOVA is enough to assess when two systems are different, the random resharding might produce spurious evidence and thus large variation among different samples. In Table 3.8, as expected from [62], the upper triangle for tANOVA(HSD) is zero, since tANOVA(HSD) with (MD3) is independent from the interpolation values. The most interesting finding is that also bANOVA(BH) with (MD3) is independent of the interpolation values. Indeed, the bANOVA approach samples the residuals and Ferro and Sanderson proved that they are independent of the interpolation value for (MD3). Therefore, using (MD3) also the bootstrap approach by Voorhees et al. does not need to re-sample shards.

3.7 Final Remarks

In this chapter we described a recent ANOVA, dubbed bANOVA. We replicated and compared it with the traditional approach. Our findings allow a better understanding of the strengths and weaknesses of the statistical tool when used to compare IR systems. In particular, we observed that bANOVA, compared to the traditional ANOVA, tends to be less conservative, allowing us to find more statistically significantly different pairs of systems. Nevertheless, this increased quality in discriminating capabilities is balanced by a decrease in the approach's stability.

Chapter 4

Applying Statistical Tools to Interpret Performance Changes: Topic Difficulty

4.1 Introduction

After exhibiting the statistical properties of the ANOVA modelling approach, we are interested in showing its explanatory power in a real-life scenario. We apply it to carry out a deeper analysis of IR systems' performance by breaking it down into its constituents to determine how each experimental condition contributes to the overall performance of a system. Following this line of thought, we further investigate the concept of "topic difficulty".

The interplay between simple keyword queries and extensive document collections has challenged researchers in IR for more than half a century. Document retrieval ranking models are now both complex and highly effective. However, poorly performing queries, sometimes referred to as *tail queries*, continue to surprise and challenge industrial and academic researchers. Some queries are highly effective, while others perform poorly, and changing the ranking models to compensate for *difficult* queries can have adverse effects on the performance of queries that were performing well previously. This notion of *query difficulty* has received a great deal of attention over the years. For example, NIST ran the Robust Track in 2004 and 2005 to reexamine sets of queries that had performed poorly across all systems evaluated in the Ad hoc track [180, 181].

A series of recent papers have begun exploring the relationship between query diversity and information needs [7, 9]. In experimental settings where an information need is clearly defined, a comprehensive analysis of query formulation is possible. While the idea that information needs can and should be expressed differently is not new, this important caveat can be lost when treating every query independently [13]. The distinction between a topic (information need) and a query can have a profound impact on the effectiveness of retrieval, as well as how IR researchers typically categorize and compare system performance.

We reexamine the idea of query difficulty from the topic perspective, where a topic can have many query formulations, and the retrieval system and the underlying document collection can change. We explore this issue by addressing the following research questions:

RQ 4.1 How does the formulation of an information need impact system performance *within corpora*?

RQ 4.2 How does the formulation of an information need impact system performance *across corpora*?

RQ 4.3 How does topic difficulty vary *across corpora* based on the formulation of an information need?

RQ 4.1 allows us to investigate the effect size of topics and query formulations with respect to systems and their components to better understand what contributes to topic difficulty, the magnitude of the effect, and which system components are most affected. This research question builds on an established body of prior work which studies the interaction between topics and systems. Here, we extend these approaches to include query formulations.

RQ 4.2 extends RQ 4.2 by looking at what happens across corpora and allows us also explore corpora-specific topic/query formulations jointly with system components. Replicates are available when queries can be used on multiple collections, allowing the interactions between query formulations and system components to be computed using ANOVA for the first time. Without the addition of multiple corpora, this interaction cannot be observed experimentally. Finally, RQ 4.3 examines the topic difficulty across multiple corpora.

To address RQ 4.1 and RQ 4.2, we develop a set of ANOVA models which allow us to break down the overall system performance into the topic, query formulation, system, and corpora effects; we call this a *macro-level* model. We also break down the system effect by component and show how each of these interacts with topics, query variations, and corpora; we refer to this level of granularity as a *micro-level* model. A GoP, i.e. a set of systems induced using all the combinations of targeted components – stop lists, stemmers, IR models, and Query Expansion (QE) in our case, is used as the data input into our models.

The *across corpora* ANOVA models also allow us to partially answer RQ 4.3 as they enable us to quantify changes in topic difficulty using multiple corpora. We measure variance in arbitrarily ranked topics across corpora to deeply investigate this research question. The key idea is that the likelihood of observing arbitrary rank orderings of topics by effectiveness

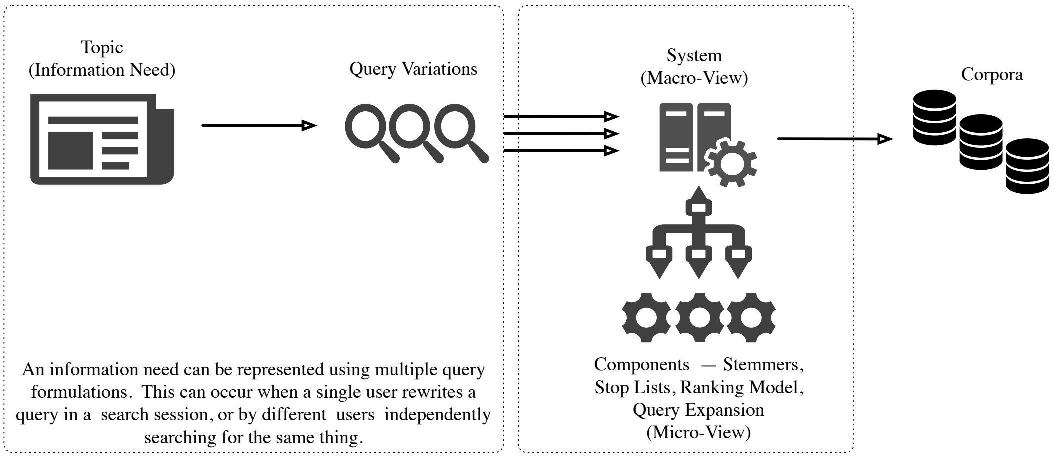

Topic
(Information Need)
Query Variations
System
(Macro-View)
Corpora
An information need can be represented using multiple query formulations. This can occur when a single user rewrites a query in a search session, or by different users independently searching for the same thing.
Components — Stemmers, Stop Lists, Ranking Model, Query Expansion
(Micro-View)

Fig. 4.1 Factor groupings used by the ANOVA models in this section. The two high-level groupings are macro-level and micro-level. The key distinction between the two groupings is that for the micro-level analysis, each system is decomposed into all possible combinations of component factors such as a stemmer, stop list, ranker, and query expansion model.

is analogous to topic difficulty being an *intrinsic* property. High volatility in topic ordering suggests that topic hardness is not absolute but is an artefact of system/corpora interaction.

We demonstrate and emphasize the fundamental difference between query difficulty and topic difficulty notions. More specifically, the idea of a single query being difficult is an artefact of collection design, but it appears that topic difficulty can reliably be circumvented through careful query reformulation. This is a promising step in a fundamentally important problem in IR — that of robust system effectiveness.

4.2 ANOVA Analysis

As many different factors are under consideration, we consider two different ANOVA models which are the *single-corpus* model and the *across-corpora* model, where the former is applied on a single corpus of documents and used to address RQ 4.1, while the latter is applied to multiple corpora in order to address RQ 4.2 and RQ 4.3. For each of these, we distinguish between a *macro-level ANOVA* model, which groups related factors, i.e. topics and query variations, system component configurations, and corpora, and a *micro-level ANOVA* model, which is a full break-down of all system factors into their respective contributions – stop list, stemmer, IR model, and query expansion.

Figure 4.1 shows the relationship between the macro-level and micro-level models.

4.3 Topic Difficulty via the Lenses of Topic Ranking

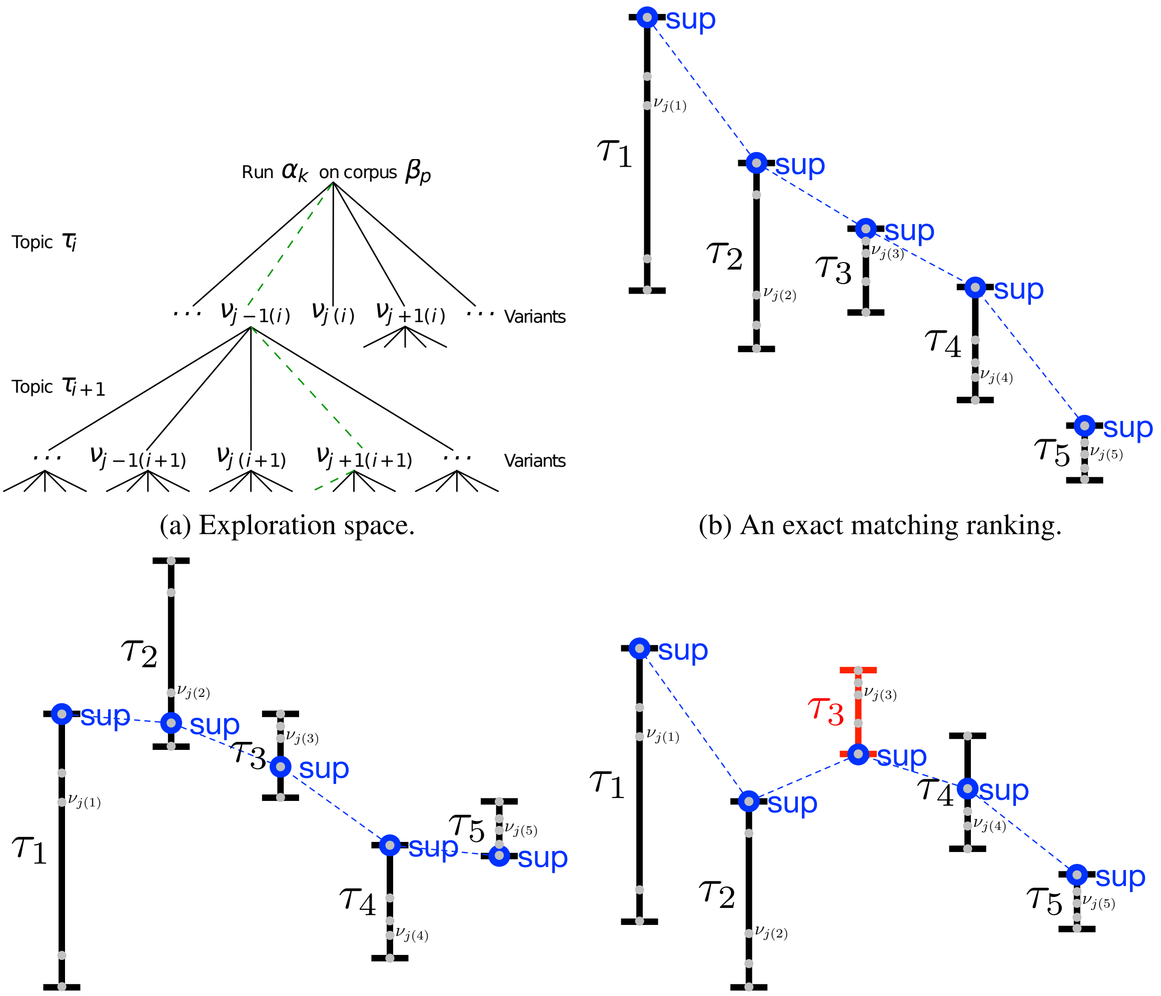
In order to gain a deeper insight into the notion of topic difficulty and further investigate RQ 4.3, we also consider the following question: to what extent can we find a system α_k whose performance on a corpus β_p results in an arbitrarily chosen ranking of topics τ_i? If we (often) succeed in finding a system that induces the desired ranking of topics, it means that the topic itself cannot be thought of as always easy or difficult, since it can appear at any position with respect to other topics within the ranking.

That is, the key insight here is that if "topic difficulty" in a system effectiveness sense is fixed, the estimator will converge towards a stable topic-wise ordering for all system configurations. However, if the estimator diverges from a fixed ordering as more samples are examined, topic difficulty is not idempotent. The more volatile the orderings, the more likely a (system, corpus) pair can be found, and topics can arbitrarily be hard or easy.

More formally, let C be the number of corpora, T the number of topics, V the number of query formulations per topic, and S be the number of systems. A random permutation $[\tau_1, \tau_2, \ldots, \tau_T]$ of topics is then selected and set as the *expected* rank ordering of the topics in the set. Then, all possible permutations within the test data are inspected to see if a match for the target rank ordering of topics exists. More specifically, for each (system, corpus) pair and all query formulations $v_{j(i)}$ available for each topic, the query formulations $[v_{j(1)}, v_{j'(2)}, \ldots, v_{j''(T)}]$ are selected when the performance of system α_k on corpus β_p induces the requested ranking of topics.

As shown in Figure 4.2a on the left, for each system α_k on a corpus β_p, there are V^T possible combinations of query reformulations and zero or more of them may induce the expected topic ordering. Therefore, in the worst case, a total of $C \cdot S \cdot V^T$ topic-wise rankings must be inspected in order to determine if exact or partial match solution exists. This process is then repeated for P random permutations, which is not computationally tractable in practice for even moderately sized test sets.

Therefore, we propose a greedy algorithm with quartic complexity $\mathcal{O}(C \cdot S \cdot T \cdot V)$ in the worst case. The pseudo-code for our greedy algorithm is shown in Figure 4.1. When at least one ranking of topics exists which exactly matches the expected ordering, the algorithm is guaranteed to find a solution. When no such ranking exists, the algorithm finds a sub-optimal ranking that is "close enough" to the one requested, but a better (although not exactly matching) ranking may still exist. So in this sense, our algorithm provides a lower bound for the case of topic rankings which do not exactly match the requested one, and is suitable for our purposes since exact matches provide empirical evidence for our hypothesis that not true topic ordering can actually exist, and therefore solutions further away from exact are conservative estimates.



Run α_k on corpus β_p
Topic τ_i
$\cdots$ $\nu_{j-1(i)}$ $\nu_{j(i)}$ $\nu_{j+1(i)}$ $\cdots$ Variants
Topic τ_{i+1}
$\cdots$ $\nu_{j-1(i+1)}$ $\nu_{j(i+1)}$ $\nu_{j+1(i+1)}$ $\cdots$ Variants
sup
τ_1 $\nu_{j(1)}$
τ_2 $\nu_{j(2)}$
τ_3 $\nu_{j(3)}$
τ_4 $\nu_{j(4)}$
τ_5 $\nu_{j(5)}$

(a) Exploration space.

(b) An exact matching ranking.

(c) Another exact match ranking.

(d) A partial match ranking: all the formulations for topic τ_3 have higher AP than for topic τ_2.

Fig. 4.2 Ranking topics by their performance.

Figures 4.2b-4.2d demonstrate the algorithm in action. Assume that the random permutation calls for the following topic ranking: $\tau_1 \geq \tau_2 \geq \tau_3 \geq \tau_4 \geq \tau_5$. So, a (system, corpus) pair is fixed and, in Figure 4.2, each "bar" corresponds to a topic, where the bar represents the range of performance of the query formulations for that topic, and each gray dot on the bar is the true performance – AP in our case – of system α_k on corpus β_p for formulation $\nu_{j(i)}$.

Next, the algorithm attempts to find the requested ordering by iterating over topics from this targeted ordering. The basic idea is that the maximum of a topic must be greater than or equal to the minimum of the next topic. This holds in the easiest cases, as the one shown in Figure 4.2b, and can even lead to topic orders which never change in certain corner cases. For example, a poor query formulation for an "easy" topic might still perform worse than the

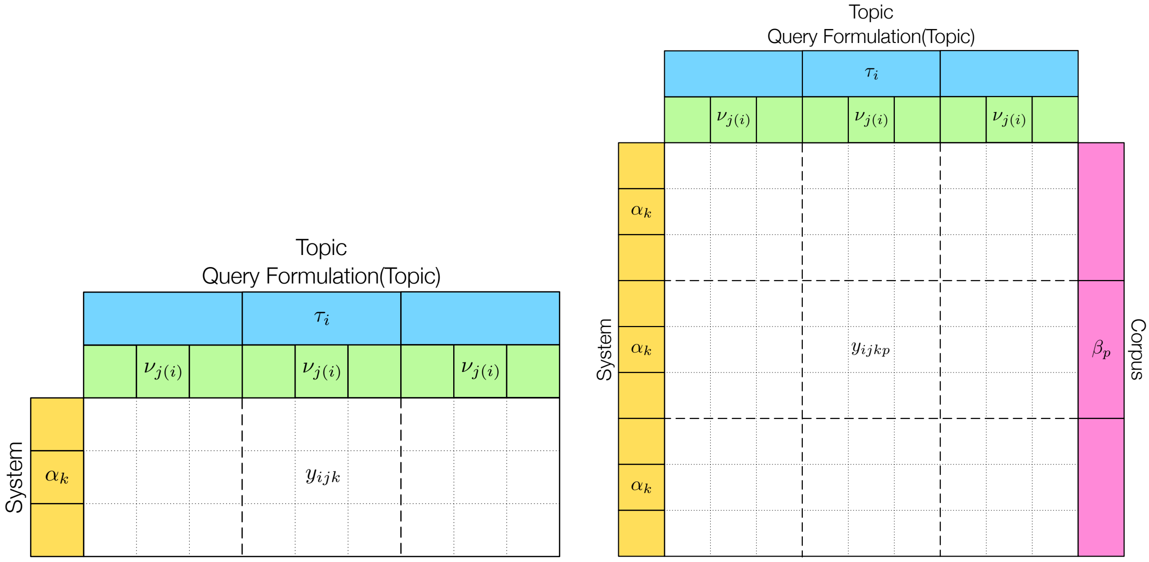
Algorithm 4.1: Pseudocode for the greedy search algorithm.

```
Data: $\mathscr{T}$: list of topics; $\mathscr{B}$: set of corpora; $\mathscr{A}$: set of systems; $\mathscr{P}$: sample of all
      possible permutations of topics; $\mathscr{V} = \{\mathscr{V}_{(\tau_i)} \forall \tau_i \in \mathscr{T}\}$ where $\mathscr{V}_{(\tau)}$ is a set of
      query reformulations for topic $\tau$; AP tensor containing AP scores for each
      triple (corpus, system, query);
$globalCorr \leftarrow 0$;
for $\pi \in \mathscr{P}$ do
    $bestCorr \leftarrow -1$;
    for $\beta \in \mathscr{B}, \alpha \in \mathscr{A}$ do
        $\tau \leftarrow \pi\ [1]$;
        $sup \leftarrow \mathrm{MAX}\ (\mathbf{AP}\ [\beta, \alpha, \mathscr{V}_{(\tau)}])$;
        /* The list sortedAP contains the topic AP score mapping of
           the query reformulations that induce the ordering with the
           highest correlation $\pi$.  */
        $sortedAP \leftarrow [sup]$;
        for $\tau \in \pi\ [2 : end]$ do
            if $\exists v_{i(\tau)}$ s.t. $\mathbf{AP}\ [\beta, \alpha, v_{i(\tau)}] \leq \mathrm{sup}$ then
                $sup \leftarrow \mathrm{MAX}\ (\mathbf{AP}\ [\beta, \alpha, \{v_{i(\tau)} \forall v_{i(\tau)} \in \mathscr{V}_{(\tau)}\ \text{s.t.}\ v_{i(\tau)} \leq sup\}])$
            else
                $sup \leftarrow \mathrm{MIN}\ (\mathbf{AP}\ [\beta, \alpha, \mathscr{V}_{(\tau)}])$;
            $sup \oplus sortedAP$;
        $bestCorr \leftarrow \mathrm{MAX}\ (bestCorr, \mathrm{KENDALL}\ (\pi, sortedAP))$;
    $globalCorr \oplus \frac{bestCorr}{|\mathscr{P}|}$
```

best known query formulation for a “difficult” topic, suggesting that it is not hard to show that a topic is either easy or difficult depending on the goal. However, as we will show in the experimental section, the patterns observed in the collections available tend to be much more complex.

Moreover, the simple max-min strategy described above really just ensures a relative ordering among topics, but not an overall ordering starting from τ_1. Indeed, it is also possible for $\tau_1 \geq \tau_i$, $\tau_i \geq \tau_{i+1}$ and $\tau_{i+1} \geq \tau_1$ to occur. Therefore, in each iteration, the maximum allowed value (*sup* in Figure 4.2) is updated by choosing *sup* as the maximum of the performance of the formulations of the next topic which are less than or equal to the current *sup*. This choice accommodates more complex cases, such as the one shown in Figure 4.2c, which still induces the desired ordering.

Finally, the requested ranking *may be impossible*, as shown by the red τ_3 in Figure 4.2d. In this case, we must choose as new *sup* which is the minimum performance of the formulations of the non-compliant topic, and the algorithm attempts the ranking selection process again.



(a) Single-corpus model (MD4$_{ma}$).

(b) Across-corpora model (MD5$_{ma}$).

Fig. 4.3 Macro-level ANOVA model design.

Since we can have both exact matches and partial matches, we compute Kendall's tau correlation [88] between the ranking of topics requested by the given permutation and the one we have found for a given (system, corpus) pair. Kendall's tau is 1 when we find an exact match and less than 1 otherwise. Finally, for each permutation, we record the maximum Kendall's tau across all of the (system, corpus) pairs to indicate the extent to which we have been able to find the request ranking of topics.

Note that we have adopted the use of Kendall's tau correlation coefficient, which weights the same a swap at any rank position, and not a more top-heavy correlation coefficient, like AP correlation [195], because our goal is to study the extent to which topics can be "arbitrarily" easy or difficult, and thus we are equally interested in swaps at any position in the ranking.

4.4 Experimental Setup

4.4.1 ANOVA Models

Multiple Topic Formulations - Single Corpus ANOVA Models

Similarly to what happens with multiple corpora, our practical experience highlights how changing the way in which we express our information needs strongly influences the quality of the results. In this sense, it is of uttermost importance to measure and statistically evaluate the role played by multiple formulations in causing changes in terms of systems' performance. Notice that, differently from the previous scenarios, for what concerns this set of models, we consider both their macro and micro variants.

The macro-level ANOVA model that embeds also the effect of using different formulations to express the information need is the following:

$$y_{ijp} = \mu + \tau_i + \nu_{p(i)} + \alpha_j + (\tau\alpha)_{ij} + \varepsilon_{ipj} \qquad (\text{MD4}_{ma})$$

where, compared to previous models:

- y_{ijp} is the score of the i-th topic and p-th query formulation for the j-th system;
- $\nu_{p(i)}$ is the effect of the $p(i)$-th query formulation;
- finally, ε_{ijp} is the error margin for the model in predicting y_{ijp}.

A visual depiction of this model is available in Figure 4.3a. Note that the query formulation factor $\nu_{p(i)}$ is nested within the i-th topic since query formulations are specific to each topic. A nested factor conceptually means that each query formulation can only exist as a "subcomponent" of a topic, which is the formal description of the information that a searcher intends to retrieve. For example, Figure 4.4 shows the full description of topic 656 from the TREC 2004 Robust test collection as described by Voorhees [180]. Note that NIST, by default, often provides a query for the topic (the title), and this is included as one of the many possible query formulations which are nested as a factor of this topic. Since every query formulation for this topic is *not* independent, it cannot be treated as a separate factor as ANOVA factors, by default, are always assumed to be independent.

So, for example, the j-th formulation "child protection laws for lead poisoning" and "lead poisoning children" are both possible query formulations for Topic 656, the latter being the title formulation provided by NIST. Considering them as nested factors allows to correctly model the variance since each formulation contributes only to the variance of a single topic. Nesting does not compare the j-th formulation of a specific topic against the j-th formulation of another one, which is analogous to the effect captured between two topics. This has the benefit of reducing computational costs while still modelling the topic effect itself.

Model (MD4$_{ma}$) extends the "classical" two-way ANOVA models of Banks et al. [12], Tague-Sutcliffe and Blustein [168] to study topic and system factors, as our initial goal is to add a query variation factor. Our adaptation also extends the model of Bailey et al. [7] so that we can observe and measure the topic*system interactions that are produced when using query variants as replicates for a topic, which is possible when query formulations are nested factors of topics. In this work, this model is used to investigate RQ 4.1.

```
<num> Number: 656
<title> lead poisoning children
<desc>
How are young children being protected against lead poisoning from
paint and water pipes?
<narr>
Documents describing the extent of the problem, including suits
against manufacturers and product recalls, are relevant. Descriptions
of future plans for lead poisoning abatement projects are also
relevant. Worker problems with lead are not relevant. Other poison
hazards for children are not relevant.
```

Fig. 4.4 Topic 656 as defined in the TREC 2004 Robust test collection.

As discussed previously, to decompose the component-wise contribution of the system factor α_k, we must apply a GoP. The following micro-level ANOVA model addresses RQ 4.1 by breaking down the component factors for a single corpus:

$$y_{ijqrst} = \mu + \tau_i + \nu_{p(i)} + \gamma_q + \delta_r + \zeta_s + \kappa_t + (\tau\gamma)_{iq} + (\tau\delta)_{ir} + (\tau\zeta)_{is} + (\tau\kappa)_{it} + \varepsilon_{ipqrst} \quad (\text{MD}4_{mi})$$

where, with respect to model (MD4$_{ma}$), the system factor α_k is replaced by its component-wise decomposition:

- γ_q is the effect of the q-th stop list;
- δ_r is the effect of the r-th stemmer;
- ζ_s is the effect of the s-th IR model;
- κ_t is the effect of the t-th query expansion;
- $(\tau\gamma)_{iq}$ is the interaction between topics and stop lists;
- $(\tau\delta)_{ir}$ is the interaction between topics and stemmers;
- $(\tau\zeta)_{is}$ is the interaction between topics and IR models;
- $(\tau\kappa)_{it}$ is the interaction between topics and query expansion.

Model (MD4$_{mi}$) extends the model proposed by Ferro and Silvello [64, 65] to decompose the component effects. The resulting model has a nested query formulation factor and captures traditional interactions between topics and components. The model also supports query expansion models as a factor, which was not explored by Ferro and Silvello previously.

Multiple Topic Formulations - Multiple Corpora ANOVA Models

Finally, we are interested in observing and measuring the effect of jointly changing the corpus of documents used and the way in which we formulate the topic. To do so, we employ two models: MD5$_{ma}$ and MD5$_{mi}$.

The macro-level ANOVA model is used to investigate RQ 4.2 and RQ 4.3 on multiple corpora:

$$y_{ijkp} = \mu + \tau_i + \nu_{p(i)} + \alpha_j + \beta_k + (\tau\alpha)_{ij} + (\tau\beta)_{ik} + (\alpha\nu)_{jp(i)} + (\alpha\beta)_{jk} + (\beta\nu)_{kp(i)} + (\tau\alpha\beta)_{ijk} + \varepsilon_{ijkp} \tag{MD5$_{ma}$}$$

where, with respect to models (MD3) and (MD4$_{ma}$), it adds:

- $(\beta\nu)_{jp(i)}$ is the interaction between corpora and query formulations;
- $(\tau\alpha\beta)_{ikp}$ is the interaction between topics, systems, and corpora.

Multiple formulations for an information need enable the computation of the above-mentioned interaction factors by acting as replicates.

Model (MD5$_{ma}$) is a combination of several models that have been used recently [198, 59, 62, 61, 184], and also extends the model by Zampieri et al. [198] in order to cover all of the new interactions that are created when nesting query formulations. The models of Ferro et al. [59], Ferro and Sanderson [62, 61], Voorhees et al. [184] were also extended so that the query formulations can be included in addition to all of the resulting cross-factor interactions. The design of experiments underlying this model is depicted in Figure 4.3b.

The micro-level ANOVA model used to address RQ 4.2 and RQ 4.3 by breaking down the system factor component-wise has the following formulation:

$$\begin{aligned} y_{ijpqrst} = {} & \mu + \tau_i + \nu_{p(i)} + \beta_k + \gamma_q + \delta_r + \zeta_s + \kappa_t + (\tau\beta)_{ik} + (\beta\nu)_{kp(i)} + (\tau\gamma)_{iq} + (\tau\delta)_{ir} + \\ & (\tau\zeta)_{is} + (\tau\kappa)_{it} + (\beta\gamma)_{kq} + (\beta\delta)_{kr} + (\beta\zeta)_{ks} + (\beta\kappa)_{kt} + (\gamma\nu)_{qp(i)} + (\delta\nu)_{rp(i)} + \\ & (\zeta\nu)_{sp(i)} + (\kappa\nu)_{tp(i)} + (\tau\beta\gamma)_{ikq} + (\tau\beta\delta)_{ikr} + (\tau\beta\zeta)_{iks} + (\tau\beta\kappa)_{ikt} + (\beta\gamma\nu)_{kqp(i)} + \\ & (\beta\delta\nu)_{krp(i)} + (\beta\zeta\nu)_{ksp(i)} + (\beta\kappa\nu)_{ktp(i)} + \varepsilon_{ikpqrst} \end{aligned} \tag{MD5$_{mi}$}$$

where, with respect to model (MD5$_{ma}$) and (MD4$_{mi}$):

- $(\beta\gamma)_{kq}$ is the interaction between corpora and stop lists;
- $(\beta\delta)_{kr}$ is the interaction between corpora and stemmers;
- $(\beta\zeta)_{ks}$ is the interaction between corpora and IR models;
- $(\beta\kappa)_{kt}$ is the interaction between corpora and query expansion;

- $(\gamma\nu)_{qp(i)}$ is the interaction between stop lists and query formulations;
- $(\delta\nu)_{rp(i)}$ is the interaction between stemmers and query formulations;
- $(\zeta\nu)_{sp(i)}$ is the interaction between IR models and query formulations;
- $(\kappa\nu)_{tp(i)}$ is the interaction between query expansion and query formulations;
- $(\tau\beta\gamma)_{ikq}$ is the interaction between topics, corpora, and stop lists;
- $(\tau\beta\delta)_{ikr}$ is the interaction between topics, corpora, and stemmers;
- $(\tau\beta\zeta)_{iks}$ is the interaction between topics, corpora, and IR models;
- $(\tau\beta\kappa)_{ikt}$ is the interaction between topics, corpora, and query expansion;
- $(\beta\gamma\nu)_{kqp(i)}$ is the interaction between corpora, stop lists, and query formulations;
- $(\beta\delta\nu)_{krp(i)}$ is the interaction between corpora, stemmers, and query formulations;
- $(\beta\zeta\nu)_{ksp(i)}$ is the interaction between corpora, IR models, and query formulations;
- $(\beta\kappa\nu)_{ktp(i)}$ is the interaction between corpora, query expansion.

Model (MD5_{mi}) extends the models proposed by Zampieri et al. [198], Ferro and Silvello [64, 65] to account for topics, query formulations and corpora interactions between all of the system components.

4.4.2 Data and Methods

We used the following collections: TREC Robust 04 Ad Hoc [180], TREC Common Core 17 [2], and TREC Common Core 18 [3] for our experiments. The Robust 04 Ad Hoc track used Disk 4 and 5 of the TIPSTER corpus minus the Congressional Record sub-collection and contains approximately 528K documents; the TREC Common Core 17 track used the New York Times Annotated Corpus which contains over 1.8 million articles; finally, the TREC Common Core 18 track used the Washington Post corpus, roughly containing 600K news articles.

A large seed set of human curated query formulations originally developed using the TREC Robust 04 Ad Hoc search collection were used in our experiments [15][1]. These were further enriched with query reformulations extracted from a Bing search log and mapped to the original 249 topic descriptions [101].

[1] http://culpepper.io/publications/robust-uqv.txt.gz

Table 4.1 Summary statistics of the collections used. The column ‘Shared’ contains statistics on topics which overlap in all three collections.

	Core 17	**Core 18**	**Robust 04**	**Shared**
# of documents	$1,855.658$	$595,037$	$528,155$	-
# of topics	50	50	250	25
total # of formulations per corpus	1286	625	3402	625
avg # of formulations per topic	25.72	25.0	13.61	25.0
min # of formulations per topic	20	20	9	20
max # of formulations per topic	53	40	53	40
avg # of words per formulation	4.8	4.7	5.3	4.7
min # of words per formulation	1	1	1	1
max # of words per formulation.	15	11	17	11

In 2017 and 2018, TREC ran the CORE track which reused many of the original topics from the Robust 04 exercise. There are 50 overlapping topics in Core 17 and 25 overlapping topics in 2018. To provide comparable results across multiple corpora, we use only the subset of topics which overlap in all three of the corpora. Thus, in the following experiments, we considered 625 query formulations for the 25 overlapping topics and not all of the 3,402 variations available for the 249 topics in Robust 04. Note that one of the original Robust 04 topics has no relevant documents in QREL judgments created by NIST, and is therefore omitted from consideration. All of the collections use graded relevance judgments, with the 3 grades being: not relevant, relevant, and highly relevant; we mapped to binary relevance judgments by using a lenient approach, i.e. everything above not relevant is considered as relevant since Average Precision (AP) is used for all evaluation comparisons in this work [25]. Table 4.1 provides additional statistical information on the collections used.

As discussed in Chapter 2, previous work has simulated collection effects by splitting a collection into shards or sub-collections [61, 59, 198]. The TREC CORE tasks were ran in 2017/2018 and reused topics originally created for the TREC Ad Hoc tasks between 2002 and 2005. Thus, they allow us to compare system performance across multiple corpora using the same set of topics and identical system configurations. This allows us to compute the effect of the collection, without simulating it and ensures that the system component factors can be the same for every collection being used. The underlying collection was composed primarily of news articles in all three TREC campaigns, but changed from the original Newswire collection in 2004 to the New York Times collection in 2017 and then to the Washington Post collection in 2018.

Table 4.2 Terrier Retrieval and Query Expansion Models.

Model	Description
BM25	Okapi BM25.
DPH	The parameter-free hyper-geometric divergence from randomness (DFR) model using Popper's normalization.
Hiemstra_LM	Hiemstra's language model.
In_expB2	Inverse expected document frequency model for randomness, the ratio of two Bernoulli's processes for first normalisation, and Normalisation 2 for term frequency normalisation.
Js_KLs	A weighted combination of Jeffreys divergence and Kullback Leibler divergence.
TF_IDF	The TF×IDF weighting function, using Robertson's TF and the IDF of Sparck Jones.
PL2	Poisson estimation for randomness, Laplace succession for first normalisation, and Normalisation 2 for term frequency normalisation .
TF_IDF_DRF	Same as above with a pBiL DFR term dependency model [124] enabled and a window size of 5.
Hiemstra_LM_DRF	Same as above with a pBiL DFR term dependency model [124] enabled and a window size of 5.
BA	The approximation of the binomial distribution using the Kullback-Leibler divergence to induce the weighted query terms during expansion.
Bo1	The Bose-Einstein 1 DFR expansion technique.
KL	Kullback-Leibler divergence based query expansion.

For this set of experiments, we used a modified version of the Terrier Search engine (version 5.1) to create our GoP. The modifications were required in order to maximize the diversity of components (stemmers and ranking models) available for our experimental setup. Based on a few preliminary runs for multiple system configurations on each collection, we selected 9 retrieval ranking models and 3 query expansion models (plus the no query expansion), which are described in Table 4.2. Moreover, runs were built using 4 different stemmers: Krovetz, Porter, S-Stemmer, and Lovins. Finally, we doubled the number of available runs by either keeping or removing the stop words.

Stemmers, Retrieval Models, and Query Expansion Models have been chosen to maximize the variety in our system configurations, in terms of overall effectiveness and the documents retrieved. The total number of available configurations was 288, which have been applied on each corpus, giving us a total of 864 runs. To aid reproducibility in the future, data and runs are publicly available.[2]

[2] https://github.com/jsc/anova-query_formulations

4.5 Validation of the Experimental Setup

We perform a preliminary inspection of our dataset in order to verify that our GoP has a performance distribution comparable to typical runs submitted to TREC and that query formulations are not skewed or biased in any specific way.

4.5.1 Validation of the Grid of Points

We now investigate how close the performance distribution of the original systems submitted to that TREC track is to the performance distribution of the GoP systems on the same track. To quantify this "closeness" we use the Kullback-Leibler Divergence (KLD) [92] between the two performance distributions. In order to compute the KLD, we need the Probability Density Function (PDF) of the performance distributions, which we estimate by using a Kernel Density Estimation (KDE) [189] approach.

Given a vector X of m elements, the KDE estimation of the PDF is given by

$$\hat{f}_X(x) = \frac{1}{mb} \sum_{i=1}^{m} K\left(\frac{x - X_i}{b}\right) \tag{4.1}$$

where X_i is the i-th component of the array, b is the *bandwidth* or *window width* and is greater than 0; $K(\cdot)$ is the *kernel* satisfying $\int_{-\infty}^{+\infty} K(x)dx = 1$. In this work, we use a Gaussian kernel with bandwidth $b = 0.015$.

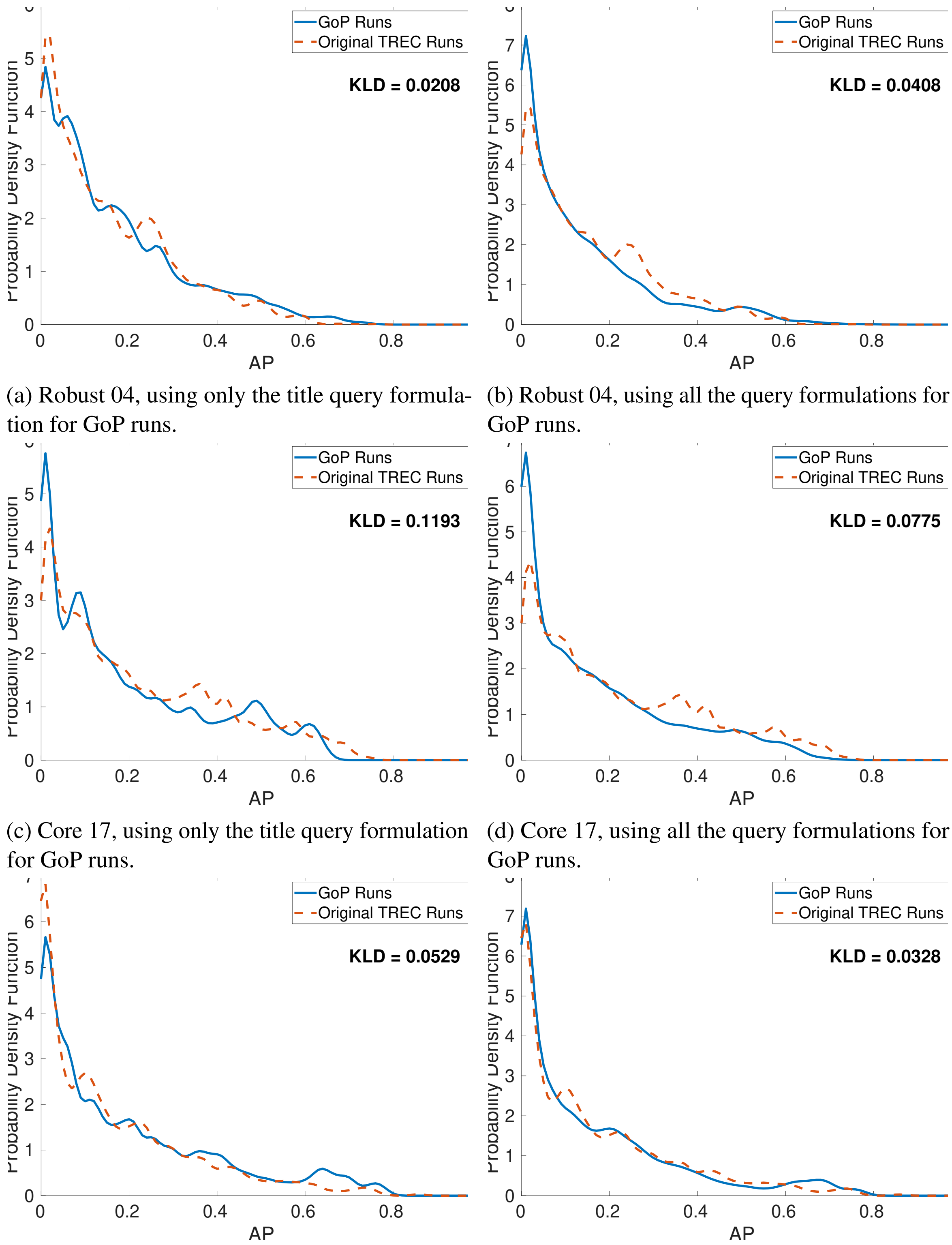
Given two m element vectors X and Y, the KLD between the PDFs is given by

$$D_{KL}\big(X\big|\big|Y\big) = \sum_{x} \ln\left(\frac{\hat{f}_X(x)}{\hat{f}_Y(x)}\right) \hat{f}_X(x) \tag{4.2}$$

Note that D_{KL} is not symmetric and so, in general, $D_{KL}\big(X\big|\big|Y\big) \neq D_{KL}\big(Y\big|\big|X\big)$.

As initially proposed by Burnham and Anderson [27], $D_{KL} \in [0, +\infty)$ denotes the information lost when Y is used to approximate X; in our context, it denotes the information lost when the GoP systems are used to "approximate" an original set of systems submitted to a TREC track. Therefore, 0 means that there is no loss of information and, in our context, that the original systems and the GoP ones are considered the same; $+\infty$ means that there is a full loss of information and, in our context, that the original systems and the GoP ones have no similarity.

Note that the TREC runs may have used the title, the description, and/or the narrative of a topic, as well as manual formulations for that topic, but generally the title is the most commonly used field. Therefore, we have chosen to compare TREC runs to our GoP runs



(a) Robust 04, using only the title query formulation for GoP runs.

(b) Robust 04, using all the query formulations for GoP runs.

(c) Core 17, using only the title query formulation for GoP runs.

(d) Core 17, using all the query formulations for GoP runs.

(e) Core 18, using only the title query formulation for GoP runs.

(f) Core 18, using all the query formulations for GoP runs.

Fig. 4.5 Comparison between AP score distribution of GoP runs and the original TREC runs. A small divergence between the original scores distributions and the scores achieved using the GoP can be observed in all six plots.

using only the title of a topic, as shown in Figure 4.5 on the left. Moreover, to validate that the other query formulations do not introduce any specific bias, in Figure 4.5 on the right, we compare the original TREC runs with respect to our GoP using all the query formulations and verify that the distributions have a similar composition.

In all of the comparisons in Figure 4.5, the estimated distributions are very similar and KLD is small. This indicates that our data mimics the behavior of the original TREC runs. Moreover, in the case when all the query formulations are used, the distributions are still very close, suggesting that the query formulation did not distort the outcomes.

4.5.2 Validation of the Query Formulations

In the boxplot shown in Figures 4.6, 4.7, we consider the performance of each query reformulation across all GoP systems; for each topic, there is a box corresponding to each corpus; the performance of the title query is highlighted with a diamond. In Figure 4.6 on the left we use, as aggregation statistics, the average of AP, also referred to as Average Average Precision (AAP) by Mizzaro and Robertson [113]; in Figure 4.7 on the right we use the median of AP as aggregation statistics.

From Figures 4.6, 4.7, we can see that there is no topic for which all the reformulations perform similarly on all corpora. Indeed, the performance distribution of the topics vary widely when using the query formulations and are even less predictable when changing the underlying corpus. However, topics 442 and 690 do have similar distributions across all corpora, suggesting that their easiness or difficulty is more stable than the others. Overall, this provides some visual intuition of how unstable topic effectiveness is when the corpora and the query formulation varies.

In general, the query formulations appear to be high quality. More concretely, in both figures, the red diamond indicates either the AAP or the median AP achieved by the title formulation for the specific TREC topic over all systems, and we can observe that even though the title formulation is often in the top quartile of possible outcomes, there are many cases where the title formulation performs poorly compared to many of the reformulations, and may even be the worst performing one. It is also interesting to observe that, even though a query formulation might perform particularly well on a one corpus, often it does not perform equally well on another one. For example, in the case of topic 378 the title formulation is the best performing formulation on the Core 17 corpus but it falls in the lowest quartile in the Core 18 corpus and is one of the worst formulations in Robust 04. Given that all three collections are essentially news documents, it is somewhat surprising that the performance is so volatile given that the same components and ranking functions are being used. That is, the search engine is fixed, but the query and documents searched are not.

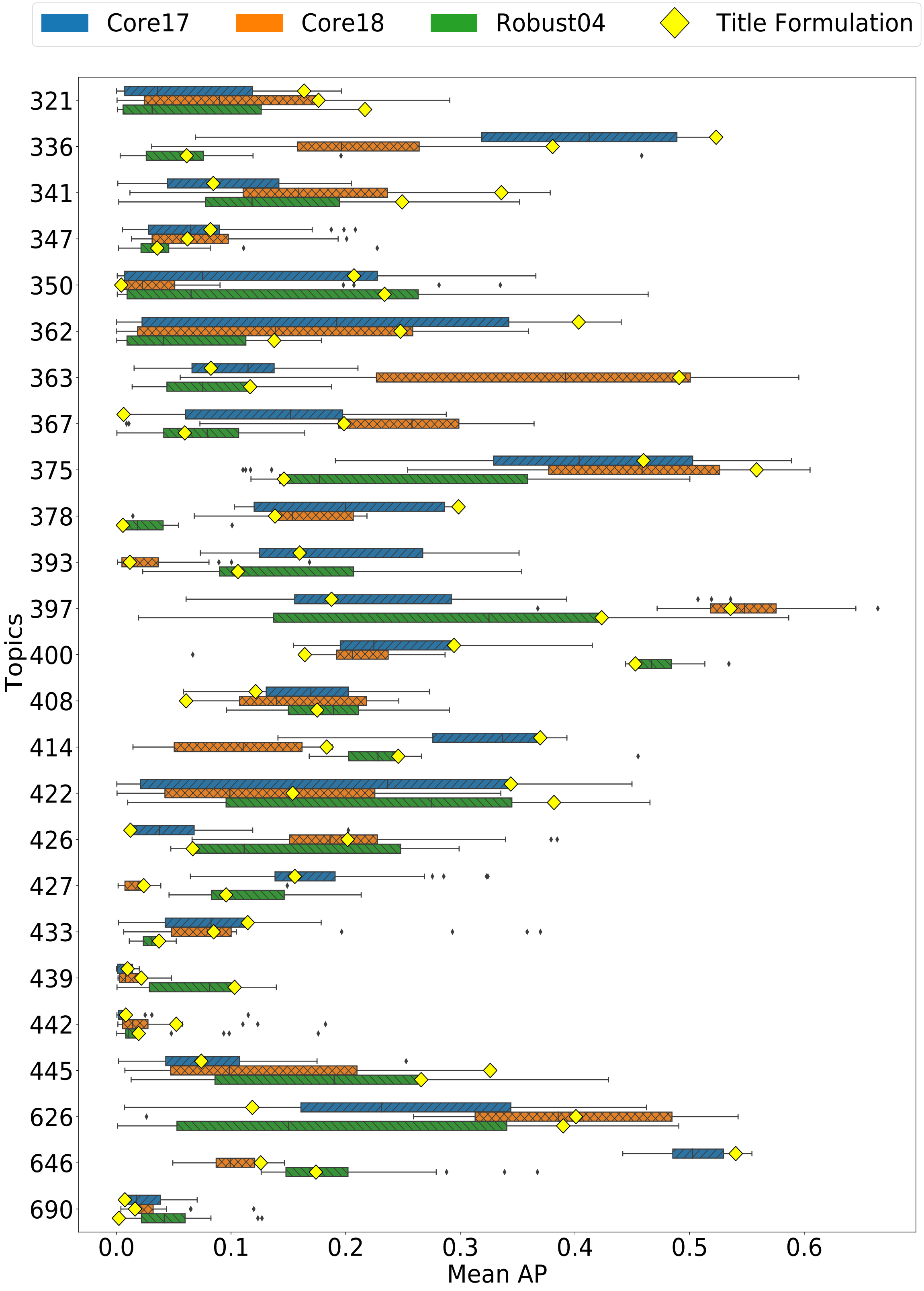


Fig. 4.6 Distribution of the Average AP of the different query formulations over all GoP systems. For each topic, there is a box corresponding to each corpus; the performance of the title query is denoted with a diamond.

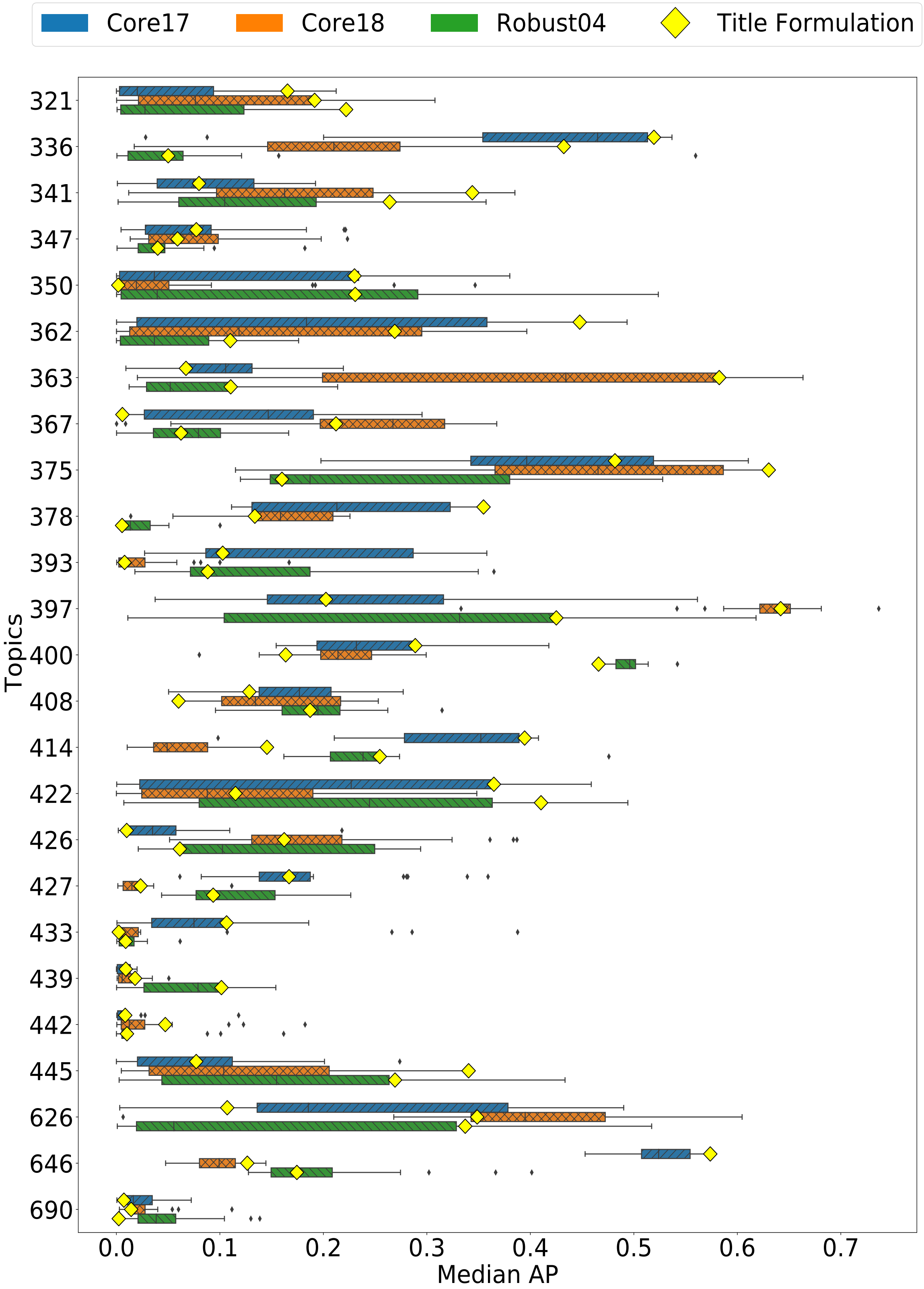


Fig. 4.7 Distribution of the Median AP of the different query formulations over all the GoP systems. For each topic, there is a box corresponding to each corpus; the performance of the title query is denoted with a diamond.

Table 4.3 A summary of the effect sizes for factors in MD4$_{ma}$ for all three collections. Blue represents the size of the factor, where dark blue is large and light blue is medium. For all three corpora, observe that the majority of the factors have a large size effect. The only medium size factor in two collections (Robust 04 and Core 18) is the System. Furthermore, observe that the Topic*System interaction has a large size effect, which indicates that system configuration and topic performance are correlated, and supports the hypothesis that the "topic difficulty" is linked to the system used and not the query formulation.

	Robust 04	**Core 17**	**Core 18**
Topic	0.7639	0.8215	0.7834
Formulations (Topic)	0.6941	0.6833	0.6038
System	0.1080	0.2193	0.1445
Topic*System	0.3385	0.3510	0.4386

Now consider the possible outcomes of the greedy algorithm discussed in Section 4.3 and the different cases shown in Figure 4.2, Figure 4.2b, which were the easiest case, and where the minimum of a topic is below the maximum of another topic respectively. Figures 4.6 and 4.7 clearly show that this rarely happens in our experiments. Instead we typically observe the more complex patterns exhibited in Figure 4.2c and 4.2d. As a consequence, the corner case where a really poor query formulation for an "easy" topic performs more poorly than a very good query formulation for a "difficult" topic, which would correspond to a very large bar (the easy topic) whose bottom is below the top of a very narrow bar (the difficult topic), was not observed in these experiments.

4.6 RQ 3.1: Query Formulation Effect Size within Corpora

4.6.1 Macro-Level ANOVA

Due to the large number permutations and the memory constraints imposed by the underlying ANOVA model, we randomly sample 18 query formulations for each topic in the following analysis.

Table 4.3 provide a summary of the effect size for each factor, for model (MD4$_{ma}$) using each corpus, where we observe similar performance trends across all three. Table 4.4, Table 4.5, and Table 4.6 contains the complete ANOVA statistics for model MD4$_{ma}$ respectively on Robust 04, Core 17, and Core 18.

All factors were found to be statistically significant. Consistently with the previous findings of Tague-Sutcliffe and Blustein [168], the topic factor has a large-size effect size, and it is indeed the largest effect for this configuration. We can also clearly see that query

Table 4.4 Model ($MD4_{ma}$) on track Robust 04 for AP. The letter in the column "effect size" indicates whether the effect is large (L) or medium (M). On Robust 04 the effect size of all factors is large, except for the System factor, which has a medium size effect. The topic (information need) and its formulations are the most prominent effect, followed by the interaction between the topic and the system. A large effect for the interaction indicates that some systems are better for specific topics while, for other systems, its the other way around.

Source	SS	DF	MS	F	p-value	$\hat{\omega}^2_{\langle fact \rangle}$	effect size
Topic	1389.69	24	57.90	17471.22	<1e-6	0.7639	L
Formulations (Topic)	976.17	425	2.30	693.03	<1e-6	0.6941	L
System	52.97	287	0.18	55.69	<1e-6	0.1080	M
Topic*System	242.63	6888	0.04	10.63	<1e-6	0.3385	L
Error	404.26	121975	<1e-2				
Total	3065.72	129599					

Table 4.5 Model ($MD4_{ma}$) on track Core 17 for AP. The letter in the column “effect size” indicates if the effect is large (L). For Core 17, all factors have a large size effect. The topic (information need) has an effect that is 3.68 times larger than the system effect. The topic formulation on the other hand has an effect much larger than the system effect. The interaction effect is again very large.

Source	SS	DF	MS	F	p-value	$\hat{\omega}^2_{\langle fact \rangle}$	effect size
Topic	2073.88	24	86.41	24852.00	<1e-6	0.8215	L
Formulations (Topic)	973.70	425	2.29	658.91	<1e-6	0.6833	L
System	127.59	287	0.44	127.86	<1e-6	0.2193	L
Topic*System	267.65	6888	0.04	11.18	<1e-6	0.3510	L
Error	424.11	121975	<1e-2				
Total	3866.94	129599					

formulations also have a large effect size in our experiments – approaching the topic effect size – suggesting that query formulations strongly influence topic difficulty. In prior work, Bailey et al. [7] also observed that query formulation had an effect in an ANOVA analysis, but their ANOVA used a different nesting of factors than ours and was based on just two systems; this may have had an impact on their reported topic effect, which was a medium-effect size and differs from all other previous literature where topic effect consistently has a high-effect size.

The system factor has a medium to large-size effect and on Core 17 almost double the size of Robust 04 and Core 18. This suggests that the interaction between system and corpus can play a role, as will be investigated in subsequent analyses.

Another interesting observation in our analysis is the topic*system interaction effect size in Table 4.3, which is large, and confirms an important supposition of Banks et al. [12],

Table 4.6 Model (MD4_{ma}) on track Core 18 for AP. The letter in the column "effect size" indicates whether the effect is large (L) or medium (M). On Core 18 Effects sizes are close to the one observed for the Robust 04 collection. We have very large effects for the topic and its formulations, a medium-large effect for the system and a large effect for the interaction between the system and the topic. Compared to the Robust, we observe an even larger effect for the interaction.

Source	**SS**	**DF**	**MS**	**F**	**p-value**	$\hat{\omega}^2_{\langle fact \rangle}$	**effect size**
Topic	2371.78	24	98.82	19532.05	<1e-6	0.7834	L
Formulations (Topic)	1001.38	425	2.36	465.69	<1e-6	0.6038	L
System	112.21	287	0.39	77.28	<1e-6	0.1445	M
Topic*System	547.15	6888	0.08	15.70	<1e-6	0.4386	L
Error	617.14	121975	0.01				
Total	4649.67	129599					

who were only able to provide a rough estimate for the topic*system interaction. While Banks et al. [12] suspected that topic*system interactions should be large, they were not able to actually confirm it due to an insufficient number of replicates in their experimental setup. More recently, Ferro et al. [59] and Voorhees et al. [184], who used collection shards to obtain the necessary replicates required to estimate the topic*system interaction effect size, found that it does indeed have a large-size effect. In our configuration, the replicates necessary to estimate this effect are provided by different query formulations which, to the best of our knowledge, have not been used for this purpose in previous work. This further confirms the prominence of the existence of this effect when using our proposed experimental design.

Overall, query formulation has the second largest effect, with nearly 1.5 times the size of the topic*system interaction which has historically been a point of emphasis in similar performance comparisons. This provides important evidence that query formulation is crucial in retrieval effectiveness, and has deeper implications in rethinking the way many IR experiments currently formalize query / topic difficulty. Topic difficulty for query performance prediction can be viewed at an abstract level as an attempt to predict topic*system interactions, and indeed the quality of these methods are often measured using a Kendall tau of the *ordering* of topics by effectiveness for any given set of topics. This is only possible if certain topics consistently perform better than others, and having a mix of “easy”, “medium”, and “difficult” tends to provide the most desirable signal. But what if there were *no* difficult topics? Do such scenarios exist given our ability to reformulate queries based on a collection and the surprisingly large factor size observed? We will explore this intriguing question in more detail in Sections 4.7 and 4.8.

4.6.2 Micro-Level ANOVA

Table 4.7 A summary of effect sizes for factors when using MD4$_{mi}$ on the three collections. The shade of blue indicates the factor size – large being dark blue, and medium or small as lighter shades of blue. The white cells are the factors with significant but negligible effects sizes. The topic and the formulation, as well as all of the micro-components have smaller effects than a system treated as a whole. The two most prominent effects in a system are the Retrieval Model and the Query Expansion Model. It is interesting to note that the Query Expansion Model has different effect sizes that depend on the corpus. Furthermore, the majority of the interactions between the Topic and the various components have medium to large effect sizes. Stopping versus not stopping has a negligible effect, both alone and in interaction with the Topic.

	Robust 04	**Core 17**	**Core 18**
Topic	0.7560	0.8116	0.7743
Formulation	0.6848	0.6687	0.5910
Stoplist	0.0007	0.0020	0.0013
Stemmer	0.0043	0.0024	0.0060
Model	0.0661	0.0728	0.0197
Query Expansion	0.0232	0.1377	0.1114
Topic*Stoplist	0.0073	0.0026	0.0033
Topic*Stemmer	0.0709	0.0355	0.0673
Topic*Model	0.2158	0.2356	0.1677
Topic*Query Exp.	0.3153	0.1029	0.2958

In order to better understand the impact of query formulation at the component level, we have also performed a detailed component-wise ANOVA analysis using model (MD4$_{mi}$), whose results are reported in Table 4.7. Additional statistics and details we used to create the summary table are also included here in the ANOVA tables for the model MD4$_{mi}$ on Robust 04 (Table 4.8), Core 17 (Table 4.9), and Core 18 (Table 4.10). See these tables for additional information.

Again, the results are consistent across all the three collections. Stop lists and stemmers have a small-size effect – stop lists were not even significant on Robust 04– which was not true in experiments ran by Ferro and Silvello [64, 65], who both reported a medium-size effect for these factors. We also found, consistent with previous work, that the IR model factor has a small to medium-size effect at the micro-level. These results are aligned with Zampieri et al. [198] who also observed that stemmers and IR models have a small-size effect, albeit two orders of magnitude larger in our configuration.

Query expansion on the other hand has a small-size to medium-size effect and is the largest among all system component factors for Core 17 and Core 18, which differ from

Table 4.8 Model (MD4$_{mi}$) on track Robust 04 for AP. The letter in the column "effect size" indicates whether the effect is large (L), medium (M), small (S) or not significant (-). Between parentheses significant yet almost negligible effects. Among the different factors, only the Topic and the formulations have a large effect. The Retrieval model has a medium effect while both the Stemmer and Query Expansion model have a small size effect. Even though significant, removing or not the stopwords and the stemmer have a very small effect. Although the interaction between topics and components is always significant, we observe variations on the different effect sizes. We observe a large interaction only with the Retrieval model. The interactions between the topic and either the Query Expansion model and the Stemmer have medium size effects.

Source	SS	DF	MS	F	p-value	$\hat{\omega}^2_{\langle fact \rangle}$	effect size
Topic	1389.69	24	57.90	16729.19	<1e-6	0.7560	L
Form. (Topic)	976.17	425	2.30	663.59	<1e-6	0.6848	L
Stoplist	0.33	1	0.33	93.92	<1e-6	0.0007	(S)
Stemmer	1.94	3	0.65	186.54	<1e-6	0.0043	(S)
Model	31.77	8	3.97	1147.20	<1e-6	0.0661	M
Query Expansion	10.78	3	3.59	1038.29	<1e-6	0.0234	S
Topic*Stoplist	3.38	24	0.14	40.65	<1e-6	0.0073	(S)
Topic*Stemmer	34.50	72	0.48	138.44	<1e-6	0.0709	M
Topic*Model	124.10	192	0.65	186.74	<1e-6	0.2158	L
topic*Query Exp.	47.35	72	0.66	189.98	<1e-6	0.0950	M
Error	445.72	128775	<1e-2				
Total	3065.72	129599					

Zampieri et al. [198] who observed a very small small-size effect for this factor. Note that we have incorporated query formulations in our comparison, and the combination of query reformulations and query expansion is the most likely contributor to differences in effect sizes we have observed. We revisit this hypothesis in the next section as we will be in a better position to measure it directly in our final model configuration.

As previously discussed, we were also able to reliably estimate the interaction effect sizes between topics and system components for the first time. In particular, topic*query expansion interaction has a notably large-size effect, followed by the topic*IR model interaction. The interaction with stemmers had a small to medium-size effect while stop lists had a very small-size effect. Note that low IDF terms are dropped when query expansion is enabled as stop words tend to have a negative impact on system effectiveness when they are not removed. In summary, our findings indicate that query expansion and IR models are the components most affected by topics, and this could provide useful hints when debugging and diagnosing which system components to target in order to improve performance.

Table 4.9 Model (MD4_{mi}) on track Core 17 for AP. The letter in the column "effect size" indicates whether the effect is large (L), medium (M) or small (S). Between parentheses significant yet almost negligible effects. In the case of Core 17, we observe that the factors with large size effects are the topic, the formulations, and the Query Expansion Model. Even though large size, the Query Expansion model has an effect that is 6 time smaller than the topic and 5 times smaller than the formulations. Both stoplists and stemmers are significant, yet have a negligible effect. The Retrieval model has a medium size effect. As previously observed, all the interactions between the topic and the components are significant.

Source	SS	DF	MS	F	p-value	$\hat{\omega}^2_{\langle fact \rangle}$	effect size
Topic	2073.88	24	86.41	23256.87	<1e-6	0.8116	L
Form. (Topic)	973.70	425	2.29	616.62	<1e-6	0.6687	L
Stoplist	0.97	1	0.97	261.72	<1e-6	0.0020	(S)
Stemmer	1.18	3	0.39	106.14	<1e-6	0.0024	(S)
Model	37.84	8	4.73	1273.00	<1e-6	0.0728	M
Query Expansion	76.96	3	25.65	6902.16	<1e-6	0.1377	M
Topic*Stoplist	1.34	24	0.06	15.02	<1e-6	0.0026	(S)
Topic*Stemmer	17.99	72	0.25	67.24	<1e-6	0.0355	S
Topic*Model	149.14	192	0.78	209.06	<1e-6	0.2356	L
Topic*Query Exp.	55.49	72	0.77	207.42	<1e-6	0.1029	M
Error	478.47	128775	<1e-2				
Total	3866.94	129599					

4.7 Query Formulation Effect Size across Corpora

In this section, we expand our analysis using Models (MD5_{ma}) and (MD5_{mi}) which can be used to measure cross collection effects, and address RQ 4.2 and RQ 4.3. Note that the computational complexity of these two models in our current configuration is substantial, and therefore the analysis was carried using only the two best performing stemmers – Porter and Krovetz based on our initial analysis. For the same reason, we also limit ourselves to 15 query formulations for each topic.

4.7.1 Macro-Level ANOVA

Table 4.11 shows the results for model (MD5_{ma}) using all three corpora – Robust 04, Core 17 and Core 18. All the factors are again statistically significant.

We can observe that the topic factor has, as always, a large-size effect even across corpora and that the system factor becomes a moderately large-size effect, being bigger than in the single corpus case (see Table 4.3); the corpus factor has a medium-size effect. Overall, these results support similar findings reported by Ferro and Sanderson [61] while Zampieri et al.

Table 4.10 Model ($MD4_{mi}$) on track Core 18 using AP. The letter in the column "effect size" indicates whether the effect is large (L), medium (M) or small (S). In the case of Core 18, the only factors with large size effects are the topic and the formulations. Both stoplists and stemmers are significant, but have a negligible effect size. The Retrieval model has a small effect on system performance, and Query Expansion has a medium size effect. As previously observed, all of the interactions between the topic and the components are significant, but the interaction between stoplists and topics have a negligible effect sizes, and interaction between the retrieval and query expansion models are large.

Source	SS	DF	MS	F	p-value	$\hat{\omega}^2_{\langle fact \rangle}$	effect size
Topic	2371.78	24	98.82	18524.14	<1e-6	0.7743	L
Form. (Topic)	1001.38	425	2.36	441.66	<1e-6	0.5910	L
Stoplist	0.93	1	0.93	174.16	<1e-6	0.0013	(S)
Stemmer	4.16	3	1.39	259.65	<1e-6	0.0060	(S)
Model	13.92	8	1.74	326.08	<1e-6	0.0197	S
Query Expansion	86.67	3	28.89	5415.26	<1e-6	0.1114	M
Topic*Stoplist	2.41	24	0.10	18.81	<1e-6	0.0033	(S)
Topic*Stemmer	50.25	72	0.70	130.83	<1e-6	0.0673	M
Topic*Model	140.35	192	0.73	137.02	<1e-6	0.1677	L
Topic*Query Exp.	290.82	72	4.04	757.14	<1e-6	0.2958	L
Error	687.00	128775	0.01				
Total	4649.67	129599					

[198] reported that both systems and corpora had a very small-size effects. We also note that the query formulation factor has a remarkably large-size effect, even across corpora, observed here for the first time, suggesting it is a key contributor to topic difficulty.

The topic*system interaction has a noticeably large-size effect but half the size of the query formulation factor alone, and roughly two-thirds of the effect observed in the single corpus case; in addition, the query formulation*system interaction is a medium (almost large) size effect. Overall, this suggests that the multiple corpora further amplify the impact of query formulations, which was already very large. Note that the size of the topic*system interaction reaffirms similar findings from Zampieri et al. [198], Ferro et al. [59].

The topic*corpus interaction also has a large-size effect, the second biggest effect, which is aligned with the findings of Zampieri et al. [198], Ferro et al. [59]. Moreover, both the query formulation*corpus interaction and the topic*system*corpus interaction, observed here for the first time, are clearly important large-size effects.

Finally, we note that the system*corpus interaction is a medium-size effect, in contrast to previous results by Zampieri et al. [198] and Ferro et al. [59] who found it to have a

Table 4.11 Model (MD5$_{ma}$) for the Robust 04, Core 17, and Core 18 tracks and AP. The letter in the column "effect size" indicates whether the effect is large (L), medium (M) or small (S). Observe that the majority of the factors have large/medium effect sizes. The corpus has a medium size effect. The interaction size between the corpus factor and the topic or formulations are of particular interest – both of which are large. This is further empirical evidence that topic difficulty is not a result of th information need: searching for a piece of information in specific corpora or using different formulations on different corpora can result in very different performance. Furthermore, this suggests that we are likely to find a specific formulation for which we achieve better (or worse) performance for any topic on any corpus with less effort than would be required when attempting to achieve similar performance differences by changing only the ranker.

Source	SS	DF	MS	F	p-value	$\hat{\omega}^2_{\langle fact \rangle}$	effect size
Topic	1544.69	24	64.36	22370.77	0	0.7682	L
Formulation (Topic)	804.82	350	2.30	799.25	0	0.6330	L
System	87.80	143	0.61	213.41	0	0.1579	L
Corpus	33.04	2	16.52	5742.72	0	0.0662	M
System*Topic	216.04	3432	0.06	21.88	0	0.3067	L
System*Form.	240.36	50050	0.00	1.67	0	0.1713	L
System*Corpus	28.57	286	0.10	34.72	0	0.0562	S
Topic*Corpus	960.75	48	20.01	6956.96	0	0.6733	L
Form.*Corpus	527.09	700	0.75	261.72	0	0.5298	L
Topic*System*Corpus	214.42	6864	0.03	10.86	0	0.2946	L
Error	287.99	100100	0.00				
Total	4945.58	161999					

small-size effect. This behavior could be attributed to the presence of query reformulations in our model which increase the variance in performance for systems on different corpora.

Overall, these findings provide further evidence supporting the possibility that difficult topics do not actually exist in any absolute sense. We will further investigate this notion in Section 4.8 where the algorithm described in Section 4.3 leverages the large-size of the above interaction effects to show that it is actually possible to find any desired ranking of topics, providing further evidence that difficulty can not be confidently attributed to a particular topic.

Figure 4.8 provides a visualization of the volatility of topic difficult as collection and query formulation change. The red, yellow, and green bands in the figure correspond to the hard, medium, and easy query performance ranges, according to the traditional definition proposed in [34], where the 38% of the worst-performing queries have been considered hard, the 30% medium-performing queries have been defined medium, while the upper 30% of queries are the easy ones. On the left of Figure 4.8 we can see the plot of the topic*corpus

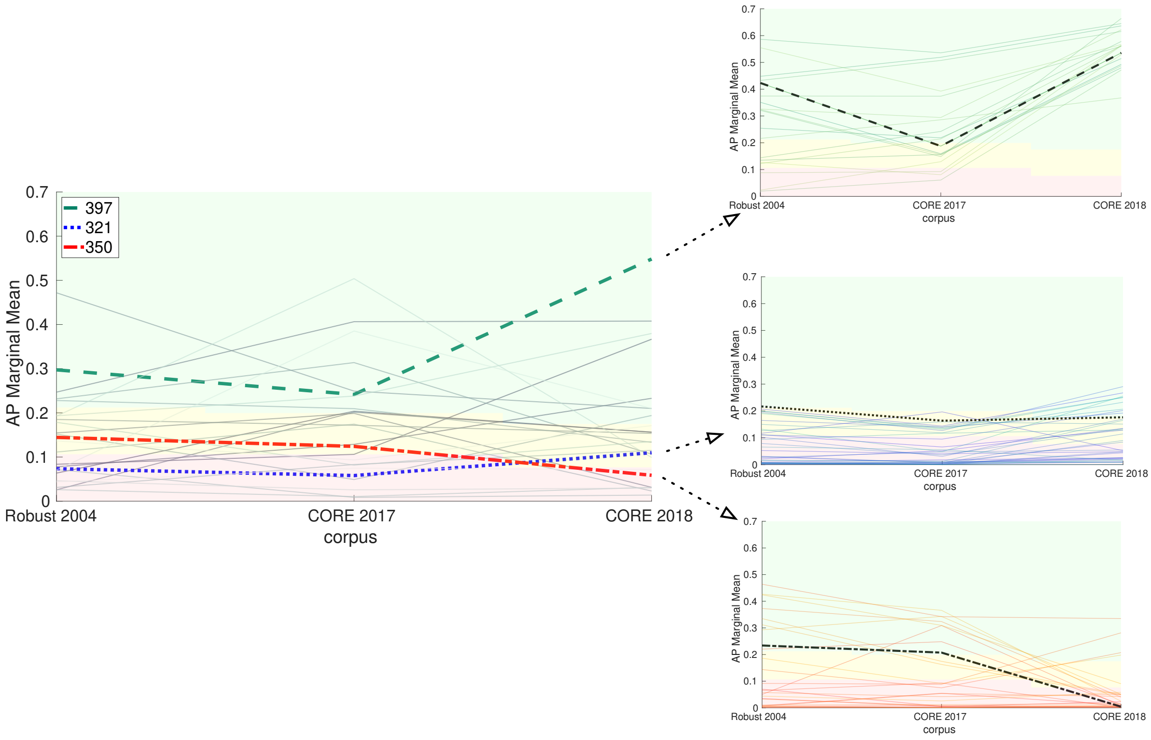


Fig. 4.8 Interaction effects between topics and corpora (on the left), where each line is a single topic. The Marginal AP is the average over all possible system configurations for either a topic considered as the combination of all its formulations (left), or a single topic formulation (right). Topic 321 in blue (‘`women in parliaments`’), is almost always difficult; topic 350 in red (‘`health and computer terminals`), almost always medium; and topic 397 in green (‘`automobile recalls`’), always easy. On the right, for each of these three topics, the interaction between query formulations and corpora are demonstrated. The black line was the original formulation corresponding to the TREC title query. The red, yellow, and green bands correspond to the hard, medium, and easy query performance ranges.

interaction factor and we can observe that topics can be easy, medium or hard depending on the corpus. We also highlight specific topics that exhibit consistent effectiveness trends across collections: Hard (topic 321 in blue), Medium (topic 350 in red), and Easy (topic 397 in green). On the right, we expand all formulations*corpus interactions for each of those highlighted topics, with the original TREC title query shown in black as a point of reference. We can see that regardless of whether a topic is classified as easy, medium, or hard, we can generally find at least one query reformulation for that topic in any of the three regions across the corpora.

4.7.2 Micro-Level ANOVA

Table 4.12 shows the results for model (MD5$_{mi}$) on the Robust 04, Core 17 and Core 18 corpora. All the factors are again statistically significant.

Table 4.12 Model (MD5$_{mi}$) on the Robust 04, Core 17, and Core 18 tracks using AP. The letter in the column "effect size" indicates whether the effect is large (L), medium (M) or small (S). For single factors, observe a similar overall behaviour to Model (MD4$_{mi}$). Overall, observe that the interactions including query formulations often have large effect sizes, indicating that using different formulations in combination with various components can induce dramatically different results. Furthermore, observe that several interactions of system components and the corpus have small or negligible effect sizes, indicating that the performance of these components are very similar in all of the corpora.

Source	SS	DF	MS	F	p-value	$\hat{\omega}^2_{\langle fact \rangle}$	effect size
Topic	1544.69	24	64.36	41853.88	<1e-6	0.8611	L
Formulations (Topic)	804.82	350	2.30	1495.32	<1e-6	0.7635	L
Stoplist	0.91	1	0.91	590.04	<1e-6	0.0036	(S)
Stemmer	0.02	1	0.02	15.46	<1e-6	0.0001	(S)
Model	19.64	8	2.45	1596.40	<1e-6	0.0730	M
Query Expansion	59.29	3	19.76	12851.05	<1e-6	0.1922	L
Corpus	33.04	2	16.52	10744.16	<1e-6	0.1171	M
Topic*Stoplist	1.14	24	0.05	30.83	<1e-6	0.0044	(S)
Topic*Stemmer	3.97	24	0.17	107.63	<1e-6	0.0156	S
Topic*Model	116.03	192	0.60	392.99	<1e-6	0.3172	L
Topic*Query Exp.	68.95	72	0.96	622.70	<1e-6	0.2165	L
Topic*Corpus	960.75	48	20.02	13015.90	<1e-6	0.7941	L
Form.*Stoplist	4.56	350	0.01	8.48	<1e-6	0.0159	S
Form.*Stemmer	18.23	350	0.05	33.86	<1e-6	0.0663	M
Form.*Model	84.64	2800	0.03	19.66	<1e-6	0.2438	L
Form.*Query Exp.	66.51	1050	0.06	41.19	<1e-6	0.2067	L
Form.*Corpus	527.09	700	0.75	489.66	<1e-6	0.6786	L
Corpus*Stoplist	0.05	2	0.03	17.41	<1e-6	0.0002	(S)
Corpus*Stemmer	0.16	2	0.08	52.31	<1e-6	0.0006	(S)
Corpus*Model	14.82	16	0.93	602.27	<1e-6	0.0561	S
Corpus*Query Exp.	11.55	6	1.93	1251.98	<1e-6	0.0443	S
Topic*Corpus*Stoplist	2.02	48	0.04	27.42	<1e-6	0.0078	(S)
Topic*Corpus*Stemmer	4.93	48	0.10	66.80	<1e-6	0.0191	S
Topic*Corpus*Model	76.70	384	0.20	129.88	<1e-6	0.2340	L
Topic*Corpus*Query Exp.	105.31	144	0.73	475.58	<1e-6	0.2967	L
Form.*Corpus*Stoplist	4.08	700	0.01	3.79	<1e-6	0.0119	S
Form.*Corpus*Stemmer	16.78	700	0.02	15.58	<1e-6	0.0593	S
Form.*Corpus*Model	104.42	5600	0.09	12.13	<1e-6	0.2778	L
Form.*Corpus*Query Exp.	65.57	2100	0.03	20.30	<1e-6	0.2001	L
Error	224.90	146250	0.00				
Total	4945.58	161999					

Table 4.12 shows the break-down by system factor for Table 4.11 by component contribution. As observed for the single-corpus case (see Table 4.7), the most important components are the IR model and query expansion, as well as their interaction with topics, while stop list and stemmers and their interaction with topics have small-size effects. However, while the single-corpus case showed that the topic*query expansion interaction is almost twice the size of the topic*model interaction, in the multiple-corpora case the opposite is true, and now the topic*model interaction are roughly 1.5 times the size of the topic*query expansion interaction. This could possibly due to IR models being a sort of "filter" with respect to corpora, whose impact change from corpus to corpus.

We can also observe, for the first time, the interaction between components and query formulations: the interaction with IR model and query expansion components have a large-size effect, almost the same size in this case, while the interaction with stemmers is now a medium-size effect, suggesting that the "clustering" induced by a stemmer can have an important impact on query reformulations. These trends are also confirmed by the third order interactions, i.e. topic * <component> * corpus and formulations * <component> * corpus.

Finally, when interpreting the component and corpora interaction, the most important components are, again, IR models and query expansion which have strong, large-size effects, with query expansion being roughly 1.3 times larger than IR models, while the interaction with stop lists and stemmers are negligible in size, indicating a very consistent behavior across corpora. This finding differs from that of Zampieri et al. [198] who found that the corpus*query expansion interaction had a negligible effect size. This also suggests that components of an IR system may indirectly contribute to topic difficulty as IR models and query expansion are clearly also sensitive to variations in topics and query formulations.

4.8 Topic Difficulty

Given these findings, we are finally in a position to revisit the fundamental tenet in IR that has been explored from many different angles in the past – the notion of *topic difficulty*.

Table 4.13 summarizes the outcomes of the topic ranking analysis using 10,000 permutations. For each permutation we considered both the *forward* and *backward* (or reverse) permutation. Thus, 20,000 total permutations were evaluated. We observed that in 2.12% of all permutations (slightly less forward than backward ones), it was possible to exactly match the permuted ranking of topics targeted[3], while on the other permutations we have a mean

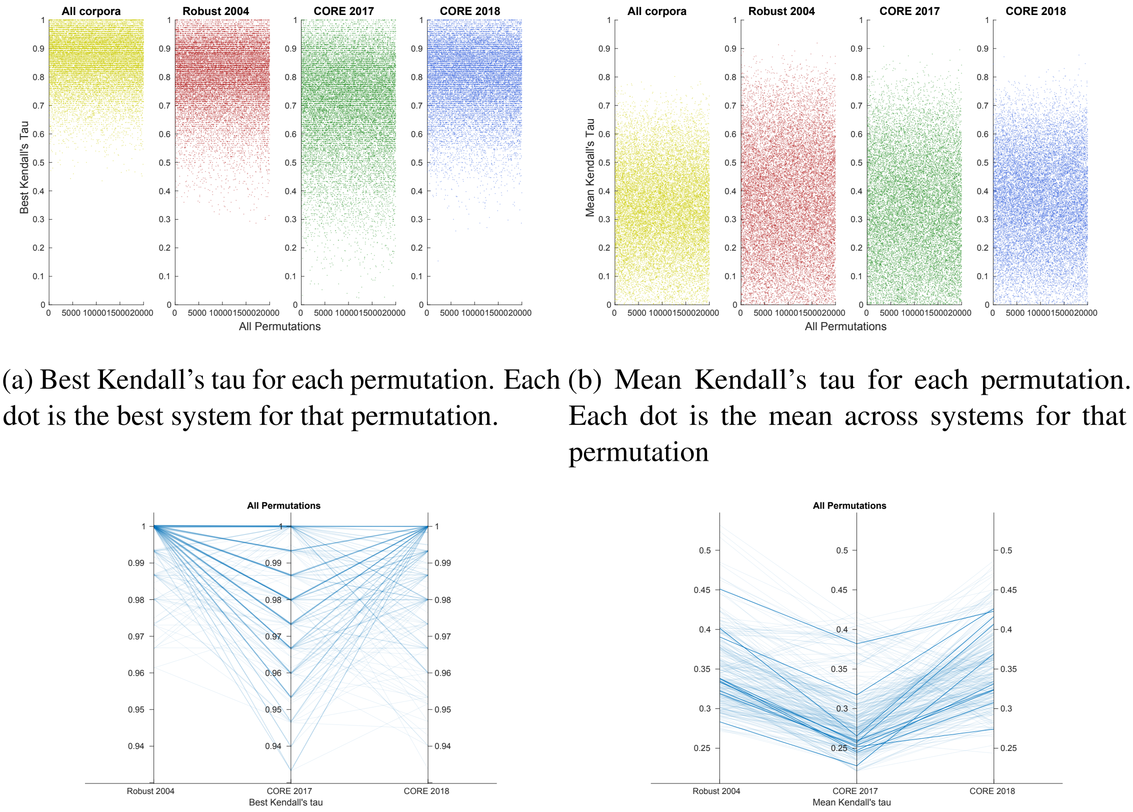
[3] The low rate of permutations for which it is possible to find an exact match provides further evidence that the corner case where a very poor query formulation for an "easy" topic still performs worse than a really good query formulation for a "difficult" topic is not a dominating factor, otherwise this value would be much higher.

Table 4.13 Ranking of topics analysis over 10,000 forward and backward permutations, 20,000 overall permutations. Observe that overall there is agreement between permutations of topics and rankings of formulations based on the relative performance. The corpus that is the easiest to target and find a given rank was Core 18, followed by Robust 04. The latter is an interesting outcome as several of the queries included in the Robust 04 collection were known to be "hard" topics in previous TREC tracks. This finding provides additional evidence to the importance of the magnitude of the effect size of query formulations across corpora. Conversely, the ratio of exact rankings is small, indicating that the formulation task is rarely effortless.

	Overall Statistics		
	Fwd Perms	**Bck Perms**	**All Perms**
Ratio of exact rankings	2.06	2.18	2.12
Best Kendall's tau	0.8588±0.0017	0.8579±0.0017	0.8583±0.0012
	Robust 04		
	Fwd Perms	**Bck Perms**	**All Perms**
Ratio of exact rankings	0.65	0.91	0.78
Best Kendall's tau	0.8053±0.0021	0.8030±0.0022	0.8041±0.0015
	Core 17		
	Fwd Perms	**Bck Perms**	**All Perms**
Ratio of exact rankings	0.55	0.43	0.49
Best Kendall's tau	0.7040±0.0032	0.7002±0.0032	0.7021±0.0023
	Core 18		
	Fwd Perms	**Bck Perms**	**All Perms**
Ratio of exact rankings	0.93	0.88	0.91
Best Kendall's tau	0.7977±0.0023	0.7958±0.0023	0.7967±0.0016

Kendall's tau of 0.85, indicating that the queries selected to induce the desired topic rankings were consistently close to the arbitrary target ordering. Given that these results represent a lower bound, they provide strong evidence that ordering topics by relative effectiveness is not intrinsically difficult, and in fact can be "arbitrarily" easy or difficult across many different corpora and system combinations.

Table 4.13 also shows what happens when we restrict ourselves to using only a single collection to find the requested ranking of topics, which is a somewhat harder case. All of the collections exhibited similar behavior in terms of exact match ratio – 0.78% for Robust 04, 0.49% for Core 17 and 0.91% for Core 18– indicating that it is more difficult to find an exact solution for a random topic ordering on a single collection. On Core 18 this ratio was slightly higher, suggesting that in this case it is easier to find the requested ranking. When comparing the Kendall's tau results, Robust 04 and Core 18 perform similarly while Core 17 was slightly worse, but in every case, there is a clear indication that it is possible to find topic orderings similar to the requested one, even on a single corpus.



(a) Best Kendall’s tau for each permutation. Each dot is the best system for that permutation.

(b) Mean Kendall’s tau for each permutation. Each dot is the mean across systems for that permutation

(c) Best Kendall’s tau for each system. Each line is a system, selecting its best Kendall’s tau across permutations.

(d) Mean Kendall’s tau for each system. Each line is a system, averaging its Kendall’s tau across permutations.

Fig. 4.9 Ranking of topics analysis across the different corpora.

Figure 4.9a is a visualization of the raw data which is summarized in Table 4.13. The Figure shows what happens when results are aggregated across all corpora as well as the outcome when each corpus is treated independently; the figure uses all available system configurations in our test set, i.e. we did not restrict our greedy algorithm only to the best configurations. Each permutation on the X-axis is plotted against the best Kendall’s tau. We can clearly see that, regardless of corpora used, the Kendall’s tau values tend to cluster above 0.6, suggesting that, for every permutation probed, it is quite possible to find at least one system which produces a similar ranking of topics being inspected. For the single corpora case, there is a higher likelihood of not finding a close mapping to the permutation being probed, but remains possible for many cases.

Figure 4.9b shows the mean Kendall’s tau across the systems for each permutation. A very similar behavior can be observed across the different corpora, with values tending to below 0.5, again with a larger spread when only a single corpus is targeted. This suggests that the average behavior of systems is very noisy and that it is much more difficult to obtain a requested ranking of topics from a whole set of systems.

In Figure 4.9c each line represents a system for which the best Kendall's tau across all the permutations was computed. We can see strong evidence once again that nearly *every* system can find a solution for at least one permutation on each corpus, albeit with high variance in a few cases. There appears to often be more than one system which is able to find at least one exact match for a permutation being probed across all of the corpora. We note that Core 17 appears to be a little more difficult than the other two collections in this respect, since several of our system configurations tended to have worse overall effectiveness in this case. In Figure 4.9d each line is a system where the mean Kendall's tau across all the permutations is compared. When viewed from this perspective, it appears to be more difficult for a system to consistently find a close match for every permutation and this behavior is quite consistent in this respect across corpora, when enforcing a fixed system configuration. Again some corpora are more difficult than others, and small clusters of our system configurations are substantially more effective than the others.

4.9 Topic Difficulty – Lessons Learned

We have presented evidence showing that topic difficulty is not an intrinsic property of an information need – meaning that query formulation based on a corpus and retrieval system, can be combined to sort topics arbitrarily based on a performance goal. While IR researchers have long been aware of the importance of query terms, the *magnitude* of the impact relative to other change to a system, such as the ranker, or even the introduction of query expansion has never been shown experimentally. ANOVA provides a powerful methodology to do it. While not discussed in detail in this work, the recent work of Liu et al. [101] show how similar query formulations are within a topic or when compared across different topics. It is remarkable how much performance can differ between two formulations of the same information need, with other factors being fixed.

This information can be used to further improve retrieval systems in IR as well as other related areas such as product, movie, or music recommendation. How can such a finding help researchers develop more effective retrieval systems? Firstly, it is worth noting that current evaluation paradigms usually consider a single formulation for each topic. All of the main evaluation campaigns, such as TREC or CLEF, allow participants to produce a single run for a given set of topics and queries. Such an arrangement prevents us to observe system behavior with small changes to each query. Automatic reformulation and query expansion (as shown in Tables 4.7 and 4.12) tend to have a large impact on the performance, one that is consistently greater than the retrieval model. If we consider carefully the conclusions reached in this work, we can see the potential value of investigating it further, in many different scenarios and applications, none of which are happening today.

We believe that multiple formulations can be extremely helpful in evaluating our IR systems, and we are hopeful future campaigns will incorporate them into their methodology. The cost of collecting new data is certainly a limiting factor in every decision, but note that there is a high economical interest here as query formulations for a single topic are all trying to find the *same* relevant documents, and there is often higher overlap in the documents retrieved. Query formulations can easily be collected through click-logs, produced automatically using ontologies or written directly by the assessors or crowdworkers. This additional information can be used to improve current evaluation methodology, with a small impact on the cost to develop a collection while providing substantial benefits. Using the ANOVA framework proposed here, practitioners can study formulations, systems, collections, and their associated interactions in exhaustive detail. More comprehensive failure analysis tools allow us to build more reliable systems, and to identify and eliminate tail cases which lead to poor performance under certain conditions.

To conclude, the lessons learned from our work have a number of potential applications and extensions. First, we have empirically observed the effect size of query formulations in IR evaluation, which is, in itself, an important issue warranting further attention. We show how query formulations interact with other components commonly found in a typical retrieval pipeline. These interactions were found often to be significant, and are rarely negligible in size. Their absence from current practices in IR evaluation should be reconsidered in the research community. If our goal is to model real performance of systems, collections creators should explore how to best include multiple formulations of each topic. Their use should also be standard practice for system builders as it is a valuable tool to perform a detailed performance analysis in complex retrieval software that is composed of multiple components – all of which can have unexpected interaction which can degrade (or improve) the overall retrieval performance. Our isolation of multiple formulations of topics has allowed us to study in detail the concept of "topic difficulty", which we now understand to be a construct of a specific retrieval configuration – the collection, the system and the query which represents the topic under consideration – and not a property intrinsic to the topic alone. The malleability of relative performance which can be induced when of multiple query variations of a topic are available enable "topic difficulty" to be fully controlled in a collection, and raises important questions in current several of the communities current evaluation practices and potentially in related fields, such as QPP and conversational IR. That is, the distribution of which topics perform well or poorly can be arbitrarily reordered using query formulations such that relative system performances change, as their performance may be better or worse depending on the specific query choice. Many open questions remain, Query formulation and its role in evaluation warrants further study in the IR community.

4.10 Efficiency and Scalability

While we have discussed several new avenues of future work in Section 5.6, we have not discussed one of the important challenges we encountered, which is the efficiency and scalability of current ANOVA modeling techniques. Query formulations and the wide-spread availability of publicly available retrieval systems such at Terrier allowed us to produce far more data than we could incorporate into our models. Given the nature of the IR experimental scenario, new factors introduced in the experiment are multiplicative with respect to the total number of results. For example, 9 rankers, 4 query expansion models, 4 stemmers, stopping (2) on 3 collections requires 864 retrieval runs, each of which are composed of hundreds (or even thousands) of queries to run (50 topics of 25 query formulations is 1,250 total queries): accruing for a total of 1,080,000 retrieval results. With current software tools available, running the ANOVA model on such a big dataset has a significant computational cost, and our current experiments we limited primarily by the RAM available, with our largest server having 1.5TB of RAM. We are aware of very few other studies using ANOVA in the IR community using data at this scale. But for IR researchers who are interested in designing efficient and scalable algorithms, a scalable and efficient ANOVA framework for CUDA and other GPU related hardware would be a valuable contribution to the community. Remarkable achievements are possible using GPU hardware in the Deep Learning community. However, most of these efforts are dedicated to building new NLP/ML models, and not in leveraging it to evaluate models we create. We should consider using this new hardware to improve current evaluation techniques too.

4.11 Final Remarks

This Chapter contains our analysis on the concept of topic difficulty via the lenses of ANOVA. The methodology is considered stable and widely adopted, but our study differentiates from previous similar endeavours due to the factors considered. We explored a multi-corpora scenario by including several collections, Robust 04, Core 17 and Core 18, that rely on the same topics. We furthermore introduced in this kind of analysis multiple formulations for the same information needs. The vast amounts of replicates for the same experimental conditions (topic and systems considered), introduced by the numerous corpora and formulations, allowed us to profoundly study the role of interaction between components in determining the IR systems' performance. In particular, we observed a significant interaction between the formulations and the systems, able almost to obfuscate the interaction between the topic and the system itself. Such findings made us question the concept of topic difficulty. In particular, we developed a methodology meant to determine whether the topic difficulty should be considered an intrinsic property of a topic or rather an artefact resulting from the combination of different factors. The results showed that we could not deem a topic to be intrinsically difficult: the complexity of a topic results from many interacting aspects, such as the collection used, the formulation used for the topic and the system at hand.

Chapter 5

Improving our Statistical Tools: a Case for Generalized Linear Models

5.1 Introduction

T-test and ANOVA, together with linear regression, belong to the family of statistical methods called GLiMs, a generalization of the multiple linear regression. To fully exploit GLiMs, data should satisfy some assumptions. The main three assumptions are: *i)* the independence of the observations; *ii)* constant variance of the data; i.e. *homoscedasticity iii)* normal distribution of the data, i.e., *normality*. A fourth often overlooked assumption is that *iv)* the expectation of the response should be correlated linearly with the experimental conditions, i.e., *linearity*. These assumptions allow for an analytical solution of the model and its practical computation. Thus, the more such assumptions are satisfied, the more precise is the model; violating them to a certain extent, makes the computation of the model more approximate.

Previous literature showed great interest in studying the empirical consequences of using data that violates the assumptions underlying the linear modelling, both from a theoretical standpoint [80, 150], but also considering empirical IR data [168, 79, 37]. Such works show that, in general, linear models are resilient to the violation of their assumptions.

Nevertheless, having better fitting models allows for obtaining more precise inference. Therefore, a great effort aimed to improve the models used. Such endeavour includes transforming the data to bring them closer to the linear assumptions, expanding models by introducing new factors or changing completely the modelling approach.

GLiMs are part of a broader set of techniques called Generalized Linear Models (GLMs). The GLMs framework represents a further generalization of the GLiMs, that relaxes some of the underlying assumptions to increase the fitness of the models. In particular, the data is no longer required to follow the normal distribution or have constant variance. GLMs relax the fourth assumption, allowing the relationship between the expectation of the response and the experimental conditions – called *link* – to have different forms, besides the linear one.

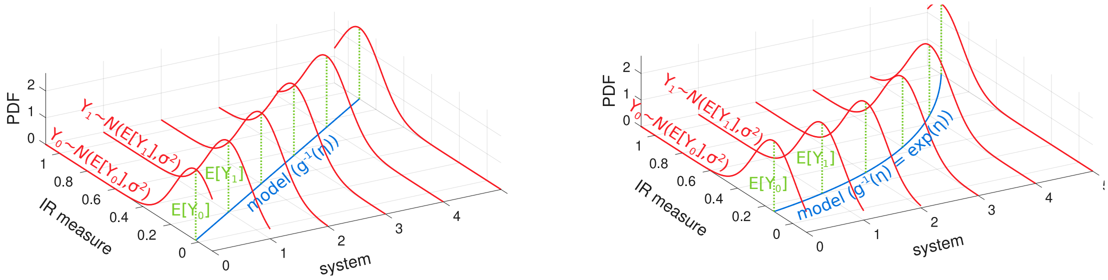
In this work, we begin the investigation of the GLMs framework applied to IR evaluation, showing how it can help to overcome the limitations mentioned above by relaxing the linear modelling assumptions.

More precisely, we focus on the link function used in the GLMs framework, investigating the impact of different links on the modelling of the IR performance.

In this sense, our contribution is multi-fold:

- We propose a new way of visualizing the data that highlights linear models' assumptions. We then illustrate the behaviour of the IR data using such visualization;
- We instantiate the GLMs framework in the case of IR experimental evaluation, illustrating how to apply them;
- We experimentally compare different links to determine the most suited to the IR scenario, providing a starting point for future investigation on GLMs applied to IR evaluation.

5.2 Generalized Linear Models



(a) The "traditional" linear model: an identity link and a Gaussian distribution of the response. Red lines represent the distribution of the explained variable Y, while the blue line, the model, tries to describe how $E[Y]$ (green lines) changes.

(b) $E[Y]$ is not directly proportional to the system, thus a different link should be used: instead of modeling $E[Y]$ we model $g(E[y])$, where g is the log function. Y distributes still normally, only $E[Y]$ has changed.

Fig. 5.1 Visual description of what changes when we change the link function in a GLM.

Parametric statistical tests, such as t-tests or ANOVA, rely on the assumption that data can be – and are – modelled using a linear model. Focusing on the IR scenario, we typically have a set of m systems to be applied to a set of n topics. Given a system s and a topic t, we can compute a measure that describes how well s performs on t. To align with previous works in the GLMs domain, we refer to such measure as y_{ts} and call it "response". The response y_{ts} is a realization of a random variable Y representing the score achieved by a system on a topic. The experimental conditions - the topic and the system used in our case - are somehow correlated with the response, and therefore we refer to them as "covariates". Using traditional linear models, the expectation of a random variable $E[Y]$, is modeled as a linear combination η of the covariates as follows:

$$E[Y] = \eta = \mu + \tau_1 t_1 + \ldots + \tau_n t_n + \alpha_1 s_1 + \ldots + \alpha_m s_m \tag{5.1}$$

Where t_i and s_j are dummy coding variables for the topic and systems considered, τ_i is the effect due to the i-th topic, α_j is the effect due to the j-th system. The intercept μ represents the grand mean of our data.

If we instantiate Y to a real observation y_{ts}, we must include the error ε_{ts}, the variability of the data that the model does not explain:

$$y_{ts} = \mu + \tau_1 t_1 + \ldots + \tau_n t_n + \alpha_1 s_1 + \ldots + \alpha_m s_m + \varepsilon_{ts}$$

Under this framework a t-test, used to determine if system i is better than system j, corresponds to verifying that the coefficient α_i is statistically significantly greater than α_j. Similarly, an ANOVA corresponds to verifying that at least one among the α coefficients is statistically significantly different from the others.

Note that, without losing generality, we can say that we model $g(E[Y]) = \eta$, where g is the identity function $g(x) = x$. In this sense, g is the function that *links* $E[Y]$ to η.

To compute the linear model and grant its statistical properties, we assume $Y \sim \mathcal{N}(\eta, \sigma^2)$ and thus $\varepsilon \sim \mathcal{N}(0, \sigma^2)$, which means that we assume that Y distributes normally, and it has the same variance everywhere.

Summing up, fitting a linear model requires modeling the following aspects:

1. a linear combination η of the different explanatory variables;

2. a *link* function g to connect $E[Y]$ to η;

3. a distribution for Y.

Under this framework, the traditional linear model is a particular case where, as aforementioned, g is the identity function and Y distributes following a Gaussian distribution with mean (i.e. expectation) η and constant variance σ^2. A visualization of an ideal linear model is depicted in Figure 5.1a. We assume to have a set of systems, each with a set of performance measurements. For each system, the distribution of the scores is depicted in red. All the distributions are normal and homoscedastic.

Following eq. 5.1, the focus of the linear modelling is the expectation of the explained variable, which is represented in Figure 5.1a by the green lines. Since all the expectations fall on a straight line, we can model the data using a traditional linear model, depicted in blue in the Figure.

Compared to a traditional linear model, a GLM relaxes items 2 and 3. First, it models $g(E[Y])$, where g, the *link*, can be any monotonic continuous function. Secondly, using GLMs, the response Y can follow a distribution $f(\theta)$ that is not necessarily Gaussian. This also relaxes the *homoscedasticity* requirement. In other words, the variance can change with the expected mean itself. Thus, a GLM can be expressed in the following form:

$$g(E[Y]) = \eta, \text{ with } Y \sim f(\theta)$$

The chosen probability distribution $f(\theta)$ must be a member of the exponential distributions family. A location parameter $theta$ characterizes distributions belonging to the exponential family - e.g., the mean of the normal distribution. If we observe that $g(E[Y]) = \theta$ for a given distribution of Y, then we say that g is the *canonical link* of such a distribution. The canonical link has some advantages related to the optimization and the speed of convergence of the model parameters. Nevertheless, which link function to use depends on the data and their characteristics, often relying on empirical observation.

Figure 5.1b shows a scenario where a traditional linear model is not suited anymore. By looking at our data, we observe that $E[Y]$ does not fall in a straight line. A simple linear model would not be expressive enough to describe the data's complexity fully. We therefore resort to use GLMs. Akin to Figure 5.1a, the performance distributes normally with equal variance for all the systems, but the expectation $E[Y]$ appears to follow an exponential line. Therefore, to bring it back to a linear space, we can transform $E[Y]$ using the log link. Thus, our model becomes $\log(E[Y]) = \eta$ or, equivalently $E[Y] = \exp(\eta)$.

Notice that, in Figure 5.1b data are not transformed: if so, also the Y axis - and the predictions - would have changed. The response remains on the same scale: what changes is the model's shape. Fitting a GLM is substantially different from transforming the response in a non-linear space. When we apply a non-linear transformation g to the response Y, we assume that, in the new space, $E[g(Y)]$ is linearly correlated with the covariates and $g(Y)$ follows a normal homoscedastic distribution. In this sense, $g(Y)$ should comply with the linear modeling assumptions. If we chose to use a GLM, we believe that the linear correlation is between the predictors and the transformation $g(E[(Y)])$. In fact, in general

$g(E[Y]) \neq E[g(Y)]$. Transforming the response also means that predictions - and errors - will be in the transformed space while, with GLM, predictions and errors remain in the original scale.

To carry out statistical inference and test whether a system is statistically better than another, we need two elements: *i)* the difference between the effects of the systems *ii)* and the *Standard Error (SE)* associated with their comparison. Both elements rely on the concept of *contrasts* [78]. A contrast is a linear combination of the coefficients of a linear model using a vector **c**, where $\mathbf{c} \in \mathbb{R}^{k \times 1}$, with k the number of possible coefficients and $\sum_j \mathbf{c}(j) = 0$. Contrasts allow to model comparisons between (groups of) factors. Each contrast corresponds to a specific hypothesis that we are interested in testing. For example, in IR we are usually interested in carrying out pairwise comparisons between systems. In such case, the contrast vector to compare i-th and j-th systems is:

$$\mathbf{c}_{ij}(h) = \begin{cases} 1, & \text{if } h = i \\ -1, & \text{if } h = j \\ 0, & \text{otherwise} \end{cases}$$

Then, called $\boldsymbol{\alpha}$ the systems coefficients vector, we can compute the pairwise difference between effects as $\Delta_{ij} = \mathbf{c}_{ij} \cdot \boldsymbol{\alpha}$. Using the procedure mentioned above, we can define all the pairwise contrasts and obtain all the differences between pairs of systems.

The SE for a pairwise contrast between coefficients α_i and α_j is computed as:

$$SE_{ij} = \hat{\sigma}^2(\alpha_i) + \hat{\sigma}^2(\alpha_j) - 2\hat{\rho}(\alpha_i, \alpha_j) \tag{5.2}$$

where $\hat{\sigma}^2(\alpha_i)$ is the variance associated with the coefficient α_i (which should not be confused with the sample variance of the scores observed for system i). Similarly, $\hat{\rho}(\alpha_i, \alpha_j)$ is the covariance between the coefficients α_i and α_j. These values can be obtained from the covariance matrix. The (asymptotic) covariance matrix for a GLM is the inverse of the negative of the matrix of the second derivative of the log-likelihood function.

Once we have the SE for each contrast, to test if the performance of systems i and j are different, we compute our test statistics as:

$$t_{ij} = \frac{\Delta_{ij}}{SE_{ij}}$$

t_{ij} can be compared to the proper critical value according to the distribution or can be used to obtain the p-value. This is a generalization of the traditional t statistics and can be used to carry out several inferential tests - t-tests, ANOVAs, F-tests. By comparing t_{ij} with the

Table 5.1 Link functions considered. Φ and Cauchy are the CDF of a Standard Normal and Cauchy Distribution respectively.

name	function	inverse
identity	$g(x) = x$	$g^{-1}(x) = x$
log	$g(x) = \log(x)$	$g^{-1}(x) = e^x$
exp	$g(x) = e^x$	$g^{-1}(x) = \log(x)$
tanh	$g(x) = \tanh(x)$	$g^{-1}(x) = \text{arctanh}(x)$
logit	$g(x) = \log\left(\frac{x}{1-x}\right)$	$g^{-1}(x) = \frac{1}{1+e^{-x}}$
probit	$g(x) = \Phi^{-1}(x)$	$g^{-1}(x) = \Phi(x)$
cauchit	$g(x) = \text{Cauchy}^{-1}(x)$	$g^{-}1(x) = \text{Cauchy}(x)$

proper value Q of the Studentized range distribution, we can carry out a Tukey's HSD [173] pairwise comparison. This allows us to correct for the multiple comparisons problem that arises due to the high numbers of comparisons typically carried out on IR collections.

5.2.1 Using GLM in IR scenarios

As pointed out in the previous section, to fit GLMs we need to select the link function and the response distribution. Any possible monotonic continuous function can be a suitable link. The choice of which link to use depends on the shape of the data. In our analyses, we resort to focusing on the most popular link functions, indicated in Table 5.1. For each link, we report its name, the function and its inverse. Notice that the inverse of the link describes how the expectation of the response changes. For example, akin to Figure 5.1b, the log link suits scenarios where $E[Y]$ appears to follow an exponential pattern. We include the log, exponential and hyperbolic tangent (tanh) functions in our experiments. We also experiment with a series of sigmoidal functions: logit, probit and cauchit. Such functions have similar shapes with different steepness and are typically used for data that can be interpreted as probabilities. As observed in previous works [37, 130, 129, 21], the logit transformation renders normal the score distribution with the drawback of rendering unusable observations for which AP is zero or one. GLMs based on the logit link, on the other hand, by transforming the expectation of the response rather than the response itself, avoid such corner cases. However, when the expectation of a system's performance is close to zero, using log-based links – e.g., log and logit – determines the high variance of the coefficients associated with such a system. Following eq. 5.2, the large variance increases standard errors, and this weakens the inference, causing us to detect less significantly different pairs. In the literature, it is therefore suggested to remove outliers with close-to-zero expected performance.

Concerning the distribution, as aforementioned, we are limited to distributions drawn from the exponential family. Among the most typical representatives of such distributions family, we can list Gaussian, Bernoulli and Binomial, Poisson, Gamma, and Inverse Gaussian. Except for the Gaussian, which has $\mathbb{R}$ as domain, all the distributions have a different domain compared to the traditional IR measures. The Binomial is defined over the natural numbers, up to a given threshold - the Bernoulli is a particular case, where the threshold is 1. The Poisson distribution is defined over $\mathbb{N}$. Finally, both the Gamma and Inverse Gaussian are defined on $\mathbb{R}^+$, excluding therefore the 0, which is a possible value for most of the IR measures. By adequately changing the IR measure, it might be possible to use different distributions besides the Gaussian. Nevertheless, as observed by [80, 150, 168, 31], most of the tests are typically resilient to the violation of the normality assumption. Furthermore, we are primarily interested in investigating the impact of the links alone. Because of that, in this work, we maintain the distribution (Gaussian) fixed, leaving the study of different distributions to future work and focusing only on the benefit of changing the link function.

Following Tague-Sutcliffe and Blustein [168] and Banks et al. [12], to study the GLMs we use a series of models like the one in eq. 5.1. All of them are based on different links, but include only two factors: system and topic. We leave multi-factor analyses as future work.

5.3 Experimental Approach

5.3.1 Measuring Models' Deviance

Traditional linear models are commonly fit using the OLS approach that minimizes the Sum of Squares of Residuals (RSS). Different models can thus be compared by comparing their RSS. This is not the typical scenario with GLMs where OLS method cannot be used, but the model-fitting is obtained via maximum likelihood. Therefore, comparing the RSS is not advisable. The most common goodness-of-fit statistics under the GLM framework is the deviance, which is analogous to the RSS under the linear framework. Deviance is defined as twice the difference between the log-likelihood of the saturated model – a model with a parameter for each observation – and the fitted one. Similarly to the RSS, a model is considered to fit better the data if its deviance is low. Therefore, as the first analysis, we measure the goodness of fit achieved by different links using the deviance.

5.3.2 Comparing the Number of Significant Differences Found

The first aspect that we investigate is whether different links have a power advantage over traditional linear modelling. A model is said to have more "power" than another if it better discriminates which systems differ, even with little difference between them. This means that a more powerful statistical test identifies more SSD systems pairs. From a practitioner perspective, being able to identify more SSD pairs is essential: correctly individuate which system performs better allows investing in more promising solutions that might have been discarded if a weak test was used. Nevertheless, an overpowered test that considers every difference between systems statistically significant is useless due to its low informativeness. Similarly to many other scenarios, we have a trade-off between *type I errors* (false positives) and *type II errors* (false negatives) associated with the identification of pairs of SSD systems.

5.3.3 Simulation

One aspect that impairs evaluating new evaluation approaches is the absence of the ground truth: we do not know ahead what systems are significantly different. We, therefore, propose to evaluate the different links on both actual data and a simulated data set. The construction of the synthetic data-set is roughly based on the strategy proposed by Robertson and Kanoulas [130]. In particular, such an approach was used initially to build statistically identical replicates of each system-topic experiment, to measure their interaction. The simulation approach, originally dubbed "BST (Bootstrap) simulation", requires taking the runs of a system as seed. For each topic, the scores for the relevant documents are sampled with replacement among scores associated with relevant documents by the seed system. Similarly, for the non-relevant documents, the scores are sampled with replacement among scores assigned by the seed system to non-relevant documents. From a theoretical standpoint, even though single synthetic runs might be slightly different, they are *expected* to yield statistically non-different performance. If we want to build a ground truth and construct statistically better (or worse) systems, besides sampling the scores for the relevant documents, we also need to scale them. This results in a system that, by construction, assigns higher scores to relevant documents and, therefore, is better in expectation. A similar approach to build statistically different runs was recently defined by Parapar et al. [122]. The difference between [130] (and thus the approach employed here) and [122] concerns the distribution used to sample the scores. Robertson and Kanoulas [130] use the empirical score distribution and sample it with replacement, while Parapar et al. [122] sample from two log-normal distributions fitted on the scores associated with either relevant or not-relevant documents. We are aware of the shortcomings associated with the simulation approach above-mentioned. In particular, it does not allow to properly model other factors that might influence the performance of a system, e.g., the interaction with the topic. Nevertheless, the simulation approach can provide a valuable intuition of the differences between the links.

5.4 Experimental Analysis

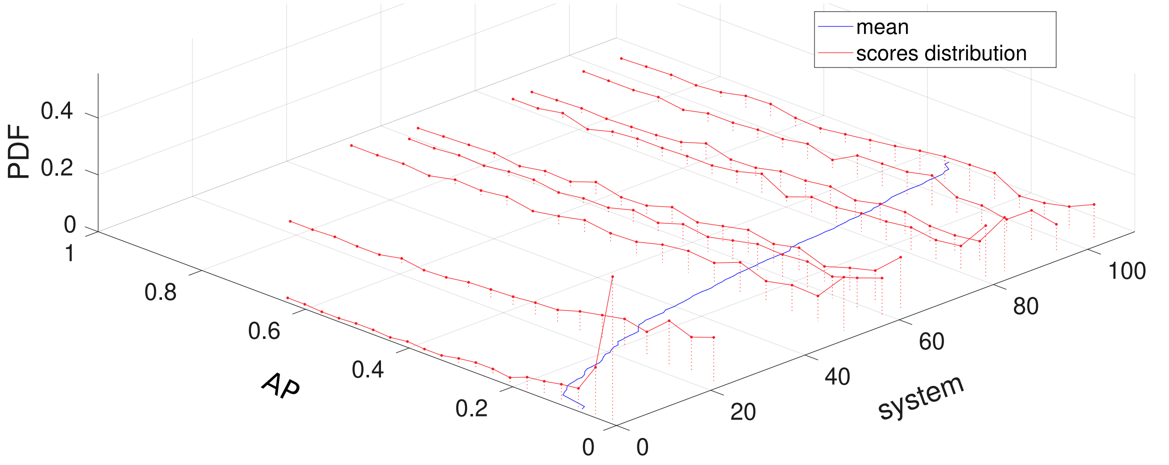
5.4.1 Experimental Setup

To evaluate the behaviour of the GLMs in the IR scenario and compare different links, we consider 2 traditional TREC collections for *ad-hoc* retrieval: TREC 13 Robust 04 [180] and TREC 27 Core 18 [3] collections. Robust 04 relies on disks 4 and 5 of the TIPSTER corpus minus the Congressual Records, has 249 topics and received 110 submissions (5995 pairwise comparisons). Core 18 collection relies on Washington Post document collection, has 50 topics and 72 runs submitted (2556 pairwise comparisons). We also include a second version of Core 18 collection, where we remove eight outlier runs, in terms of empirical MAP. Following Laurikkala et al. [95], we define "outliers" those runs having MAP 1.5 times the inter-quartile range lower than the lowest quartile. We dub the reduced version of Core 18 *without outliers* "Core 18-wo". It has 64 runs that lead to 2016 pairs of systems. All the collections have ternary relevance judgements with possible values {0, 1, 2}, that indicate respectively, not relevant, partially relevant and highly relevant documents. As performance measures, we use Average Precision (AP), precision with cutoff 10 (P@10), Recall (R), Normalized Discounted Cumulated Gain (nDCG) [81], and RBP with persistence of 0.8 [115]. In section 5.4.5, we repeat the topic sampling 1000 times.

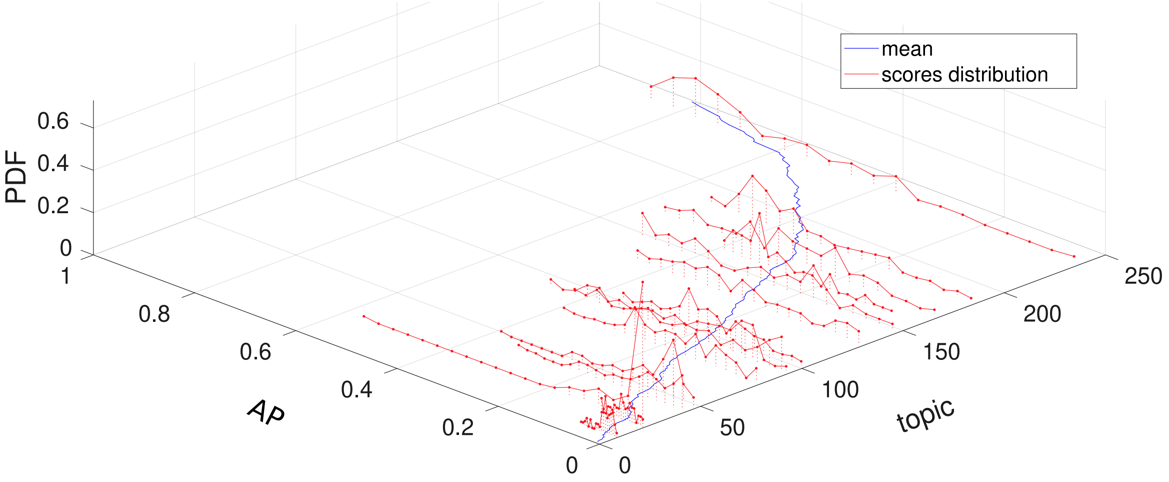
5.4.2 Fitting Linearly IR Data

Figure 5.2 illustrates what happens when we plot the empirical data obtained using Robust 04 collection. In Figure 5.2a we plot the MAP for each system (blue line). Similarly, in Figure 5.2b we plot the average AP performance over all the systems for each topic. We plot using a red line the distribution of the scores for a subset of systems and topics for visualization. By looking at Figure 5.2, it is interesting to notice that the blue line, which represents the average performance, does not follow a straight line. The identity link is not expressive enough to describe the complexity underlying the IR data. Similarly, the distributions of the observations (red lines) are far from being normal. A GLM, thanks to its additional descriptive power might better fit our data[1]. Figure 5.3 shows that a very similar behavior is observable also on the systems of Core 18 collection. Additionally, notice that the eight lowest-performing systems of Core 18, all submitted by the same group, have a completely different distribution compared to the others and this qualifies them as outliers.

[1] For the sake of completeness, the actual linear model fitted on the IR data is more complex. There is a dimension for each system – and topic – and not a "blue line", but a hyper-plane. Nevertheless, this representation gives the idea of how far we are from the ideal scenario to apply a linear model.



(a) Robust 04- Systems



(b) Robust 04- Topics

Fig. 5.2 Figures above show that, for the Robust 04 collection *i)* the distributions of the AP scores (red lines) are not normal for a given system/topic (and they hardly have the same variance) *ii)*, the mean is not linear.

5.4.3 Goodness-of-fit

Table 5.2 illustrates the deviance measured for different GLMs using several link functions, IR performance measures, and experimental collections. Interestingly, the traditional approach based on the identity link, that corresponds to the current evaluation methodology, presents a low goodness-of-fit, given its high deviance compared to other links. The exponential link is the worst, systematically underperforming on all experimental conditions. This suggests a convergence problem: such a link is not capable of correctly modelling the IR data. To explain this phenomenon, we remark that with the exponential link and performance values below 1 – always, given the IR measures considered – small changes in performance

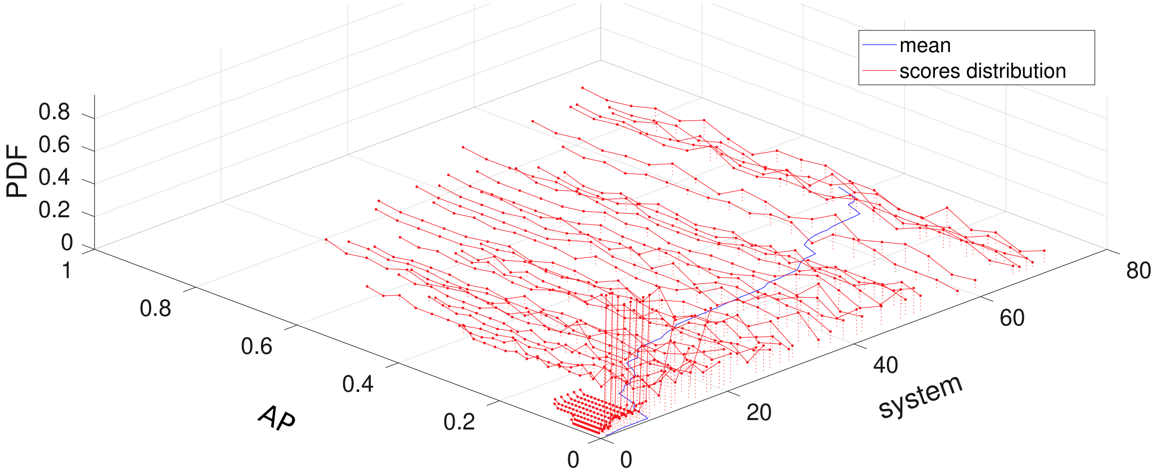


Fig. 5.3 Systems performance distribution for the Core 18 collection. Observe the non-linear distribution of the means, as for the Robust 04. The first 8 runs have different distributions compared to the others: they are outliers.

Table 5.2 Seviance observed for different models on several different scenarios. In bold minimum value observed for each experimental setting. The color indicates optimal (green), average (white) or low (red) results.

	robust04					core18					core18-wo				
link	AP	P@10	Recall	nDCG	RBP	AP	P@10	Recall	nDCG	RBP	AP	P@10	Recall	nDCG	RBP
identity	356.79	901.51	622.67	467.98	381.62	52.48	156.32	99.05	69.21	75.90	44.93	135.61	74.46	59.89	63.23
log	334.06	886.42	646.34	471.87	368.30	40.90	**132.00**	91.57	61.73	**59.00**	40.76	**128.12**	75.36	60.18	**57.76**
exp	387.52	955.56	719.93	505.65	400.98	61.31	219.01	159.33	90.92	98.80	49.19	160.79	79.90	64.29	72.55
tanh	348.91	894.19	640.07	467.54	376.76	50.30	147.13	95.89	66.00	71.15	43.81	132.13	75.84	59.80	61.00
logit	**329.46**	**882.14**	593.82	**458.04**	**366.89**	**40.50**	132.77	**85.47**	**60.51**	59.44	**40.36**	128.52	72.21	**58.93**	58.12
probit	330.36	882.66	**590.87**	458.05	367.65	40.71	133.26	85.52	60.63	59.81	40.57	128.84	**72.13**	58.99	58.42
cauchit	332.49	884.67	627.34	463.81	367.40	40.87	132.24	86.80	60.63	59.03	40.73	128.42	74.05	59.05	57.81

need to be counterbalanced by noticeable changes in parameters' magnitude. This leads to overall instability, especially concerning shallow performing systems. To further support this thesis, notice that if we consider the `core18-wo`, the degradation in terms of goodness-of-fit due to the exponential link is lower compared to the `core18`. In general, the log link shows improved goodness-of-fit compared to identity one. It exhibits high goodness-of-fit, especially for the core18 collection (both with and without outliers) when combined with precision-oriented measures – P@10 and RBP –, where it appears to be the most suited model, given its minimum deviance. Tanh link exhibits an intermediate behaviour in all scenarios: it appears slightly better than the identity without providing relevant improvements. Logit link has the best goodness-of-fit in the majority of the cases, being in most cases the link with the lowest deviance. Logit, probit and cauchit links tend to perform similarly. This behaviour is somehow expected: their shape is overall very similar. The choice of which link to consider often relies on prior knowledge of the process that generated the data.

Table 5.3 number of statistically significantly different pairs of systems found using different GLMs. The first line represent the traditional scenario: a standard linear model ANOVA with Tukey's HSD correction. The color indicates if the new modelling allows to identify more or less pairs of SSD systems – green and red respectively – compared to the baseline. The shade indicates whether the change is big or small compared to the overall distribution. In gray we indicate outliers compared to the distribution of results.

	robust 04 - (5995 systems pairs)					core 18 - (2556 systems pairs)					core 18-wo - (2016 systems pairs)				
link	AP	P@10	Recall	nDCG	RBP	AP	P@10	Recall	nDCG	RBP	AP	P@10	Recall	nDCG	RBP
identity	3427	2347	3848	3704	2435	1210	1054	1115	1270	1104	789	596	427	786	648
log	3556	2383	3622	3550	2412	925	934	672	1097	948	878	635	384	748	659
exp	5527	5547	5841	5769	5556	2149	2259	2316	2281	2264	1675	1777	1590	1797	1789
tanh	3509	2354	3639	3641	2418	1301	1130	1086	1283	1161	843	633	380	766	671
logit	3700	2557	4018	3773	2654	976	1034	1267	1251	1086	926	713	594	818	762
probit	3693	2541	4027	3766	2644	974	1079	1304	1340	1171	926	710	597	815	754
cauchit	3682	2552	3929	3764	2668	848	739	877	872	777	796	713	597	823	756

5.4.4 Empirical Links Comparison

Following Subsection 5.3.2, we are first interested in observing the absolute power of the different links. Table 5.3 contains the number of Statistically Significantly Different (SSD) pairs identified by the GLMs based on different links, using Tukey's HSD [173] test on distinct collections and with different measures. The identity link corresponds to the current evaluation approach. The exponential link is likely to cause model overfitting on all collections and measures. It identifies an extremely high number of SSD pairs, which is an outlier, according to the distribution of the number of SSD pairs found by other links. Considering Robust 04 collection, all the links outperform the traditional modeling strategy – i.e., identity link – on SSD pairs identified using AP and P@10 as performance measure. Ignoring the exponential, on both AP and P@10, the best-performing link is the logit one: it identifies 7.9% and 8.9% more SSD pairs compared to the identity on AP and P@10 respectively. On the other hand, considering recall, nDCG and RBP, we observe that both log and tanh fail to identify more pairs than the identity, while logit, probit and cauchit obtain an increase in the number of SSD pairs. nDCG tends to be the measure that benefits the least from the new links, with the logit link providing only 1.9% more pairs. The fact that logit, probit, and cauchit obtain good results suggests that their sigmoidal shape is well suited to model IR data.

Concerning Core 18 collection, Table 5.3 confirms what we pointed out in Subsection 5.2.1 about log-based links: if used in presence of outliers, they tend to underperform compared to the identity link. In particular, we observe that log, logit and cauchit almost always fail to beat the baseline. Upon further inspection, it is interesting to notice that almost all the pairs lost are associated with the eight outlier runs[2] submitted to the TREC track by a single group. As shown also on Figure 5.3, these runs have extremely low mean performance: their MAP ranges between 0.003 and 0.007. When compared to the distribution of the performance, they are outliers according to the definition proposed in [95]. As pointed out in Subsection 5.2.1, we know that the lower the expectation of the response, the higher is the variance in the coefficients, and thus the standard errors, causing instability in the comparisons. Concerning the recall, both logit and probit yield more SSD pairs than the identity. The mean recall for the eight outliers ranges between 0.01 and 0.02, one order of magnitude greater than the previous case. These runs are still outliers compared to the rest of the distribution, but they are not "pathologically" close to zero and thus are treatable by logit and probit links. Finally, it is interesting to notice that the probit, even though similar to logit and cauchit, outperforms the identity on all measures except AP. This might be correlated to the shape of the functions. Cauchit function is the steepest, and thus the most vulnerable to outliers. Logit function has intermediate steepness, exhibiting medium vulnerability to outliers. Finally, probit is the least steep and the more resilient to outliers.

As a final analysis, we consider Core 18-wo collection, where we removed the eight outlier runs. Table 5.3 shows that on the reduced collection all the new links obtain a consistent improvement over the baseline for what concerns AP, P@10, and RBP. Logit and probit links are the best, gaining 17.4% new pairs on the AP. Logit, probit and cauchit perform well also on recall and nDCG. Compared with the baseline, akin to Robust 04 collection, both tanh and log links lose several pairs in terms of recall and nDCG. Akin to Robust 04, on Core 18-wo nDCG is the measure that benefits the least of the new links, gaining only 5% of pairs at most. This lower increase in SSD pairs found for nDCG is likely due to the distribution of the nDCG scores. From an empirical standpoint, the plots for the nDCG like those of Figure 5.2, omitted for space reasons, tend to be closer to the assumptions of the linear model compared to other measures. The lower increase in the number of SSD pairs thus highlights two insights: i) nDCG is, as a matter of fact, one of the most stable measures violating less the assumptions underlying linear models; ii) other measures, by departing more from the assumptions, produce worse comparisons and GLMs mitigate this phenomenon.

Given the high number of possible setups, subsequent analyses will focus on the AP, being traditionally the most popular evaluation measure. Furthermore, given the similarity between log and tanh links, we report the results only for the first. Similarly, among logit, probit and cauchit links, we further investigate only the logit, given their similar behaviour.

[2] 8 runs appear in 540 pairwise comparisons

We maintain the exp link in the analysis to show the behaviour of “pathologically” overfitted models.

5.4.5 Link Stability

To investigate the stability of the links, we adopt the agreement indicators described in section 3.6.1, to measure the agreement of a test used on a pair of non-overlapping equal-sized topic sets. We also experiment with various dimensions of the topic sets. More in detail, the topic sets dimensions considered for Robust 04 are {125, 50, 25, 10}. For Core 18-wo collection, since it has 50 topics, we consider only topic-sets of size {25, 10}. For each size, we re-sample the pairs of topic-sets 1000 times. The patterns are overall similar both in terms of sample sizes and collections. As expected, the number of significant decisions – i.e., AA and MA – decreases with the dimension of the topic set, since less evidence is available. Conversely, measures of uncertainty – i.e., PA and PD – and instability – i.e., MD and AD – increase with the decrease of topics-set size. In most of the cases, the links with the highest AA is the exponential one, followed by the logit. The only exception to this is Core 18-wo with topic-sets of size 10, where the identity link has higher AA in comparison to the logit. Compared to the identity link, both the logit and the log links have more MA: this indicates that they decrease the type II error – i.e., falsely accepted null hypotheses. They need less evidence to consider two systems statistically different. Passive decisions (PA and PD) tend to be more for the identity in most cases. The identity link is the least powerful, this means that it is more conservative, preferring to avoid making decisions, to avoid false positives, but likely obtaining more false negatives. Concerning MDs, which are a mild sign of instability, they are low for identity, log and the logit links, while being much higher for the exponential link. Observe that AD are mostly absent, except in the case of the exp link: this indicates that, even though such a link has increased power, it also incurs a substantial intrinsic instability. The similar behaviour between identity, log and logit links indicates comparable quality in terms of stability. Conversely, the exp link has a very different behaviour that suggests its instability compared to the other links.

5.4.6 Comparing Decisions Taken

After observing the larger power and comparable stability exhibited by several new links, we now are interested in investigating which SSD pairs of systems each link finds. For Robust 04 collection, we compute the stability indicators among decisions taken, as defined in section 3.6.1, using two different links instead of two different topic sets. Figures 5.4a and 5.4b report the comparison between the identity link and the log and logit ones respectively. The

Table 5.4 Mean agreement over two topic sets for different links.

		125				50				25				10			
collection		identity	log	exp	logit	identity	log	exp	logit	identity	log	exp	logit	identity	log	exp	logit
robust 04	AA	2332.45	2491.36	4876.03	2616.35	1229.23	1415.57	4265.20	1542.27	489.06	533.35	3657.01	705.29	62.22	12.49	2703.10	65.08
	MA	633.34	675.22	500.40	703.57	665.41	766.20	749.10	797.01	616.21	787.56	988.01	836.24	280.01	214.32	1333.31	454.29
	PA	2593.33	2345.11	171.38	2215.39	3399.18	3027.13	256.17	2907.45	3909.39	3594.09	343.34	3415.37	4236.23	4246.35	514.09	3993.31
	PD	435.20	479.27	122.35	455.52	700.32	780.13	200.22	741.49	979.24	1073.27	280.09	1028.08	1415.34	1517.12	432.37	1470.24
	MD	1.07	4.44	248.05	5.36	1.26	6.38	395.54	7.17	2.29	7.53	541.51	10.42	2.39	5.12	759.14	13.28
	AD	0.00	0.00	78.39	0.00	0.00	0.00	130.37	0.00	0.00	0.00	186.24	0.00	0.00	0.00	254.20	0.00
core18-wo	AA	—	—	—	—	—	—	—	—	300.12	247.56	1335.06	318.50	66.04	11.57	1044.23	43.23
	MA	—	—	—	—	—	—	—	—	268.29	362.37	258.49	410.22	163.35	116.39	379.07	207.22
	PA	—	—	—	—	—	—	—	—	1161.31	1138.24	152.05	1013.13	1386.25	1481.43	197.39	1360.08
	PD	—	—	—	—	—	—	—	—	286.04	266.10	109.17	271.49	398.21	404.59	157.27	401.16
	MD	—	—	—	—	—	—	—	—	1.04	2.12	128.09	3.05	2.15	2.03	184	5.51
	AD	—	—	—	—	—	—	—	—	0.00	0.00	34.34	0.00	0.00	0.00	55.24	0.00

axes represent all the possible 110 systems of the Robust 04 collection, with the systems sorted by MAP. We compare each system against all the systems that performed, on average, worse, therefore, only the upper triangular area of the matrix is colored (the lower part is symmetrical). The cell's color describes whether two tests using GLMs based on different links agree or not in considering the pairs of systems statistically different. Note that we do not observe such extreme scenarios as MD or AD in both figures. The majority of the observations in both cases are either Passive decisions (PD and PA) or AA.

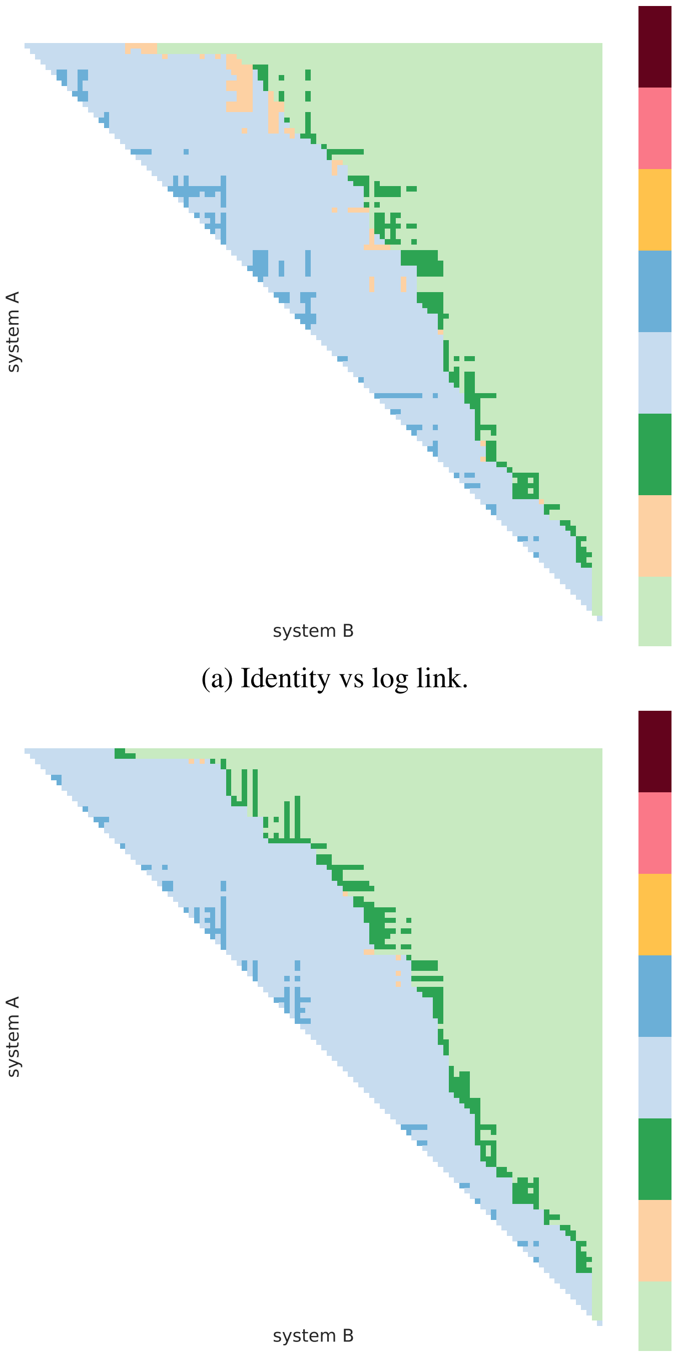
Looking at Figure 5.4a, we notice that the identity link identifies several significant pairs in the upper part of the system ranking – i.e., M1A, light orange squares. For example, five systems are deemed SSD compared to the best-performing one only by the identity link. On the other hand, the log link gains several significant decisions for average and low performing systems – i.e., M2A, dark green squares. Indeed, thanks to the shape of the log function, the log link identifies statistical differences among low-performing systems more easily. This behaviour of the log link, shared with the GMAP and the logit transformation, might be helpful to identify which system performs better on a set of particularly hard or *tail* queries.

Figure 5.4b shows the comparison between the identity and the logit link. In general, the logit link identifies more pairs of SSD systems for almost all systems tiers, regardless of their quality (M2A). Additionally, while the top tier – i.e., systems are not statistically different from the best – has dimension 19 for the identity link, the logit link removes two systems from the top tier, better discriminating between top-performing systems. It is interesting to notice that the logit link loses eight SSD pairs compared to the identity link (M1A). Out of those eight pairs, six were deemed not significant also by the log link: a piece of further evidence that they might be false positives wrongfully identified as SSD by the identity link.

5.4.7 Simulating the Ground Truth

The following analysis is based on the simulation strategy described in Subsection 5.3.3. We select, randomly, the run titled `uogRobSWR10` and use it as seed to simulate AP data. Different seed runs provide similar results. We apply the "BST simulation" on such a run, to build 70 statistically equally performing runs, 10 runs having a 1% increase in performance and 10 with a 1% decrease. We also include runs with a 2% and 3% increase and decrease by sampling 5 runs each (20 runs total)[3]. This leads to 110 synthetic systems, with 249 topics each – mimicking Robust 04 collection itself. By construction, 3450 pairs of runs are statistically different while 2545 are not. Table 5.5 reports the confusion matrices computed

[3] The number of simulations in each group has been chosen to obtain comparable values of true positives and negatives, while the siz of the increase has been kept low to avoid making it too easy to distinguish between pairs of different systems



(a) Identity vs log link.

(b) Identity vs logit link.

Fig. 5.4 Comparison between decisions taken by different links. Each square represents a pair of systems. System A (y axis) has a higher MAP than System B (x axis). The color of the square describes whether two tests using either the identity or the log/logit links agree on the relation between systems A and B. M1* and M2* indicate a significant decision taken only by the identity link and log or the logit links respectively.

Table 5.5 Confusion matrices obtained using the simulated ground truth.

		oracle	
	observed	A>B	A∼B
identity	ssd	842	0
	not ssd	2608	2545
log	ssd	779	0
	not ssd	2671	2545
exp	ssd	3142	1380
	not ssd	308	1165
logit	ssd	1127	0
	not ssd	2323	2545

between the decisions on systems pairs taken by different links and ground truth. The upper-right corner represents Type-I errors – the risk of falsely rejecting a null hypothesis. The lower-left corner contains the number of Type II errors. All the links beside the exponential one avoid Type I errors (False Positives). This further highlights the weaknesses associated with the exp link, which should be avoided. The absence of false positives is likely due to *i*) the fact that we have used the strict Tukey's HSD test; *ii*) the simplicity of the simulation – by ignoring the interaction between systems and topics, we reduce the variance and make it easy to recognize equally performing systems. Secondly, we can observe that the logit approach is the one that allows obtaining the lowest number of Type II errors (False Negatives), overcoming the identity link by 285 pairs.

5.4.8 Discussion

The log link performs well on low performing systems, in terms of SSD pairs identified, and it also exhibits similar stability when varying the topic-set. Therefore, we suggest adopting it to discriminate low-performing systems on tail queries. Concerning the logit link, as shown by Table 5.3, it exhibits increased power compared to the traditional evaluation based on the identity link and has stability comparable to other links, when dealing with different topic sets. Moreover, Figure 5.4b shows us that the logit link can identify more significant pairs over the entire spectrum of systems performance, including among top tier systems comparisons. This improved behaviour is supported also by the simulated data. However, to apply the logit link some precautions are needed. In particular, it is necessary to inspect the data and remove possible outliers that might impair the overall quality of the evaluation.

Nevertheless, the overall increased power and stability suggest to switch from the traditional linear models based on the identity link to GLMs relying on the logit link.

5.5 Final Remarks

To conclude Part I of this manuscript, we investigate in this chapter how to use GLMs to enhance the goodness of fit of our statistical tools with two objectives: i) build better models to describe (and predict) IR systems' performance; ii) improve how we evaluate IR systems. In line with previous works, we observed that traditional linear models, including ANOVA, fail to model the IR data properly. We showed how by using GLMs we are capable of overcoming such weakness and reaching far more satisfactory results in terms of statistically significantly different pairs of systems identified by our statistical models.

Part II

Performance Prediction

We are now interested in exploring predictive models applied to IR evaluation. More in detail, we start from a very well-developed research path, Query Performance Prediction (QPP), expand and analyze it based on the findings and tools described in the previous chapter.

Our analysis in the QPP domain begins with an investigation of a novel predictive model based on the concept of *Gain* – as defined by Umemoto et al. [174] – applied it to the systematic reviews task domain. In particular, we address the task of predicting and selecting which query formulation among those representing an information need will perform the best in terms of recall.

Secondly, we investigate if the query can provide insights on which models will perform better. In this sense, we shift the focus from predicting the performance of a system – to forecasting which system will perform better based on the query, getting closer to the automatic model selection domain. Given the complexity of the task itself, we address it by considering it at a coarser grain. In particular, we are interested in determining if lexical or semantic systems will perform better for a specific query. As before, we also embed in our analysis multiple formulations for the same information need.

While addressing the previous research paths we noticed major challenges linked to the complexity of evaluating the systems. Currently, the most common evaluation strategy for QPP is based on measuring the correlation between QPP predictions and observed IR performance. This approach allows for obtaining a single point estimation that – should – fully describe the performance of a QPP model. We argue that this solution does not provide sufficiently detailed insights to discriminate between predictive models being not expressive enough to model the performance of a QPP model. We develop a new measure, dubbed sMARE, capable of turning the QPP performance into distributions over the topics. This new measure allows us to attain better results in terms of discriminative power between QPP models.

To summarize, we investigate three main aspects concerning the prediction of the performance of IR systems:

- Can we exploit the knowledge gained on query variations to predict which query among a set of formulations for the same information need will be the best if used in the context of the systematic review?

- Can we predict which category of systems will find a given query particularly challenging? More in detail, given a query, are we capable of determining if systems based on lexical IR models and systems relying on semantic ones will perform better?

- Given the challenges linked to the evaluation of the previous two approaches, but also the QPP evaluation in general, how can we improve how we evaluate the performance of QPP models?

The remainder of Part II is organized as follows: Chapter 6 reports the description of the proposed QPP model to predict the recall in the systematic review scenario and select the best query. Chapter 7 contains our analyses concerning a predictor capable of determining the “semantic” query complexity. Finally, given the challenges presented by the evaluation of the approaches mentioned before, Chapter 8 outlines a new measure, dubbed sMARE, that helps in evaluating QPP models.

Chapter 6

A Gain-based Approach to Predict Recall for the Systematic Review Task

6.1 Introduction

The study of the query representation in Information Retrieval has driven a lot of interest in recent years [8, 9, 19, 170, 200, 43]. Several works in the past [26, 158, 14] showed the positive effect on the retrieval results of fusing runs retrieved with human-made multiple formulations of the same information need. Recent studies have shown how query reformulations automatically extracted from query logs can be as effective as those manually created by users [101]. Furthermore, the performance of a system can greatly improve when the "right" formulation of an information need is selected [170, 200]. One of the main challenges in this research area is being able to suggest the best performing query (or queries) among the possible variations [46, 170, 151, 200, 54]. For example, Thomas et al. [170] observed that, the most prominent effect in predicting the performance of a query formulation is due to the information need and not to the "*query wording*". In this sense, query performance predictors actually predict the complexity of the information need, rather than the one the query itself. Zendel et al. [200] pursue a slightly different task. Following the literature on reference lists [160, 136] they try to predict the performance for a query using information about queries representing the same information need. Benham et al. [19] define a fusion approach for multiple query formulations based on the concept of "topic centroid", which describes the information need as combination of its formulations. Dang et al. [46] address also the problem of improving the ranking results through a query formulation selection phase. Note that, Dang et al. [46] show how they are often capable of putting the best query in the first *two* positions (not only the first one), a further evidence of the complexity of the task.

A use case of query performance prediction is the systematic compilation of literature review. In fact, systematic reviews are scientific investigations that use strategies to include a comprehensive search of all potentially relevant articles. As time and resources are limited for compiling a systematic review, limits to the search are needed: for example, one may want to estimate how far the horizon of the search should be (i.e. all possible cases/documents that could exist in the literature) in order to stop before the resources are finished [84]. Scells et al. [151] apply several state-of-the-art Query Performance Predictors to select the best query in the Systematic Reviews domain. They show how current Query Performance Prediction approaches perform poorly on this specific task. International evaluation campaigns have organized labs in order to study this problem in terms of the evaluation, through controlled simulation, of methods designed to achieve very high recall [134, 69]. The CLEF initiative[1] has promoted the eHealth track since 2013 and, the CLEF 2018 eHealth Evaluation Lab Consumer Health Search (CHS) task [82] investigated the problem of building search engines that are robust to query variations to support information needs of health consumers.

In this Chapter, we study an alternative formulation of the intent-aware metric proposed by Umemoto et al. [174], in which the authors analyze a metric to estimate the amount of missing information for each query reformulation during a search session. Note that in [174] the authors do not propose an approach capable of predicting the recall of different formulations. Nevertheless, our perception is that, their approach can be easily adapted with good results also to the predictive task. In our case, our research goal is to understand whether a gain based approach can be used to predict the relative importance of each reformulation in terms of recall performance, in the context of Consumer Health Search where users need support for medical information needs.

In this sense, with respect to [174], our contribution is two-fold and can be summarized with the following research questions:

- **RQ 6.1** is it possible to apply the GAIN measure proposed in [174] to obtain a recall predictor over a set of formulations for the same topic?
- **RQ 6.2** how can we improve the results of such predictor by exploiting also the information obtained through the various formulations?

[1] http://www.clef-initiative.eu

6.2 Estimating the Query Gain

Umemoto et al. [174] define the intent-aware gain metric and the requirements that it should satisfy. They identify the following properties: *importance*, documents relevant to a central aspect of the search topic produce higher gain than those relevant to a peripheral one; *relevance*, highly relevant documents produce higher gain than partially relevant ones; *novelty*, documents relevant to an unexplored aspect produce higher gain than those relevant to a fully explored aspect.

The set of aspects A_t of a topic t is estimated through the process described in [172]: first, a set of subtopics S_t is mined given a topic t; then, the subtopics are grouped into a set of clusters C_t. These clusters are regarded as the "facets"[2] of t. The most representative subtopic s is chosen from each cluster as formulation of the topic aspect a using the formula $a = argmax_{s \in C_t} \mathrm{Imp}_t(s)$, where the importance of a subtopic s is defined as:

$$\mathrm{Imp}_t(s) = \sum_{d \in D_s^N \cap d \in D_t^N} \frac{1}{\mathrm{Rank}_t(d)} \tag{6.1}$$

D_s^N and D_t^N denote the sets of the top N retrieved documents for a subtopic s and the topic t, respectively, and $\mathrm{Rank}_t(d)$ is the rank of the document d in the ranked list for t.

It is crucial to stress that the definition of *importance*, and the following definition of *gain*, derives from the assumption that there is a known "reference" topic t that describes completely the information need. For such topic t the retrieved documents can be different compared to the ones observed for a query which represents just one aspect a of the topic. In Fig. 6.1, we show an example of a number of subtopics found for a topic t and grouped into three clusters, each one with a representative aspect.

The Intent-Aware Gain is defined for a set of documents D as:

$$\text{Gain-IA}_t(D) = \sum_{a \in A_t} P(a|t) \cdot \mathrm{Gain}_{t,a}(D) \tag{6.2}$$

which is a sort of expected value of the gain across the different aspects. $P(a|t)$ is the probability that an aspect a is important to the topic t, and $\mathrm{Gain}_{t,a}(D)$ is the gain that can be obtained by the aspect a from the documents D. The importance probability for an aspect of a topic is computed as:

$$P(a|t) = \frac{\mathrm{Imp}_t(a)}{\sum_{a' \in A_t} \mathrm{Imp}_t(a')} \tag{6.3}$$

[2] We use *facets* instead of *aspects* to not repeat the same term that will be use to identify the most representative subtopic.

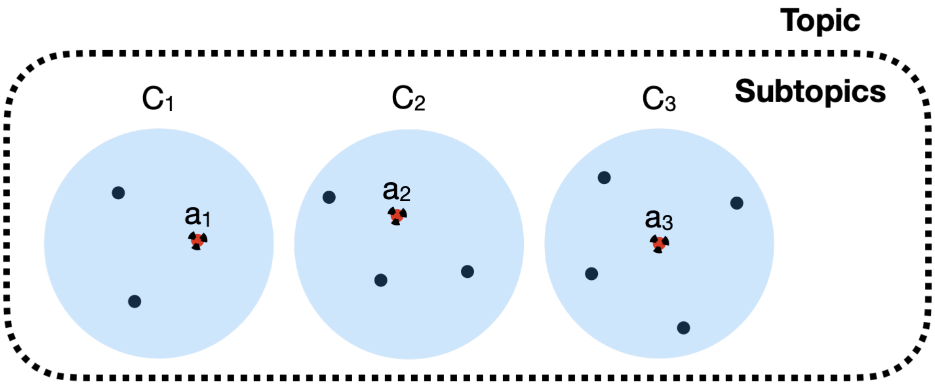


Fig. 6.1 An example of clusters of subtopics and aspects

while the gain which measures how the documents D retrieved for a query contribute to increment the information relative to a specific aspect of the topic is:

$$\text{Gain}_{t,a}(D) = \left[1 - \prod_{d \in D} (1 - \text{Rel}_{t,a}(d))\right] \tag{6.4}$$

This last part that is required to compute the Intent-Aware Gain contains the term $\text{Rel}_{t,a}(d)$ which is the relevance degree of a document d with respect to an aspect a, estimated as follows:

$$\text{Rel}_{t,a}(d) = \frac{\sum_{s \in C_a} \text{Imp}_t(s) \cdot \text{Rel}_s(d)}{\sum_{s \in C_a} \text{Imp}_t(s)} \tag{6.5}$$

where $C_a \in C_t$ is the cluster of subtopics belonging to the aspect a, and $Rel_s(d)$ is the relevance degree of a document d to a subtopic s estimated as $\text{Rel}_s(d) = 1/\sqrt{\text{Rank}_s(d)}$.

6.3 A Gain-based Approach to Predict Query Reformulations Performance

Our initial hypothesis is that: a) we have one information need expressed with different query reformulations, and b) the topic t is unknown. In particular, given an information need i and its set of reformulations V_i, we assume that each reformulation $q \in V_i$ is able to 'reveal' different facets of i. Consequently, we need to redefine the expression of the gain of Eq.6.4 as:

$$\text{Gain}_{i,q}(D) = \left[1 - \prod_{d \in D} (1 - \text{Rel}_{i,q}(d))\right] \tag{6.6}$$

where i is the *information need* and q is a specific (re)formulation.

The main difference with the original approach, apart from changing variable names, is the fact that i) we do not have a 'reference' topic t that describes completely the information need i, and ii) we have one single cluster of query reformulations, or *variants*, V_i. For these reasons, we also need an alternative definition of relevance that adapts to our case study:

$$\text{Rel}_{i,q}(d) = \frac{\sum_{s \in V_i} \text{Imp}_q(s)\text{Rel}_s(d)}{\sum_{s \in V_i} \text{Imp}_q(s)} \tag{6.7}$$

where the relevance of d, retrieved by the query variant q of the information need i, is computed as the weighted average of the relevance of d with respect to all the alternative reformulations in V_i. The two terms $\text{Imp}_q(s)$ e $\text{Rel}_s(d)$ remain unaltered compared to the previous definitions:

$$\text{Imp}_q(s) = \sum_{d \in D_s^N \cap D_q^N} \frac{1}{\text{Rank}_q(d)} \quad , \quad \text{Rel}_s(d) = \frac{1}{\sqrt{\text{Rank}_s(d)}}$$

A Similarity Matrix for Recall Prediction

In the proposed context, we can think of an 'optimal' query as the one capable of combining all the diverse facets of the information need it represents. In order to estimate which query reformulation q is the closest to the unknown optimal one, we propose the following procedure:

1. we define D_q as the set of documents retrieved by q;
2. $D_i = \bigcup_{q \in V_i} D_q$ as the set of all documents retrieved by *at least* one reformulation q;
3. $\mathbf{R} \in \mathbb{R}^{|V_i| \times |D_i|}$ as the matrix of rankings for the information need i where each row corresponds to a specific reformulation and each column to a document. The value of an element $r_{k,d}$ of $\mathbf{R}$ is defined as $|D_q| - \rho_{q,d}$ where $\rho_{q,d}$ is the rank of document d retrieved by q. $\mathbf{R}$ is at the end normalized with norm $l2$.

At this point, we want to build a similarity matrix to predict the impact in terms of recall that each reformulation will have on the retrieval. We compute the cosine similarity between each pair of rows in $\mathbf{R}$, obtaining a symmetric matrix $\mathbf{S}$ where each row (or column) represents how a reformulation is similar to the others. We use the sum the k-th row (or column) of $\mathbf{S}$ to predict the importance of the k-th query; then, we order the query reformulations in decreasing order where greater values indicate a higher probability of retrieving more relevant documents. This measure describes how close each query is to the ideal "centroid" query that perfectly describes the topic.

6.4 Experiments and Analysis

In this section, we describe the analysis of our experiments. In particular, we want to compare the performance in terms of predicted recall among: i) the gain defined in Eq 6.6, ii) an alternative definition that mitigates some arithmetical issues, iii) and the similarity matrix. To the best of our knowledge, this is the first effort in predicting the recall for the systematic reviews task, when multiple formulations are considered. Therefore, we are not able to directly compare it with an approach explicitly thought for such task. We thus compare our solution with traditional QPP strategies. Furthermore, we use the techniques presented in Umemoto et al. [174] as baselines.

6.4.1 Test Collection and Retrieval Model

The CLEF 2018 eHealth Evaluation Lab Consumer Health Search (CHS) task [82] investigated the problem of retrieving Web pages to support information needs of health consumers that are confronted with a health problem or a medical condition. One subtask (i.e., subtask 3) of this lab is aimed to foster research into building search systems that are robust to query variations.[3]

Queries There are 50 information need for which we have 7 query reformulation for a total of 350 queries: the original 50 queries issued by the general public augmented with 6 query variations issued individually by 6 research students with no medical knowledge.[4]

Collection The collection contains 5,535,120 Web pages and it was created by compiling Web pages of selected domains acquired from the CommonCrawl [82].

Relevance Assessments For each information need, the organizers of the task provided about 500 documents assessed for a total of 25,000 topic-document pairs.

Retrieval Model The index provided by the organizers of the task, an ElasticSearch index version 5.1.1, comes with a standard BM25 model with parameters b = 0.75 and k1 = 1.2.[5]

Notice That, among the queries of the CLEF 2018 eHealth CHS collection, the two identified by ids *160006* and *164007* will not retrieve any document in common with the other variants of the same information need (at least for $N \leq 1000$). This is because the text of query 160006 is “nan”, while query 164007 has a typo “pros and cons spirculina”, instead of spirulina, a type of algae. We stress on this aspect since, for those queries, it will not be

[3] https://github.com/CLEFeHealth/CLEFeHealth2018IRtask

[4] The queries and the process to obtain them are described in http://www.khresmoi.eu/assets/Deliverables/WP7/KhresmoiD73.pdf

[5] https://sites.google.com/view/clef-ehealth-2018/task-3-consumer-health-search

Table 6.1 Pre- and Post-retrieval predictive baseline models considered.

type	predictor
pre-retrieval	MaxIDF [155] AvgIDF [42] StdIDF [42] SumSCQ [204] AvgSCQ [204] MaxSCQ [204]
post-retrieval	WIG [206] NQC [162] SMV [169]

possible to compute the value of the gain by definition, since the intersection of their ranked list with the ones for other formulations of the same topic will be empty.

6.4.2 Using traditional Query Performance Predictors applied to recall prediction for systematic reviews

To have a better grasp on the peculiarities of the problem, we first try to apply traditional techniques of Query Performance Prediction (QPP) to our specific setting. We aim at showing that, traditional QPP techniques fail to correctly order formulations when *i)* the recall is the key performance indicator; *ii)* we sort formulations of the same topic and not queries representing different topics. Showing this, is a further evidence of the importance of using appropriate tools, such as the *gain* as described in Subection 6.3 to correctly tackle the problem. More in detail, we select a set of very well-know QPP models, in order to determine whether they can be satisfactory applied to the prediction of the recall and can be used with the documents and queries that we have at hand. Table 6.1 reports the predictors that we include in our analyses and a brief description of how they work. It is important to notice that, as for many QPP models, the models that we selected do not actually predict the performance measure. They associate a score to each of the queries, which is expected to correlate with the performance measure, but is on a different scale and cannot be used directly as estimate of the performance.

Notice that, there are two main aspects that might impair traditional QPP models in our specific setting:

- Remember that we are in the setting of the systematic reviews. Therefore, it is by far more important to retrieve as many as possible relevant documents, rather than putting

Table 6.2 Kendall's τ correlation observed between recall and prediction scores for both pre- and post-retrieval traditional predictors, if we compare the default formulations of different topics. Results are in line with correlation values previously observed in other scenarios. The symbol † indicates that the correlation is statistically greater than 0 at level $\alpha = 0.05$, while the ‡ indicates a significance level of 0.01, the absence of any symbol indicates that results cannot be deemed statistically greater than 0. We compute the Kendall's τ correlation at different cutoff levels of the ranked lists (100, 1000, and 10000).

type	**predictor**	**kendall's τ**		
		100	**1000**	**10000**
pre-retrieval	MaxIDF	0.3185‡	0.3260‡	0.2875‡
	AvgIDF	0.2996‡	0.3218‡	0.2555‡
	StdIDF	0.2947‡	0.2989‡	0.2343†
	SumSCQ	0.2637‡	0.2581‡	0.1739
	AvgSCQ	0.3479‡	0.3652‡	**0.3299**‡
	MaxSCQ	**0.3502**‡	**0.3724**‡	0.2833‡
post-retrieval	WIG	0.3029‡	0.3218‡	0.2882‡
	NQC	0.2865‡	0.1911	0.1135
	SMV	0.1797	0.1332	0.0229

them in the first positions. Therefore, we are not interested in estimating the AP, which is a precision based measure, but our aim is to predict which query will have the best recall;

- We do not compare queries meant for different information needs, which is the typical evaluation scenario for QPP models.

On the other hand, we aim at understanding which one, among a set of queries representing the same information need, achieve the best result.

To determine whether we are impaired by the first problem, we first apply the traditional QPP considering only the default formulation of each topic, and we compare whether the predictors are capable of correctly determining the inter-topic performance. More in detail, with this first experiment, we are interested in understanding whether the baseline predictors are capable of predicting which *topic* will have the best recall, using a single formulation for each of them. Table 6.2 reports the result of such analysis.

We can observe that, by looking at Table 6.2, the results are in line with previous similar experiments in the literature, such as [55, 200]. Almost all the predictors are able to achieve a significant correlation with the recall (with level $\alpha = 0.01$). Two noticeable exceptions are represented by nqc and smv: traditionally, they are considered among the best predictors, but in this specific scenario they fail, with correlations not statistically different from 0. Our hypothesis is that, while pre-retrieval predictors tend to be estimators of the recall base of a query, and therefore tend to correlate with the recall itself, post-retrieval predictors tend to compute their predictors based on the scores that the retrieval model assigns to the top-ranked documents. In this sense, post-retrieval predictors are “top-heavy”: they focus on the upper part of the ranked list of documents. This behaviour favours predicting the performance for top-heavy measures, such as Average Precision or nDCG. Instead, our task consists in predicting the recall, given a *long* list of documents. It is not unlikely that the upper part of the list of retrieved documents is saturated with relevant ones; nevertheless, we are more interested in being sure that *every* relevant document has been considered, rather than saying whether the top part of the ranked list contains relevant documents.

We now switch the focus from predicting the performance *across* topics, to predict the performance *within* topics. Instead of comparing the performance that the standard formulation is expected to achieve for each topic, we try to sort different formulations for the same topic, according to the predicted performance. Table 6.3 reports the results of our analysis.

Compared to the results observed in Table 6.2, the performance achieved by traditional predictors for the “within”-topics prediction, is extremely lower, with very few cases of significantly positive correlation between the predicted and observed recall. Note that, even though we agree with [151] on the fact that predicting the best query among a series of formulations of a topic is a hard task, we end up with diametrically opposite conclusions. Scells et al. [151] observed severe flaws in traditional QPP techniques when *predicting the performance across topics*. On the other hand, they found the task of predicting the performance within topics (which they refer to as Query Variation Performance Prediction (QVPP)) to be easier, achieving higher (although still very low) results. What we observe here, is diametrically opposite: we found the worst results when predicting results within topics, and performance in line with previous literature for the predictions across topics. A possible explanation for this phenomenon is that we use the traditional QPP models for a different task compared to Scells et al. [151]. In fact, our aim is to predict the recall, while Scells et al. [151] aim at predicting the Average Precision. As a final remark, we want to point out that, Zendel et al. [199] recently showed how the “QVPP” is a harder task, compared to traditional QPP, confirming in this sense our findings.

6.4.3 Comparing Different Gain-based Approaches

Given what we observed in Subsection 6.4.2, we are interested in understanding whether the GAIN-based proposed by [174] (cfr. Equation 6.4) can overcome the problems in this

Table 6.3 Performance achieved by traditional predictors, applied to our specific case. Each predictor has been used to predict the performance of the different formulations. We report the mean score and standard deviation of the correlation computed over the different topics. We also report the first quartile, third quartile and number of topics (over the 50 available) for which the correlation between the predicted and observed recalls for their (re)formulations is significantly greater than 0.

				kendall's τ		
type	**predictor**	**cutoff**	**Q1**	**mean (std)**	**Q3**	**sign.**
pre-retrieval	MaxIDF	**100**	-0.5417	-0.1085 (0.4825)	0.1183	3
		1000	-0.5295	-0.0973 (0.4755)	0.2263	1
		10000	-0.5699	-0.1227 (0.4628)	0.1584	2
	AvgIDF	**100**	-0.4214	-0.0449 (0.5272)	0.3333	4
		1000	-0.4821	-0.0617 (0.4898)	0.2167	4
		10000	-0.4214	-0.0549 (0.4698)	0.2473	3
	StdIDF	**100**	-0.4880	-0.1606 (0.4927)	0.0915	4
		1000	-0.4190	-0.0999 (0.5107)	0.1938	**5**
		10000	-0.4064	-0.1537 (0.4347)	0.1576	1
	SumSCQ	**100**	-0.2381	-0.0102 (0.4021)	0.2381	1
		1000	-0.2985	**0.0893 (0.4276)**	0.4000	4
		10000	-0.3126	0.0150 (0.4558)	0.2750	**5**
	AvgSCQ	**100**	-0.3250	**0.0322 (0.5231)**	0.3901	**6**
		1000	-0.3898	0.0135 (0.4838)	0.3333	**5**
		10000	-0.3250	**0.0005 (0.4505)**	0.2985	3
	MaxSCQ	**100**	-0.3541	-0.0369 (0.4333)	0.2765	1
		1000	-0.3341	-0.0312 (0.4447)	0.2568	2
		10000	-0.3459	-0.0484 (0.4441)	0.1912	2
post-retrieval	WIG	**100**	-0.6790	-0.1743 (0.5206)	0.1376	4
		1000	-0.4088	**-0.0266 (0.5031)**	0.2519	**6**
		10000	-0.4214	**0.0171 (0.5185)**	0.3849	**6**
	NQC	**100**	-0.5611	**-0.0880 (0.5554)**	0.2381	**6**
		1000	-0.4405	-0.1244 (0.5004)	0.1511	4
		10000	-0.4850	-0.1539 (0.4991)	0.1539	3
	SMV	**100**	-0.5621	-0.1653 (0.4836)	0.1849	1
		1000	-0.5542	-0.1626 (0.4604)	0.1859	0
		10000	-0.6243	-0.2207 (0.4882)	0.0994	2

specific setting shown by traditional QPP models. The results are shown in Figures 6.2, 6.5 and 6.8. Each figure is divided into two parts: top, we show the distribution of values of the GAIN (or similarity), ordered increasingly, for each query reformulation (350 in total); bottom, we plot for each topic (50 topics) the value of the correlation Kendall τ between the query reformulations ordered by decreasing GAIN (or similarity) and the reformulations ordered by decreasing true recall. The blue dots indicate a statistically significant correlation greater (or lower) than zero, while black dots the topics for which it is not possible to compute the correlation.

Saturated GAIN distribution

In Figures 6.2, 6.5, and 6.8, we show that the value of the gain saturates to 1 for most query reformulation. This is more evident when we increase the number of documents N of Eq. 6.6 from $N = 100$ up to $N = 10000$. This behavior, due to the importance in Eq. 6.1 that multiplies N numbers less than one, makes the GAIN not useful to discriminate the different query variants of an information need, since every variant will have gain equal to 1. In addition, when all the reformulations have the same gain, it is impossible to compute the Kendall τ correlation to predict the performance (black dots with correlation value 0 in the figure). Being not saturated is not by itself a desirable feature for the gain measure. Nevertheless, the faster the gain saturates, the harder it is to discriminate between different formulations. In this sense, a GAIN measure capable of spreading better the options in the entire domain is preferable.

Alternative GAIN Definition

In order to mitigate the aforementioned problems, we propose an alternative definition of the gain of Eq. 6.6 substituting the product with an average:

$$GAIN_{i,q}(D) = \left[1 - \frac{\sum_{d \in D}(1 - \mathrm{Rel}_{i,q}(d))}{|D|}\right] \tag{6.8}$$

The results of this new formulation are shown in Figures 6.3, 6.6, and 6.9. The distribution of the gain is more spread across all the reformulations and does not saturate to one. There is also a more stable prediction of the performances for each topic: the number of statistically significant predictions of the recall of the reformulation is between 17 and 19, from $N = 100$ and $N = 10000$; in addition, the number of negative correlations (wrong predictions of performance) decreases. This indicates (as we may expect) that with more information (more documents, greater N) we can predict better the order of importance, in terms of recall, of each reformulation.

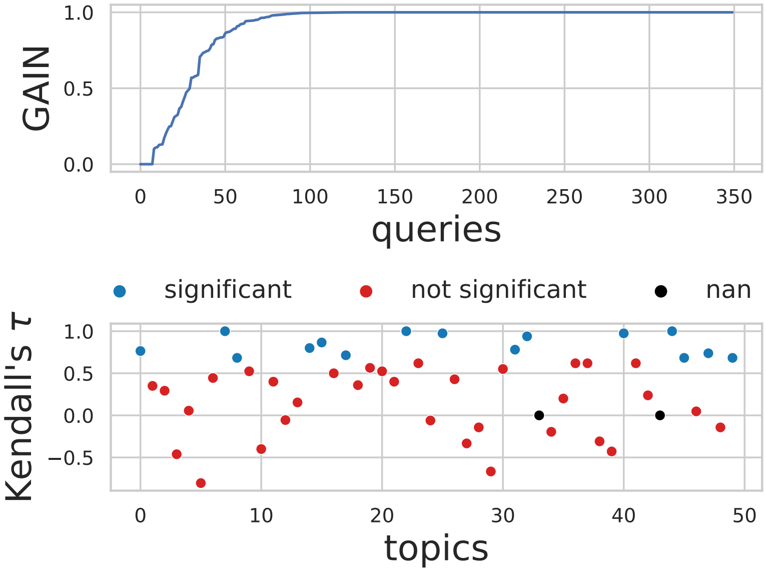


Fig. 6.2 N=100. Distribution of the GAIN, ordered increasingly, of the 350 queries and correlation between the reformulations ordered by predicted GAIN (or similarity) and the reformulations ordered by the true recall, using the GAIN as proposed in [174]

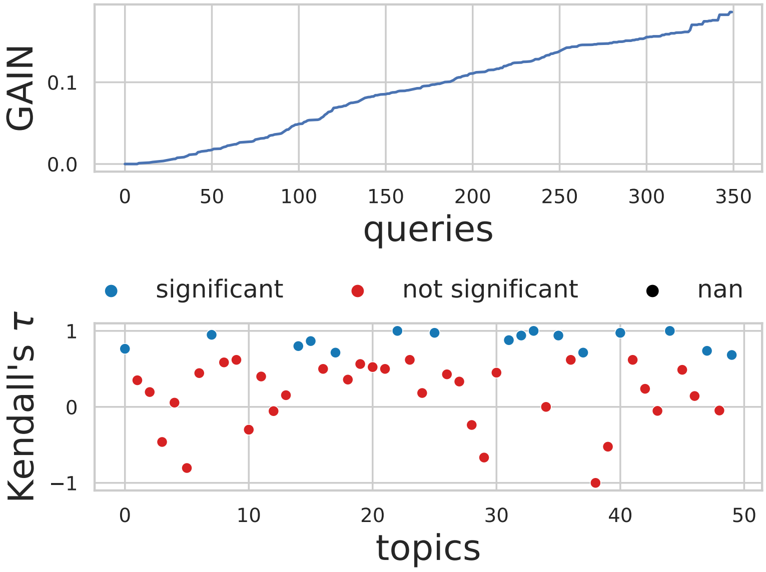


Fig. 6.3 N=100. Distribution of the GAIN, ordered increasingly, of the 350 queries and correlation between the reformulations ordered by predicted GAIN (or similarity) and the reformulations ordered by the true recall, using the GAIN with the mean aggregation strategy

Using Similarity Matrix for Recall Prediction

In Figures 6.4, 6.7, and 6.10, we show the ability to predict the performance of a query reformulation using the correlation between the similarity-based approach presented in

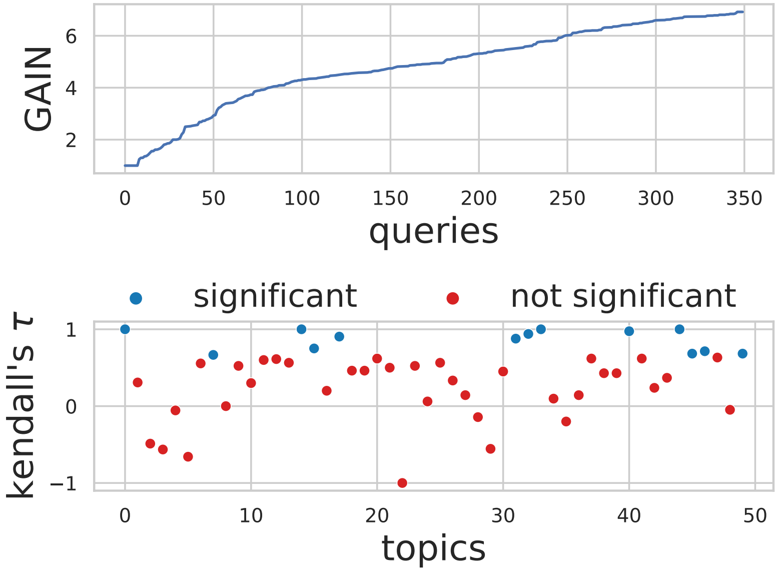


Fig. 6.4 N=100. Distribution of the GAIN, ordered increasingly, of the 350 queries and correlation between the reformulations ordered by predicted GAIN (or similarity) and the reformulations ordered by the true recall, using the GAIN with the similarity-based aggregation strategy

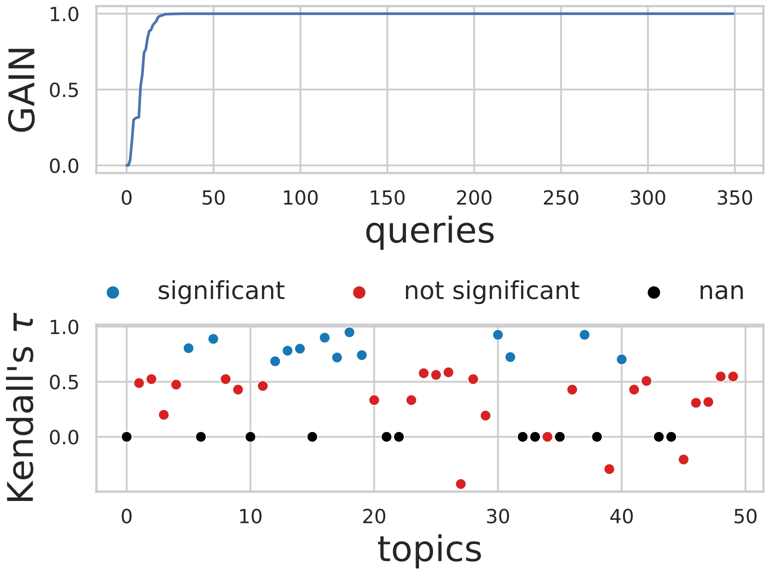


Fig. 6.5 N=1000. Distribution of the GAIN, ordered increasingly, of the 350 queries and correlation between the reformulations ordered by predicted GAIN (or similarity) and the reformulations ordered by the true recall, using the GAIN as proposed in [174]

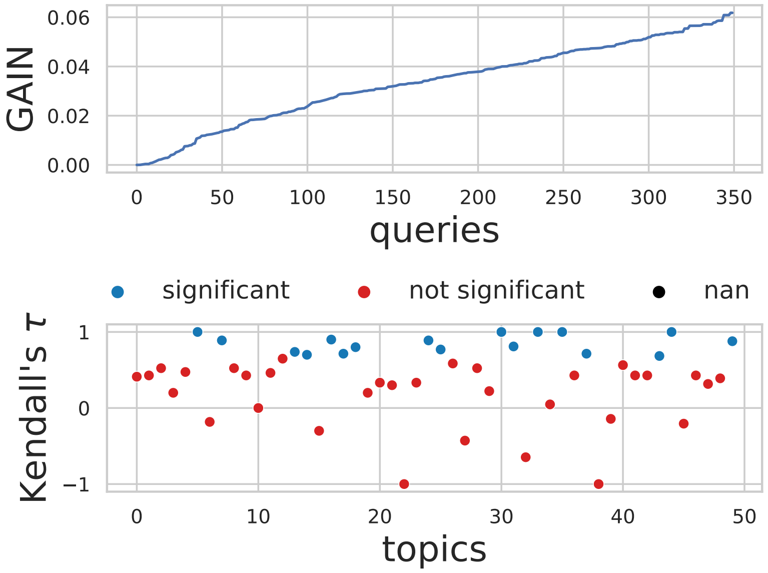


Fig. 6.6 N=1000. Distribution of the GAIN, ordered increasingly, of the 350 queries and correlation between the reformulations ordered by predicted GAIN (or similarity) and the reformulations ordered by the true recall, using the GAIN with the mean aggregation strategy

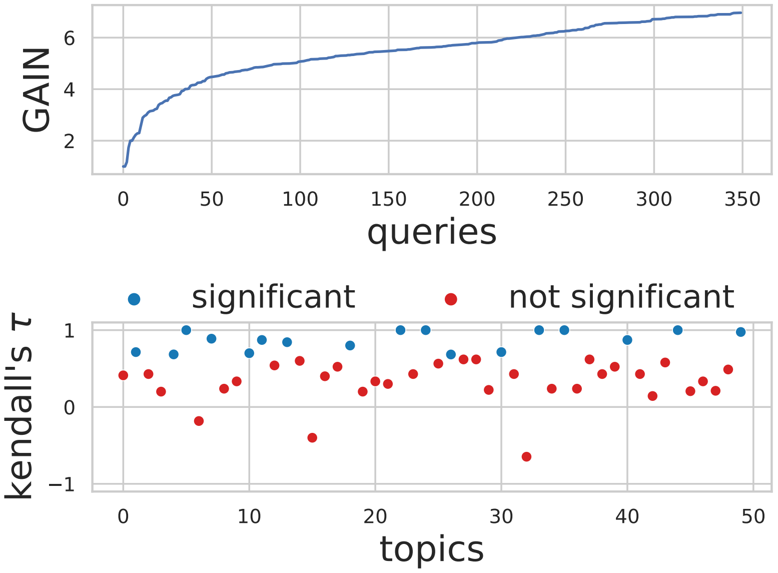


Fig. 6.7 N=1000. Distribution of the GAIN, ordered increasingly, of the 350 queries and correlation between the reformulations ordered by predicted GAIN (or similarity) and the reformulations ordered by the true recall, using the GAIN with the similarity-based aggregation strategy

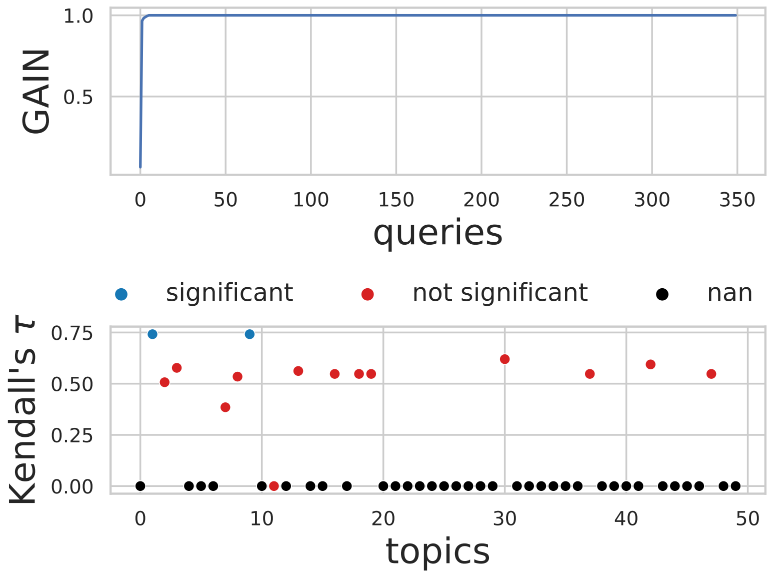


Fig. 6.8 N=10000. Distribution of the GAIN, ordered increasingly, of the 350 queries and correlation between the reformulations ordered by predicted GAIN (or similarity) and the reformulations ordered by the true recall, using the GAIN as proposed in [174]

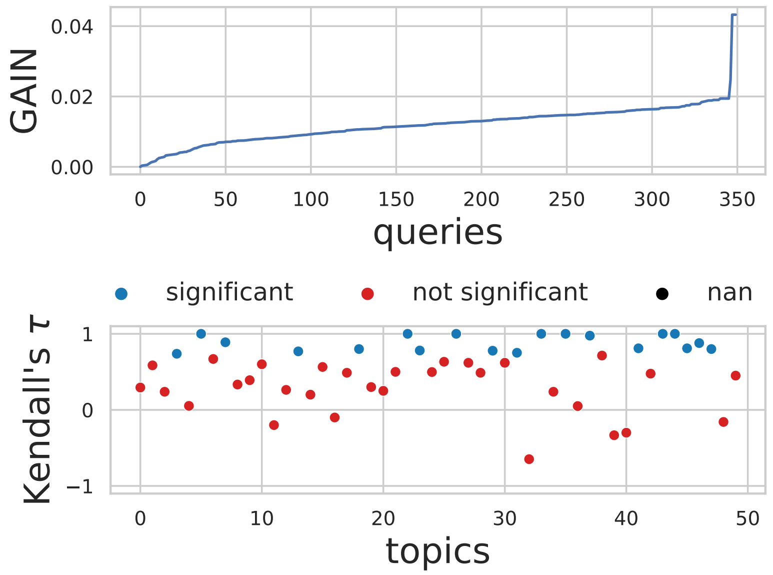


Fig. 6.9 N=10000. Distribution of the GAIN, ordered increasingly, of the 350 queries and correlation between the reformulations ordered by predicted GAIN (or similarity) and the reformulations ordered by the true recall, using the GAIN with the mean aggregation strategy

Sec. 6.3. The values of the Similarity are spread and do not saturate to the maximum value of the sum of a row of S (in our experiments equal to 7). By increasing the number N of

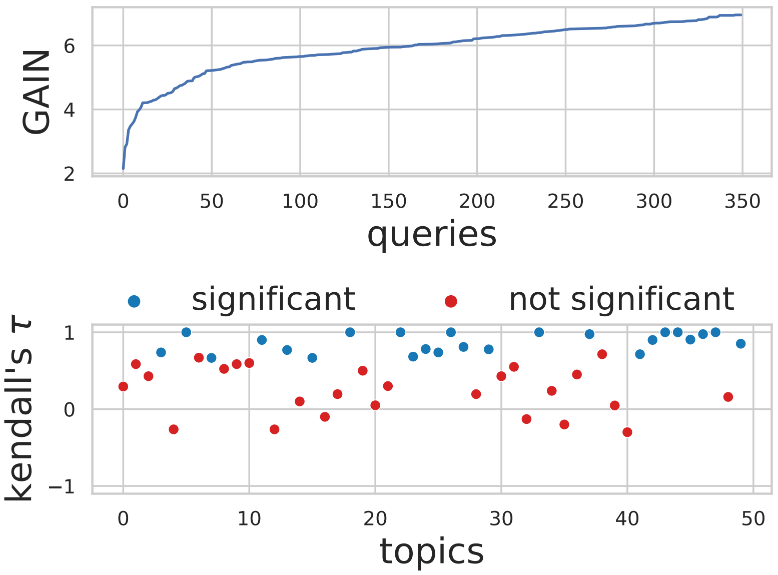


Fig. 6.10 N=10000. Distribution of the GAIN, ordered increasingly, of the 350 queries and correlation between the reformulations ordered by predicted GAIN (or similarity) and the reformulations ordered by the true recall, using the GAIN with the similarity-based aggregation strategy

documents, we improve the capability to predict the performance of the query reformulation; in particular, there are no statistically significant negative correlation and the total number of negative correlations decreases from $N = 100$ to $N = 10000$.

Besides the qualitative aspects, Table 6.4 reports also the numerical performance comparison between the GAIN as proposed by [174], its version which employs the mean, and the similarity-based gain.

6.5 Final Remarks

In this last section of the analysis of the results, we want to briefly summarize our findings. As a remainder, we want to point out that, the GAIN measure proposed by [174], was originally used to estimate the missing information that the user could have gained, by using different subtopic formulations, showed in a user-interface. Although such task shares similar aspects with the one of predicting the recall, they are not fully overlapping. Our main contributions in this Chapter are:

- First, adapting an already established technique to a different task. In this sense, to the best of our knowledge, this is the first effort in adapting the GAIN measure proposed by Umemoto et al. [174] to the query formulation recall prediction task.
- Secondly, its "mean" version, which we refer to as "Mean Gain", is observed here for the first time, as a better adaptation of [174] to the predictive task.
- Finally, the Similarity-based Gain is a completely new contribution of this manuscript, which exploits similar elements to the gain measure proposed by Umemoto et al. [174].

Table 6.4 Kendall's τ correlation observed for the task of predicting the query formulation recall, using the similarity based approaches. In bolt, best mean score for each cutoff. Note that all the methods considered perform better than traditional predictors(cft Tbl. 6.3). We also have a higher number of significant rankings compared to the one observed before.

		Kendall's τ			
predictor	**cutoff**	**Q1**	**mean (std)**	**Q3**	**sign.**
Original Gain [174]	**100**	0.0000	0.3422 (0.4696)	0.6831	15
	1000	0.0000	0.3783 (0.3527)	0.6609	13
	10000	0.0000	0.1600 (0.2606)	0.4765	2
Mean Gain	**100**	0.1456	**0.3822 (0.4936)**	0.7320	16
	1000	0.2417	0.4042 (0.4796)	0.7320	17
	10000	0.2709	0.5111 (0.3984)	0.8000	19
Similarity based Gain	**100**	0.1429	0.3768 (0.4636)	0.6581	13
	1000	0.3083	**0.5069 (0.3505)**	0.7143	17
	10000	0.2521	**0.5443 (0.3930)**	0.8876	24

Table 6.4 shows that the similarity based gain has the overall best performance both compared to other gain based measures and traditional predictors (cfr. Table 6.3). Interestingly, while the original gain worsen with the increase of the cutoff (as observed both in Tables 6.2 and 6.3), both the mean based and the similarity one tend to improve their performance when the cutoff increase. The original gain suffers of the "saturated gain", as reported in 6.4.3, while our proposal (both "mean" and "similarity" versions) improve as new relevant information is added.

Chapter 7

Predicting the Semantic Difficulty Query-Wise

7.1 Introduction

The semantic gap is a long-standing problem in IR that refers to the difference between the machine-level description of document and query contents and the human-level interpretation of their meanings [91]. In other words, it represents the mismatch between users' queries and the way retrieval models understand such queries [203].

The semantic gap affects any domain, but it is prominent in medical search [50, 90, 91]. Within biomedical literature, the large presence of (quasi-)synonymous and polysemous terms – along with the use of acronyms and terminological variants – represents a critical challenge for retrieval models. In this regard, a query containing the word "tumor" might not be effectively answered if the retrieval model does not identify the synonymy relationship between "tumor" and, for example, "neoplasm". Besides, given a query containing the term "cold", a retrieval model might retrieve erroneous documents if it does not distinguish between the different meanings the term "cold" assumes depending on the context, such as "common cold", "cold temperature", or even "Chronic Obstructive Lung Disease". These queries are known as *semantically hard* queries [1].

Traditional IR models, which are known as lexical models as they compute the relevance score using heuristics defined over the lexical overlap between queries and documents, fail to effectively address semantically hard queries. Semantic models were thus introduced to bridge the semantic gap [99] and to overcome the limitations of lexical models. However, semantic models have been shown to provide complementary signals to lexical models that prove effective for semantically hard queries, but less for other queries [106]. Thus, it becomes necessary to identify what category of models – between lexical and semantic – best suits a user query given the document collection at hand. In other words, we need to understand what are the inherent features of query and documents that make lexical or semantic models more effective.

To this end, we address the following research questions:

RQ 7.1 How and to what extent does the semantic gap impact query performance?

RQ 7.2 What features determine the prominence of the semantic gap within queries?

The research questions mentioned before are addressed in Chapter 7.

For RQ 7.1, we investigate and compare the impact of lexical and semantic models on different topics. How large is the interaction between topics and model categories? To what extent does this interaction reflect in the different topic formulations (i.e., queries)?
For RQ 7.2, we explore a different set of well-known features that relate to lexical and semantic models. In particular, we seek to understand whether pre-retrieval features – based on corpus statistics or synonymy/polysemy aspects – can be used to categorize queries as semantically easy or hard. In other words, how effective are well-known pre-retrieval features for category selection?

To address the research questions, we first perform statistical analyses quantifying the interaction between topics, queries, and lexical and semantic categories using ANOVA [140]. Based on the outcomes of the statistical analyses, we propose a labeling strategy to categorize queries into semantically easy or hard. The labeled queries are used to train a category selector. The selector serves as a proxy to evaluate the effectiveness of the considered pre-retrieval features in determining the prominence of the semantic gap within queries.

We conduct an experimental evaluation on two test collections for ad hoc medical retrieval: OHSUMED [77] and TREC-COVID (Round 1) [182]. For lexical models, we adopt standard state-of-the-art retrieval models. Regarding semantic models, we focus on first-stage semantic models, which are best suited to tackle the semantic gap [179]. In particular, we consider unsupervised first-stage semantic models, which have shown to be competitive with lexical models in medical collections [1]. Besides, unsupervised semantic models rely on textual signals only – and not on relevance signals – thus allowing us to focus exclusively on semantic and lexical features.

The results of the experimental evaluation show that topics, queries, and model categories strongly interact to determine retrieval effectiveness. This evidence further highlights the need to adopt the proper model category to improve retrieval performance. Therefore, identifying the right features to distinguish between semantically easy or hard queries becomes necessary in domains where the semantic gap is prominent – and this work poses the cornerstone towards this direction.

However, we refrain from using rank-time or post-retrieval features in our analyses as we want to keep the approach model-agnostic – and thus less dependent on the specific sets of considered retrieval models.

Finally, we are interested in determining if it is possible to identify queries as either semantic or lexical automatically and prior to their empirical evaluation. We define "semantic" – and lexical in opposition – queries for which semantic models perform on average better. Even though the definition itself of "semantic models" is up to debate, we consider it to be "semantic" a model that tries to define a semantic representation on the tokens (e.g., models based on word embeddings). Conversely, we use "lexical" for those models that rely on the lexical similarity between the query and the document (e.g., BM25, Language Model, TF-IDF). Prior work, such as [106] investigated the interaction and synergy between queries and specific categories of models, showing that often queries that are particularly easy for lexical models – for example, due to few very relevant terms identifying relevant documents – tend to be hard for semantic models and vice-versa. We are then interested in determining if it is possible to exploit the query's features to predict whether semantic or lexical models will perform better.

We focus on pre-retrieval approaches and we adopt two types of features in our analyses: lexical- and semantic-oriented features. Regarding lexical-oriented features, we consider features proposed by He and Ounis [76] and by Zhao et al. [204]. He and Ounis [76] explore the possibility to use the distribution of the IDF over query terms to determine the ability of lexical models to retrieve relevant documents. Similarly, Zhao et al. [204] propose a re-weighting schema based on IDF, called SCQ. As for semantic-oriented features, we adopt features similar to those proposed by Mothe and Tanguy [116], who consider linguistic aspects – such as synonymy and polysemy – linked to the query terms. Compared to [116], however, we consider signals from both the query and its interaction with documents.

We consider two collections in the following analyses: OHSUMED [77] and TREC-COVID (Round 1) [182].

OHSUMED contains 349K documents and 63 topics. Topics in OHSUMED have two fields: *title* and *description*. We use *description* as topic formulation since the *title* field poorly describes the underlying information need. TREC-COVID (Round 1) has 30 topics and relies on the CORD-19 corpus [190], which includes around 51K papers. Each topic in TREC-COVID has three fields: a short keyword *query*, a *description*, and a *narrative*. In our experiments, we consider each field as a different formulation of the topic. We also include the *concatenation* of the keyword query and the description. Thus, the total number of queries we consider for TREC-COVID is equal to 120.

Regarding lexical and semantic models, we consider five different models for each category. The lexical models used are: TF-IDF [39]; BM25 [131]; Query Likelihood Model with Dirichlet Smoothing (QLM) [201]; Divergence From Randomness (DFR) [5]; and Divergence From Independence (DFI) [89]. All lexical models perform stopwords removal and stemming. As for semantic models, we adopt: a Word2Vec [110] based approach where query and document representations are built by summing up the IDF-weighted representation of the words contained in them [185, 103]; the Neural Vector Space Model (NVSM) [179]; and three variants of the Semantic-Aware neural Framework for IR (SAFIR) [1]. The three variants of SAFIR are $\mathrm{SAFIR}_{\mathrm{sp}}$, which integrates both polysemy and synonymy, $\mathrm{SAFIR}_{\mathrm{p}}$ which integrates polysemy but not synonymy, and $\mathrm{SAFIR}_{\mathrm{s}}$ which integrates synonymy but not polysemy. All semantic models have been trained for 10 epochs with parameters set as in [1].

We evaluate models using AP at cutoff 1000, obtaining an experimental GoP as defined in [58]. The performances of the retrieval models in terms of AP are reported in Table 7.1 for both OHSUMED and TREC-COVID collections.

Table 7.1 MAP of the models on OHSUMED and TREC-COVID collections. Models performance are comparable both within and across models categories.

Model	OHSUMED	TREC-COVID
	Lexical	
TF-IDF	0.524	0.362
BM25	0.620	0.488
QLM	0.577	0.434
DFR	0.641	0.496
DFI	0.592	0.467
	Semantic	
Word2Vec	0.568	0.482
NVSM	0.595	0.455
SAFIR$_s$	0.604	0.463
SAFIR$_p$	0.610	0.461
SAFIR$_{sp}$	0.612	0.466

7.2 Topic and Category Interaction

Several works have shown that queries strongly interact with retrieval models in determining their performance [12, 65]. This means that two models might have similar average performance on a set of queries but, when looked at the query-level, their performance might vary greatly. A similar consideration also applies to lexical and semantic models. Some queries are best suited to semantic models, while some others to lexical ones [106, 1]. We are thus interested in quantifying such an effect. In other words, we want to evaluate the interaction between queries and model categories.

To determine whether the models category – that is, lexical or semantic – has a significant effect on performance, we conduct an ANOVA on the runs obtained with the considered retrieval models. ANOVA is a well-known statistical technique that allows identifying statistically significant differences among experimental conditions. Several works in IR applied ANOVA to determine the effect of different factors on the overall performance of an IR system [12, 184, 65, 53]. ANOVA models the explained variable, which in our case is AP, as a linear combination of the effect of each factor in the experimental setup, plus an error component. The error term accounts for the variance in the data unexplained by the model.

In our analyses we first consider the following model:

$$y_{ijk} = \mu_{\cdots} + \tau_i + \gamma_j + \alpha_{k(j)} + \tau\gamma_{ij} + \varepsilon_{ijk}, \tag{MD7.1}$$

where y_{ijk} is the performance (measured using AP) observed on the i-th topic using the k-th model of the j-th class; $\mu_{\cdots}$ is the grand mean over all the data; τ_i is the effect of the i-th topic; γ_j is the effect of the j-th class; $\alpha_{k(j)}$ is the effect of the k-th model inside the j-th class; $\tau\gamma_{ij}$ is the interaction between the i-th topic and the j-th class and ε_{ijk} is the prediction error. Note that the *model* factor is nested inside the *category* one. In the above-mentioned ANOVA model, a IR model is meaningful only in relation to its category. In other words, since we cannot consider, for instance, BM25 inside the "semantic" category, nor we can consider NVSM in the "lexical" one, we define the model factor as nested inside the category, and thus each model contributes only to the variance of its category.

For each ANOVA, we report the Sum of Squares (SS), the Degrees of Freedom (DF), the Mean Squares (MS), the F-statistic (F), the p-value and the Strength of Association (SOA), using the ω^2 indicator. The SOA indicates the impact of each factor on the variability of the data. Typically, a factor with $0.01 \leq \omega^2 < 0.06$ is considered small-sized, while $0.06 \leq \omega^2 < 0.14$ indicates a medium-size effect, and $\omega^2 \geq 0.14$ a large-size effect. Table 7.2 reports the results of the ANOVA on OHSUMED using the above-mentioned GoP of runs.

Table 7.2 ANOVA summary table on runs for the OHSUMED collection. Observe the large interaction between the topic factor and category factor. ω^2 for not significant factors is ill-defined and thus not reported.

Source	SS	DF	MS	F	p-value	$\hat{\omega}^2_{\langle fact \rangle}$
Topic	19.740	62	0.318	79.831	$< 1e-4$	0.886
Category	0.007	1	0.007	1.805	0.1797	—
Model(Category)	0.584	8	0.073	18.306	$< 1e-4$	0.180
Topic*Category	1.583	62	0.026	6.403	$< 1e-4$	0.347
Error	1.978	496	0.004			
Total	23.892	629				

From the results in Table 7.2 we observe that the effect of the sole models category is not significant (p-value>0.05) – which means that lexical and semantic categories are not statistically significantly different. In other words, we cannot say that either lexical or semantic models perform best in absolute terms. Nevertheless, the interaction between topic and category is significant and the ω^2 value indicates a large effect. This means that the category significantly impacts on how good the results on a specific topic will be. Such a finding suggests that the semantic gap is an inherent property of the topics, less related to the specific retrieval models and more on their category. To further support this intuition, the interaction between the topic and the category is larger than the effect of the sole model. Thus, if we understand when a topic is lexical or semantic, we can achieve large performance improvements.

As for TREC-COVID, each topic is represented by four different formulations: the keyword *query*, the *description*, the *narrative* and the *concatenation* of query and description. Each formulation of a topic can only be used in relation to that topic and therefore the formulations have to be treated as a nested factor inside the topic. Therefore, we define a second ANOVA model, called MD7.2:

$$y_{iljk} = \mu_{\dots} + \tau_i + \phi_{l(i)} + \gamma_j + \alpha_{k(j)} + \tau\gamma_{ij} + \phi\gamma_{l(i)j} + \varepsilon_{ijlk}, \qquad \text{(MD7.2)}$$

which also includes $\phi_{l(i)}$, the effect of the l-th formulation, nested inside the i-th topic, and $\phi\gamma_{l(i)j}$, the interaction between the l-th formulation of the i-th topic with the j-th class. Table 7.3 summarizes the ANOVA results with MD7.2 on TREC-COVID.

From the results on TREC-COVID we observe that both the topic and its formulations have a large effect. The importance of the formulation factor indicates that, with an appropriate topic formulation, the performance on the topic can change greatly. Similar to

Table 7.3 ANOVA summary table on runs for the TREC-COVID collection. Observe the high $\hat{\omega}^2$ effect for the interaction topic*category that shows the importance of selecting the proper model category for each topic.

Source	SS	DF	MS	F	p-value	$\hat{\omega}^2_{\langle fact \rangle}$
Topic	24.100	29	0.831	301.291	$< 1e-4$	0.879
Query(Topic)	15.568	90	0.173	62.712	$< 1e-4$	0.822
Category	0.074	1	0.074	26.732	$< 1e-4$	0.021
Model(Category)	1.470	8	0.184	66.628	$< 1e-4$	0.304
Topic*Category	2.200	29	0.076	27.506	$< 1e-4$	0.390
Query(Topic)*Category	1.060	90	0.012	4.270	$< 1e-4$	0.197
Error	2.626	952	0.003			
Total	47.098	1199				

what we observed in Table 7.2, the interaction between the topic and the models category is large ($\omega^2 = 0.390$), larger than the effect of both the sole category and the model. Also the interaction between the topic formulation and the models category is large ($\omega^2 = 0.197$), although not as large as the one between topic and category. This suggests that the semantic gap relates more to the underlying information need than the different topic formulations.

Overall, we hypothesize that the relation between topics and model categories, highlighted by ANOVA, links to the semantic gap and the association of a topic with its relevant documents. For instance, if a topic has many relevant documents containing synonyms of the query terms, then a semantic model might be best suited to perform retrieval. In fact, in this case, most of the topic formulations will not contain all the possible query synonyms and will thus be affected by the semantic gap. Conversely, topics that can be easily represented by few keywords – likely to be found within relevant documents – will have less ambiguous formulations, which are best suited to lexical models.

7.3 Features Importance for the Semantic Gap

Section 7.2 showed the impact of choosing the proper models category depending on the query at hand. If we could classify queries as semantically hard or easy, we might also adopt an IR model from the right category. To properly train a classifier capable of doing that, we need *i)* to label queries as “semantic” or “lexical”, and *ii)* to find a set of features that correlate with such aspects of the queries. The next two paragraphs tackle the above-mentioned challenges.

Labeling queries The first aspect we address is the labeling of queries as "semantic" or "lexical". The absence of a rigorous definition of *semantically hard* or *easy* for a query prevents us from manually labeling queries as "semantic" or "lexical". In this regard, also the definition of "hard" topic is a debated aspect [43]. Therefore, we propose to label queries according to how the two models categories perform on them. To the best of our knowledge, this is the first automatic approach to address this problem.

To this end, we first compute the average performance of each model. Then, for each query, we perform the following three steps. Firstly, we compute for each model the relative improvement over its average performance. Secondly, we determine whether the relative improvement is, on average, greater for lexical or semantic models. Finally, we label the considered query as "semantic" if the improvement over the average model performance is greater for semantic models than for lexical ones; vice versa, we label the query as "lexical".

Note that we do not consider absolute performances to label queries, since even a poorly performing lexical method like TF-IDF (cfr. Table 7.1) might prove effective when the query is semantically easy. Thus, we focus on relative improvements, which provide more robust signals to performance outliers.

Let $\mathscr{S}$ be the set of models and $\mathscr{Q}$ the set of queries. We call $AP_s(q)$ the AP observed for the model s on the query q, and $\text{MAP}_s(\mathscr{Q})$ and $\text{std}_s(\mathscr{Q})$ respectively the MAP and the standard deviation of the AP observed for the model s over the queries $\mathscr{Q}$. We define $Z_{s,q} = \frac{AP_s(q) - \text{MAP}_s(\mathscr{Q})}{\text{std}_s(\mathscr{Q})}$ the relative improvement over the mean performance.

By standardizing relative improvements, we account for the variability in models performances. Then, let $\mathscr{S}_s$ be the set of semantic models, and $\mathscr{S}_l$ the set of lexical models.

Definition 7.3.1. A query q is labeled as "semantic" iff

$$\frac{\sum_{s \in \mathscr{S}_s} Z_{s,q}}{|\mathscr{S}_s|} >_\alpha \frac{\sum_{s \in \mathscr{S}_l} Z_{s,q}}{|\mathscr{S}_l|},$$

where $>_\alpha$, with $\alpha \in [0.5, 1)$, indicates that the mean relative improvement for semantic models is statistically significantly higher than that for lexical models at significance level α. Queries are labeled as "lexical" using the opposite ordering relation ($<_\alpha$).

Therefore, using the above-mentioned definition we can label queries as either "semantic" or "lexical" at a specific level of α. In practice, given a query q, we call $\mathscr{Z}_{q,sem} = \{Z_{s,q} \; \forall \; s \in \mathscr{S}_s\}$ the set of relative improvements of the semantic models for q, and $\mathscr{Z}_{q,lex} = \{Z_{s,q} \; \forall \; s \in \mathscr{S}_l\}$ the set of relative improvements of the lexical models for q. Using an unpaired t-test, we determine whether $\mathscr{Z}_{q,sem}$ has greater mean than $\mathscr{Z}_{q,lex}$. If so, then q is labeled as "semantic". On the other hand, if $\mathscr{Z}_{q,lex}$ has statistically significantly greater mean than $\mathscr{Z}_{q,sem}$, then q is labeled as "lexical". Otherwise, q is labeled as "neutral".

Table 7.4 OHSUMED queries classification.

Label	**Confidence**			
	$\alpha > 0.95$	$\alpha > 0.90$	$\alpha \leq 0.90$	**Total**
Semantic	13	3	10	26
Lexical	13	6	18	37
Both	26	9	28	63

Table 7.5 TREC-COVID queries classification.

Label	**Confidence**			
	$\alpha > 0.95$	$\alpha > 0.90$	$\alpha \leq 0.90$	**Total**
Semantic	27	7	26	60
Lexical	27	8	25	60
Both	54	15	51	120

Tables 7.4 and 7.5 report the statistics of our labeling approach for OHSUMED and TREC-COVID collections, respectively, at different levels of confidence.

We can observe that, in both collections, queries labeled with confidence above $\alpha = 0.90$ (*p-value* < 0.1) make up more than half of the total queries (i.e., 55.6% and 57.5% respectively). Another interesting observation is that queries labeled with high confidence split evenly between lexical and semantic categories. This confirms what we observed in Tables 7.2 and 7.3, where the effect of the sole category plays a marginal role on performance.

Focusing on TREC-COVID queries, we observe that different formulations of the same topic are either classified always in the same category or, when this is not the case, such formulations are labeled with low confidence[1]. This further explains the magnitude of the effects observed in Table 7.3, where the topic formulation showed a lower, although significant, interaction with the models category compared to that of the topic. The only exceptions are topics 16 and 23, where the *narrative* formulation is lexical while *concatenation* and *query*, for topic 16, and *concatenation*, *description*, and *query*, for topic 23, are semantic with confidence > 0.95. In this regard, it is interesting to note that, for both topics, the formulation labeled as “lexical” is always the *narrative* one. We attribute the reason for this to the richer linguistic structure of the *narrative* formulation, which, in both topics, presents a better description, as well as several relevant concepts, of the underlying information need – thus limiting the semantic gap and reducing the need for semantic models.

[1] we omit these statistics, due to space reasons

In the following, we restrict to queries labeled with confidence above 0.90, as we want to focus on queries that have been labeled with a high degree of confidence. Moreover, queries labeled as “neutral” for $\alpha = 0.90$ have been discarded.

Features and Category Selection To address the second aspect of **RQ 7.2** – that is, classifying a query as “semantic” or “lexical” – we explore two different sets of pre-retrieval features: Lexical- and Semantic-oriented features. Lexical-oriented features are based on query and corpus statistics and depend on the distribution of terms within the collection. Regarding semantic-oriented features, we first perform semantic indexing on OHSUMED and TREC-COVID collections as in [1]. Then, we adopt features similar to those proposed by Mothe and Tanguy [116], but, instead of considering only query-based features, we take into account both query- and corpus-based features. The considered features are reported and described in Table 7.6.

We employ three well-known classification models to understand the effectiveness of the considered pre-retrieval features when used to classify queries into lexical and semantic categories. The adopted models are: Decision Tree (`DTr`), Support Vector Machine (`SVM`), and Multi-Layer Perceptron (`MLP`). To perform experiments, we label queries using the process described above and we restrict to “semantic” and “lexical” queries that present a significance score greater than 0.90. For each classifier, we perform grid search with cross-validation to obtain the best hyper-parameters. We adopt 5-fold cross-validation for TREC-COVID, whereas we use 3-fold cross-validation for OHSUMED to avoid obtaining single-class folds due to the low number of samples. The results of the different classifiers are reported in Table 7.7, where we report mean and standard deviation over the different folds. To determine results significance (marked as †), we apply a randomization test with Bonferroni correction for multiple comparisons [156].

Regarding OHSUMED, we first highlight that `MLP` is the best performing method. However, `MLP` is also the method with the largest standard deviation for F1. This is likely due to the small number of samples – i.e., 35 queries labeled with confidence above 0.90. On top of this, none of the considered methods perform statistically better than the random classifier. Conversely, results for TREC-COVID are more stable – highlighting the impact the number of samples has on the stability of the classifiers performance. Also in TREC-COVID, both `SVM` and `MLP` are not statistically better than the random classifier. On the other hand, however, `DTr` obtains preliminary yet promising performance (i.e., 67% for accuracy and 66% for F1) and it is significantly better than the random classifier for both measures. This suggests the presence of underlying patterns within data and the potential of the considered features to distinguish between semantically hard (“semantic”) and easy (“lexical”) queries.

Table 7.6 Pre-retrieval features considered for the category selection task.

Name	**Description**
	Lexical-oriented features
QL	Number of terms in the query [116]
{std,mean,max}IDF	Features based on the distribution of the IDF over the query terms [76]
{sum,mean,max}SCQ	Features based on the similarity between corpus [204]
QDF	Number of documents containing at least one query term
	Semantic-oriented features
QPD	Number of polysemous words within the query
{sum,std,max}NCQT	Sum, standard deviation, and max over the number of concepts related to query terms
{sum,std}NCPQT	Sum and standard deviation over the number of concepts related to polysemous query terms only
QSD	Number of synonymous words within the query
{sum,std,max}NSEQC	Sum, standard deviation, and max over the number of different synset elements related to query concepts
{sum,std}NSQC	Sum and standard deviation over the number of different synonyms related to query concepts
SDF	Number of documents containing at least one synonym of a query term
WSDF	Number of documents containing at least one query term and no synonyms of the query terms
WTDF	Number of documents containing at least one query synonym and no query terms

Table 7.7 Classifiers performance. We report mean and standard deviation over 3- and 5-folds for OHSUMED and TREC-COVID, respectively. † indicates statistical significance over the random classifier, according to a permutation test with significance 0.95 and Bonferroni correction.

	OHSUMED		**TREC-COVID**	
	Accuracy	**F1**	**Accuracy**	**F1**
DTr	0.626 (0.089)	0.586 (0.057)	**0.668 (0.093)**†	**0.659 (0.141)**†
SVM	0.687 (0.074)	0.611 (0.079)	0.623 (0.053)	0.610 (0.136)
MLP	**0.740 (0.081)**	**0.675 (0.146)**	0.628 (0.217)	0.590 (0.269)

Relying on the results of the decision tree, we further investigate the features importance to determine which features correlate the most with the semantic gap, causing the query to be either semantically easy or hard. We only consider the decision tree built for TREC-COVID, since results on OHSUMED are not statistically significant. The first two features by importance are `QDF` (number of documents containing at least one query term) and `WSDF` (number of documents containing only query terms and no synonyms). Their importance is, respectively, 17.6% and 16.7%. These features are both related to the distribution of the query terms in the collection. For this reason, they are likely used by the classifier to identify semantically easy queries. Indeed, a large number of documents containing query terms is a potential indicator for the performance of lexical models. Besides, the fact that `WSDF` is the second most important feature is a further evidence of this: if several documents contain query terms, but only few of them present also synonyms of such terms, then the semantic gap will likely be small and lexical models will be effective. The third feature by importance is `meanSCQ` (12.1%): a pre-retrieval score based on IDF. A query having a high `meanSCQ` score indicates that lexical models are likely to perform well. This is due to the fact that most of the lexical approaches rely on heuristics based on IDF. Note also that `SCQ` is considered a “low performing” feature for predicting queries performance [55]. Nevertheless, in our scenario, it gains relevance in determining which models category performs best for the query. The fourth feature is `stdNCPQT` (the standard deviation over the number of concepts for each polysemous word in the query). This feature has importance 10.1%, which indicates the relevance of polysemy in determining the models category: having (several) query words with different concepts associated makes the query ambiguous and semantic models best suited to address it.The two subsequent features are `sumNSEQC` (8.8%) and `maxNSEQC` (7.3%). They represent, respectively, the sum and the maximum of the number of synset elements related to the query concepts. Both features are related to synonymy, which is another relevant aspect that identifies the presence of the semantic gap between queries and documents. Similarly to our intuition about polysemy, having query words with several synonyms suggests that semantic models are best suited to retrieve relevant documents.

Other features with decreasing, but significant, importance are `SDF` (5.7%) and `sumNCPQT` (5.2%).

As for the remaining features, they are negligible according to the classifier.

Thus, even though the results are preliminary and indicate there is large room for improvement, they still highlight that the considered lexical- and semantic-oriented features relate with models categories. Therefore, they can be used as a starting point to investigate the presence of the semantic gap within test collections and to build better approaches for category selection.

7.4 Final Remarks

In this chapter we switched the focus from the query performance prediction to the system performance prediction. In particular, we started from the assumption that some queries perform better if used with traditional lexical approaches, while others achieve better results if used with semantic (neural) models. In this sense, our work aimed at identifying what characteristics of a query make it synergize better with either semantic or lexical models, or, in other terms, what factors make a query more challenging for lexical models. We first developed a classification strategy to divide queries into "prominently lexical" and "prominently semantic" queries. In particular, our approach, based on ANOVA, consisted of labelling as lexical (semantic) those queries for which all the lexical (semantic) models had a statistically significant advantage in terms of performance over the other category of models. We observed that queries roughly divide evenly between semantic and lexical ones. We, therefore, identified a series of features that we expected to correlate with the "semantic complexity" of the query (e.g., the number of synonyms for the query terms or the number of concepts represented using the query words). Using such features, we trained a classifier based on these features to predict which class of models is the best performing one. Even though preliminary, our findings indicate that the identified features correlate with the best performing category of systems.

Chapter 8

Improving QPP Evaluation: sMARE

8.1 Introduction

The IR community has long recognized the importance of applying statistical tests to evaluation results. Although best practices continue to evolve, conference and journal guidelines, and discussion papers including those of [66] and [144] have led the community to appreciate the importance of a more theoretically grounded evaluation. Practitioners in IR have been urged over the years to include sound analyses using statistical tests of significance or confidence intervals in submitted manuscripts. While this has led to higher quality analytical comparisons in many IR-related fields, not all areas have adopted the practice. An example of a common IR problem that might benefit from alternative evaluation techniques is Query Performance Prediction (QPP).

The goal of QPP is to estimate the effectiveness of a retrieval system in response to a query when no relevance judgments are available [28]. The most widely-used method for evaluating QPP approaches is based on the strength of a relationship between per-topic prediction scores, and the actual per-topic system effectiveness as measured using a standard IR effectiveness metric, usually AP. The association is measured using a correlation coefficient, with different papers reporting the Pearson (linear) correlation, Spearman's rank correlation, or Kendall's τ. A QPP approach that achieves a higher correlation value than another is taken to be the superior approach. This evaluation method compares QPP effectiveness at a very high level, with the performance of a QPP approach over a whole set of topics being summarized by a single correlation coefficient as a *point value*.

In order to statistically validate the results two alternatives are available. First, we can test whether or not the correlation between a predictor and the retrieval results is significantly different from zero [76, 205, 41, 206, 35, 49, 204, 29, 44, 73, 116, 159]. However, this validation approach just tells us how reliable the conclusions are for a single QPP method, and does not allow two or more QPP approaches to be directly compared. Second, by relying on repeated randomized topic sampling, we can test whether or not the correlation coefficients for two different QPP methods are significantly different from each other. A statistically appropriate method to test the latter would rely on Fisher's z transformation of sample correlation coefficients. In fact, this approach was previously suggested by Hauff et al. [72] and again more recently by Roitman [139] to more reliably test for significant differences in QPP model performance. However, this practice has not been adopted in published QPP work to date. Instead, a Student's t-test for the difference of means of the correlated correlation coefficients is currently the preferred approach [137, 197, 200]. However, it is important to note that both of these approaches are fundamentally different from the pair-wise significance test used for system retrieval effectiveness, which is now common practice in IR evaluation exercises.

Motivated by these observations, we re-examine how QPP effectiveness can be analyzed using a more fine-grained approach – by modeling the performance of QPP techniques as *distributions*. This approach has also previously been applied successfully in system evaluation exercises. A distribution-based model can be constructed as follows. First, an estimate of the performance for each system-topic combination is computed using a traditional performance measure, such as AP. Then, all of the topics for a collection are used to model the performance distribution. Note that this is fundamentally different from a classical QPP evaluation approach. Indeed, even when various sampling techniques (e.g., randomization or bootstrap) are currently used in QPP, this is a re-sampling of topics, and leads to a new (aggregated) *point estimate*, e.g., Kendall's τ, for that sample. The different re-samples are then used to compute an expectation and a confidence interval for the point estimate. In contrast, when randomization/bootstrap techniques are used for the evaluation of retrieval effectiveness [165], it is topics that are re-sampled; for *each* topic a performance score such as AP is computed, and a *distribution* of performance for that sample is obtained. A summary of this distribution, e.g., a mean or a confidence interval, is then computed, and finally the different re-samples are used to compute a further expectation and confidence interval for the summary. We propose a methodology similar to the latter approach.

Our evaluation approach has several appealing properties: it allows formal inferential statistics to be applied, which generalizes the results to the entire population of topics; it allows the behavior of a QPP approach to be more clearly isolated, for example through confidence intervals; and, it enables factor decomposition, which in turn allows us to measure the relative contributions to observed effectiveness systematically. In particular, we compare the performance with the distance between the rank predicted by a QPP model for a query and the rank of the query using a given traditional performance measure. Being a measure of the rank error made by a predictor, we call the above measure scaled Absolute Ranking Error (sARE). So, we now have a measure of error for each of the topics, given a specific predictor. To have a measure of the overall quality of the predictor, we can average sARE over all topics and compute the scaled Mean Absolute Ranking Error (sMARE). We also incorporate recent work in retrieval effectiveness on query variation and reformulation for each topic [8, 9, 19, 170, 200] into our framework, which allows a finer-grained sampling of retrieval performance, and allows us to estimate interaction between systems, topics and query formulations, which was not possible using only single pointwise estimates.

Concerning the evaluation of QPP approaches, our work focuses on two closely related research questions:

- **RQ 8.1**: How can detailed statistical analysis and testing be applied to QPP evaluation exercises?
- **RQ 8.2**: What factors contribute to improving or reducing the performance of a QPP model?

The chapter is organized as follows. In Section 8.2 we describe the experimental setup. Section 8.3 contains details on the traditional evaluation of QPP models, used as a baseline for the subsequent analyses. Section 8.5 contains the analysis of how the framework behaves when using several approaches to compute the error and to break ties. In Section 8.6 we describe how to use the sARE measure to break down the performance of multiple QPP models. Finally, in Section 8.7, we include observations that can now be made on QPP models and query formulations when performing an evaluation using ANOVA and the sARE measure.

8.2 Experimental Setup

In our analyses concerning sMARE, we use the TREC 13 Robust 04 Ad Hoc [180] collection. The Robust 04 ad hoc track consists of approximately 528K documents from TREC disks 4 & 5, minus the Congressional Record from the TIPSTER corpus, and contains 249 topics with at least one relevant document in the original TREC relevance judgments. We enrich the set of queries for the corpus using publicly available human-curated query reformulations for each topic [15].[1] Our experiments use a Grid of Points (GoP) of runs as described by Ferro and Harman [58], using 4 different stopword lists (`atire`, `zettair`, `indri`, `lingpipe`), plus the `no stop` (not applying stopword removal) approach and 2 different stemmers, (`lovins`,

[1] http://culpepper.io/publications/robust-uqv.txt.gz

porter) plus a nostem approach. The indexes are constructed from the raw postings lists created with the Apache Lucene search engine[2], and the Common Index File Format (CIFF) [100]. All runs were produced using our own implementation of the query-likelihood model and use Dirichlet smoothing ($\mu = 1000$), as described originally by [202]. Each run was repeated 15 times. We test 16 QPP models ($12+4$ UEF-based methods) in our analyses, all of which are summarized in Table 2.3. Our goal was to choose representative and well known system configurations and QPP models, and the evaluation framework is not limited to any specific configuration. It can easily be extended by others for further experiments in the future. In total, 240 different predictor-system combinations were generated for the Robust 04 collection. The pre-retrieval approaches are parameter-free and do not require tuning. For the parameters of the post-retrieval predictors we used fixed settings that have been demonstrated to be effective for the Robust 04 collection previously [162, 159, 169]. We apply Average Precision (AP) to measure the effectiveness of the different retrieval pipelines, as our primary goal is to be consistent with previous evaluation exercises, as AP was the most common effectiveness metric used in prior QPP work.

8.3 Traditional QPP Evaluation Using Correlations

Prior work on QPP has relied primarily on a single evaluation paradigm. Given a set of topics (information needs), where each topic is represented by a single query, a single retrieval method, and a single document corpus, the prediction quality of the predictors is evaluated as follows:

1. Retrieval effectiveness of the queries is measured with a common IR metric, usually AP or possibly nDCG, to induce a ranking of the queries. This ordering serves as the ground truth in the evaluation process.

2. The QPP method is applied to the queries, which generates a candidate list where the queries are ranked by their prediction values.

3. A correlation coefficient is computed between the ground truth list and the candidate list produced by the predictor.

4. The correlation coefficients of different predictors are then compared, with an underlying assumption that a higher correlation value attests to the superior quality of a predictor.

[2] https://lucene.apache.org

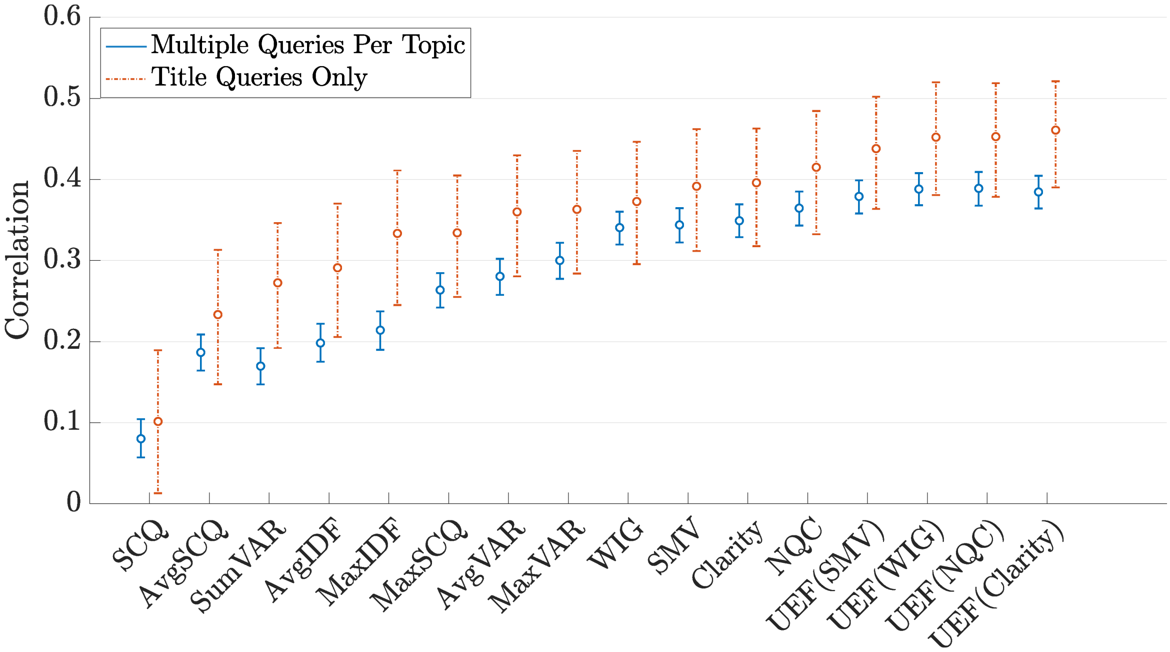


Fig. 8.1 Prediction quality of the selected QPP models on Robust 04 (Confidence Intervals computed with Kendall's τ), using either title queries or all available formulations.

The correlation coefficient is usually reported as Pearson's r for linear correlation, Kendall's τ, or Spearman's ρ for the monotonic rank correlation.

Figure 8.1 shows the performance of 16 different QPP models when using this common evaluation approach – Kendall's τ correlation in this case – with 95% confidence intervals shown as well. In this example, the results are generated for a specific retrieval pipeline, using the `indri` stoplist and `porter` stemmer. To compute the 95% confidence intervals, we used a bias-corrected and accelerated bootstrap procedure with 10,000 samples. Observe that when using title queries only (orange bars), there is a large degree of overlap between the different QPP approaches. Similar results were observed when using all of the other pipelines described in this work. Conducting pairwise comparisons on the data from Figure 8.1 (title queries only), a bootstrap hypothesis testing [51] shows that 57 pairs of predictors are statistically significantly different at significance level $\alpha = 0.05$, out of 120 total pairs of QPP models (47.5%). In particular, among the best performing predictors, UEF(Clarity) is not statistically different from UEF(WIG), UEF(NQC), UEF(SMV), Clarity and NQC. A large number of statistical "ties" between different QPP models may be caused by one of the following two reasons: *i)* methods are in fact equal and there has been little to no improvement since Clarity was proposed by Cronen-Townsend et al. [41]; or *ii)* our current evaluation strategy is not powerful enough to measure any difference between the models. We are more inclined to believe our second hypothesis, which is inline with the observations of Hauff et al. [72]. That is, using confidence intervals can make it difficult to conclusively determine which QPP system is the best performing one. Figure 8.2 shows a heat-map plot of the pairwise ranking similarities between the different QPP methods. The similarity is measured with Kendall's τ correlation [87]. Given two sorted lists of real values, the original Kendall's τ [86][3] is defined as follows:

$$\tau = \frac{\text{number of concordant pairs} - \text{number of discordant pairs}}{\text{total number of pairs}} \tag{8.1}$$

Given the formulation of Kendall's τ as defined in 8.1, if we define C and D as:

$$\begin{aligned} C &= \frac{\text{number of concordant pairs}}{\text{total number of pairs}}, \\ D &= \frac{\text{number of discordant pairs}}{\text{total number of pairs}}; \end{aligned}$$

The general Kendall's τ formula (as defined in eq. 8.1) becomes:

$$\tau = \frac{\text{number of concordant pairs}}{\text{total number of pairs}} - \frac{\text{number of discordant pairs}}{\text{total number of pairs}}$$

And therefore:

$$\begin{aligned} \tau &= C - D, \\ C + D &= 1; \end{aligned}$$

We can observe that:

$$\begin{aligned} C &= \tau + D, \\ D &= 1 - C; \end{aligned}$$

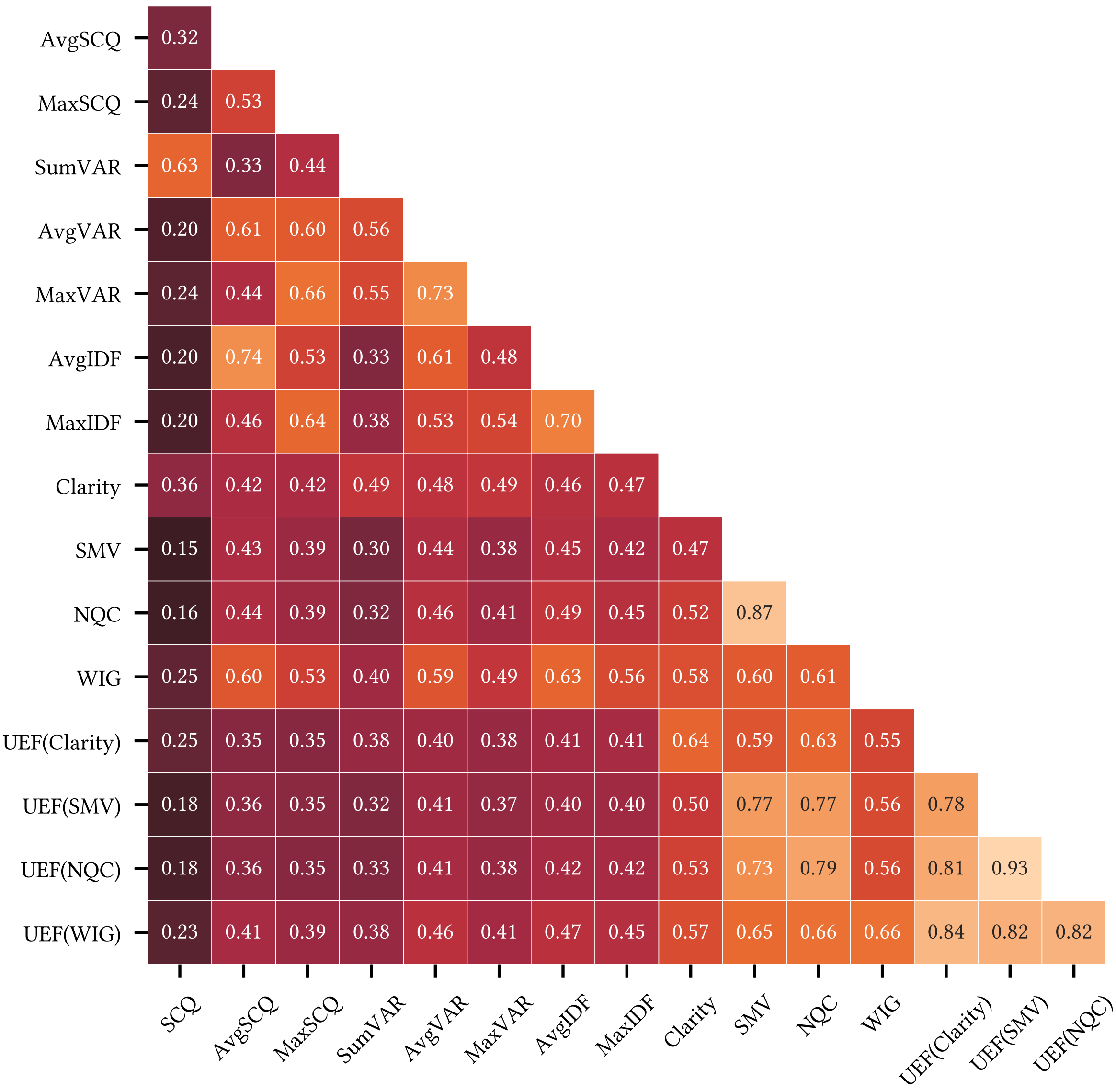
Thus $C = \tau + (1 - C) = \tau + 1 - C$. Therefore $C = \frac{\tau+1}{2}$ C can ve interpreted as an intuitive approximation of the ratio of agreement. For example, for $\tau = 0.6$, $C = 0.8$; means that 80% of the topic pairs are ranked identically using either pair of predictors.

Note that this is the original version of Kendall's τ [86], the actual formula applied in the correlation calculations throughout this manuscript is a later version, which is commonly known as τ_b (τ_s in the original paper) [87]. The correlation coefficient τ_b is extending the

[3] The original formula has no adjustments for ties in the rankings, it is mentioned here for its simplicity.

original formula to treat ties. Note there are later formulations of Kendall's τ which *do* account for ties.

Figure 8.2 further supports this result as all of the UEF based predictors show no significant differences from each other in the current setup. However, the noticeable drop in the similarity of the NQC and Clarity methods when compared to UEF(Clarity) suggests that a more powerful statistical analysis may yield a different outcome. This is a key motivation for our work and will be examined in greater detail.



	SCQ	AvgSCQ	MaxSCQ	SumVAR	AvgVAR	MaxVAR	AvgIDF	MaxIDF	Clarity	SMV	NQC	WIG	UEF(Clarity)	UEF(SMV)	UEF(NQC)
AvgSCQ	0.32														
MaxSCQ	0.24	0.53													
SumVAR	0.63	0.33	0.44												
AvgVAR	0.20	0.61	0.60	0.56											
MaxVAR	0.24	0.44	0.66	0.55	0.73										
AvgIDF	0.20	0.74	0.53	0.33	0.61	0.48									
MaxIDF	0.20	0.46	0.64	0.38	0.53	0.54	0.70								
Clarity	0.36	0.42	0.42	0.49	0.48	0.49	0.46	0.47							
SMV	0.15	0.43	0.39	0.30	0.44	0.38	0.45	0.42	0.47						
NQC	0.16	0.44	0.39	0.32	0.46	0.41	0.49	0.45	0.52	0.87					
WIG	0.25	0.60	0.53	0.40	0.59	0.49	0.63	0.56	0.58	0.60	0.61				
UEF(Clarity)	0.25	0.35	0.35	0.38	0.40	0.38	0.41	0.41	0.64	0.59	0.63	0.55			
UEF(SMV)	0.18	0.36	0.35	0.32	0.41	0.37	0.40	0.40	0.50	0.77	0.77	0.56	0.78		
UEF(NQC)	0.18	0.36	0.35	0.33	0.41	0.38	0.42	0.42	0.53	0.73	0.79	0.56	0.81	0.93	
UEF(WIG)	0.23	0.41	0.39	0.38	0.46	0.41	0.47	0.45	0.57	0.65	0.66	0.66	0.84	0.82	0.82

Fig. 8.2 The Kendall's τ correlation coefficient computed between several different QPP predictors. The correlation is calculated over topics, which are represented by TREC title queries on Robust 04 *indri-porter* pipeline.

In addition to using the traditional title queries, following what done in the previous chapter, we also explore the scenario of using multiple query formulations for a topic, which allows us to produce replicas for the same experimental conditions (i.e., the retrieval system or the QPP model used) on the same subject (i.e., the topic). While the correlation is generally lower when using multiple topic formulations (the blue bars shown in Figure 8.1), there is a high degree of similarity between the ordering of the QPP models for multiple query formulations to the ordering for title-only (Kendall's tau correlation between using title-only versus multiple queries per topic is 0.98, $p < 0.0001$). Notice that, to prevent the number of formulations for each topic from influencing the result, we randomly sample each topic using 5 different formulations. Overall, the statistically induced bootstrap intervals are substantially larger if a traditional title-only evaluation approach is used, which makes it less suitable for determining if any single system is a clear winner, while using multiple queries does induce smaller intervals and better discriminative power between the QPP approaches. Even if, as shown, using query variants does not dramatically impact the ranking of QPP models, it is nevertheless important to consider whether adding variants has an impact on the distribution of the raw AP scores. The MAP values are 0.211 and 0.254 for the set of all query formulations and title queries only, respectively, and thus are quite consistent. Figure 8.3 shows the PDF for the AP scores for the two scenarios – title-only (red line) and multiple queries per topic (blue line). The KLD, a measure of the distance the two distributions, is 0.039, which suggests there is a high similarity between the two distributions. In summary, the distributions are similar and thus the introduction of the multiple query formulations does not appear to skew the overall AP score distribution.

8.4 ANOVA Modeling and Analysis of QPP

To support a more detailed analysis of QPP methods and associated factors, we now explore the use of ANOVA, which can be achieved by modifying steps 3 and 4 of the traditional QPP evaluation process shown above. Instead of computing the correlations between the complete lists, we measure the difference, for each query, in the rank position assigned by a QPP method and the ground truth rank position assigned by AP. Ties in ranks are broken using the average of the ranks span, as is the default in many statistical applications [67]. Since the choice of tie breaking rule could have an impact on the results, several possible approaches are evaluated and discussed in greater detail in Subsection 8.5. Observe that this approach transitions us from *point estimates* of a single correlation value for the two lists over a whole set of topics to a *distribution* of the rank differences between the two lists for each query in the set. In order to scale the scores to the range $[0, 1]$ we divide them by the number of

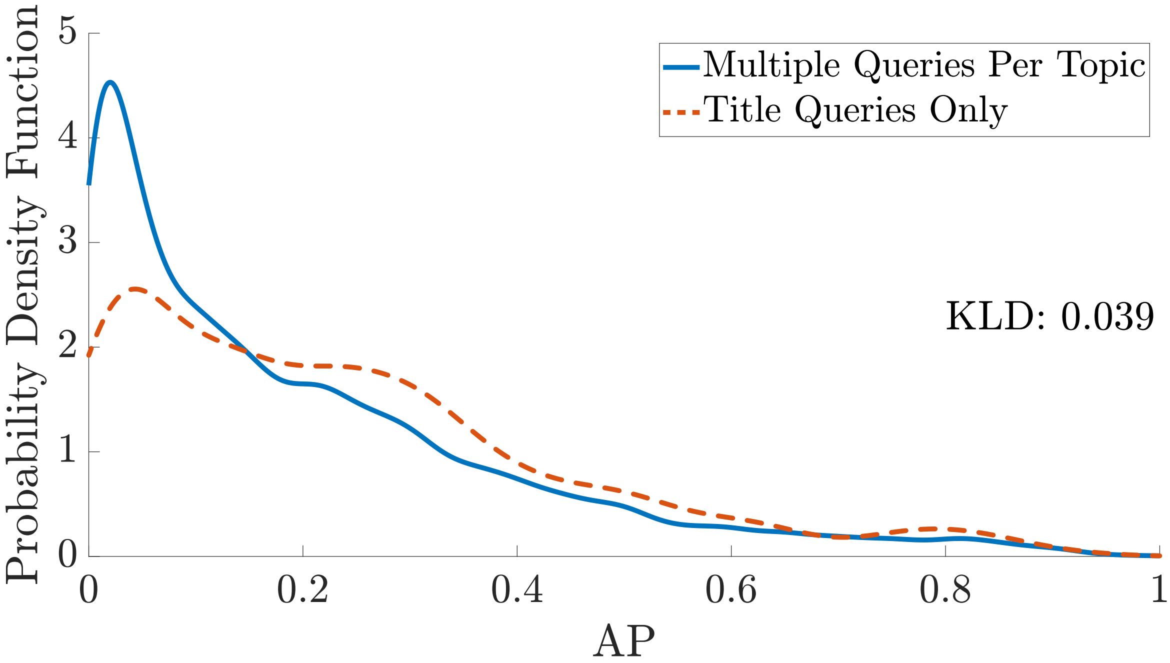


Fig. 8.3 A comparison of the AP score distributions of the title-only queries and multi-query topic formulations.

samples. The error, labeled as AP induced scaled Absolute Rank Error (sARE-AP) , for each query is:

$$\text{sARE-AP}(q_i) := \frac{|r_i^p - r_i^e|}{|Q|}, \tag{8.2}$$

where r_i^p and r_i^e are the ranks assigned by the predictor and the evaluation metric respectively for query i; Q is the set of queries. If we need the single point estimate of the prediction quality for each predictor $\mathscr{P}$, we can calculate the sMARE-AP as follows:

$$\text{sMARE-AP}(\mathscr{P}) := \frac{1}{|Q|} \sum_{q_i \in Q} \text{sARE-AP}(q_i). \tag{8.3}$$

Note that sMARE-AP can be seen as a derivation of *Spearman's Footrule distance*, making it a distance metric for the full rankings instead of a correlation. Among the properties of Spearman's Footrule distance, Diaconis and Graham [48] list that it is bounded between [0, $\lfloor 0.5n^2 \rfloor$], where n is the length of the ranking. Since both sARE-AP and sMARE-AP are normalized by the number of queries, sMARE-AP is bounded between [0, 0.5].

To demonstrate the agreement between the proposed evaluation method with existing evaluation practices from a high-level (point estimate) perspective, we use the QPP methods over the Robust 04 title queries. Figure 8.4 plots the ranking of the predictors, based on the median of the point estimates for each predictor for all 15 system configurations

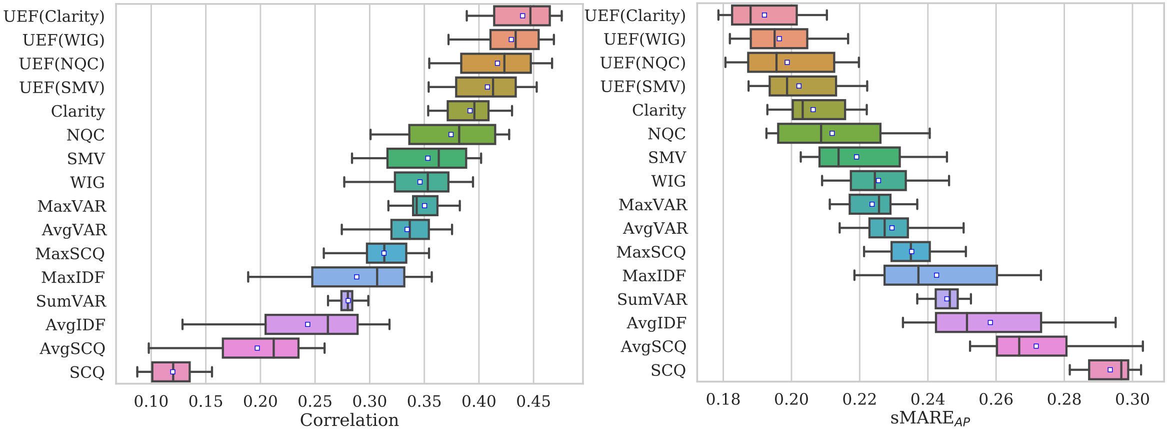


Fig. 8.4 Prediction quality when measuring correlation with Kendall's τ and sMARE-AP for Robust 04 title-only queries and 15 different system configurations. The line inside the interquartile range (IQR) is the median, and the white square is the mean.

(which is simply the median of the Kendall's τ correlation for the traditional evaluation approach), and the median of sMARE-AP for our evaluation approach. Each predictor consists of 15 values that represent the prediction quality. Though the directionality of the two approaches is inverted, the ranking of the predictors clearly agrees on the overall rank ordering. The corresponding box-plots also demonstrate the similarity of the variance estimate. In order to validate the agreement we computed the Pearson's correlation coefficient over the point estimates for the predictors for each of the 15 system configurations. The resulting correlations coefficients were all -0.99 or higher ($p < 0.0001$ for each).

8.5 Computing the Measure of Deviation of an Optimal Rank Ordering

When defining a measure to accurately represent the distance between the rank of a query w.r.t the rank of all queries when sorted in decreasing order by their AP score and their associated QPP score, two choices need to be made: the tie-breaking strategy and the approach used to quantify the deviation from the optimal rank.

8.5.1 Tie-breaking strategies

The sMARE framework is based on computing differences between the expected and observed ranks. The expected rank corresponds to the rank that the query achieves if we sort them by performance. The observed rank, on the other hand, is the rank assigned considering the prediction of a given QPP approach. Since we are considering rankings induced by scores for either observed or predicted performance, we can expect that two or more queries will obtain the same observed / predicted scores. In such cases, we must decide how to assign the value of the rank for each of the queries. Let $\mathscr{Q}_t$ be a set of queries that includes either identical QPP or AP scores, s. Given also r_k the rank of the query with the maximal score such that $s_k < s$, we can define the following tie-breaking strategies, using the list $(0.1, 0.2, 0.2, 0.3)$ as an example:

- `average` $(1, 2.5, 2.5, 4)$: the rank for all the queries in $\mathscr{Q}_t$ is the average rank in the set, equal to $(2r_k + |\mathscr{Q}_t| + 1)/2$. The main advantage of this method is that the sum of the ranks equals to the sum of the ranks when no ties exist.

- `min` $(1, 2, 2, 4)$: all the tied queries have the lowest rank in the tie set, equal to $r_k + 1$.

- `max` $(1, 3, 3, 4)$: all the tied queries have the highest rank in the tie set, equal to $r_k + |Q_t|$.

- `first` $(1, 2, 3, 4)$: ties are sorted "alphabetically" or "lexicographically", according to the order of appearance in the ranked list: where all possible values between 1 and $|\mathscr{Q}|$ are associated to a query. Note that this is similar in spirit to tie-breaking in the `trec_eval` tool which breaks ties by sorting on the document ID. However here we are sorting by query score and not scoring ranked documents.

- `dense` $(1, 2, 2, 3)$: similar to the `min` approach, the rank of all the queries in the set of ties will always be $r_k + 1$, but the rank between groups will always increase by 1. This means that, given $n \leq |\mathscr{Q}|$ unique scores associated to queries in a ranked list, every possible value between 1 to n will be assigned to at least one query.

To further highlight the importance of the analysis on the number of ties, we also report in Figure 8.5 the number of ties observed. The blue line shows the mean number of ties over all 13 QPP models, and the shaded area represents the 95% confidence interval. Note that, even if we consider as many as 6 digits, we still have on average more than 500 ties. Note that we have used 6 significant digits in the subsequent experiments for each raw observation, and more than is common practice, and only because it reduces the number of observed ties to a more conservative level – making them less likely to influence any of the observations being made.

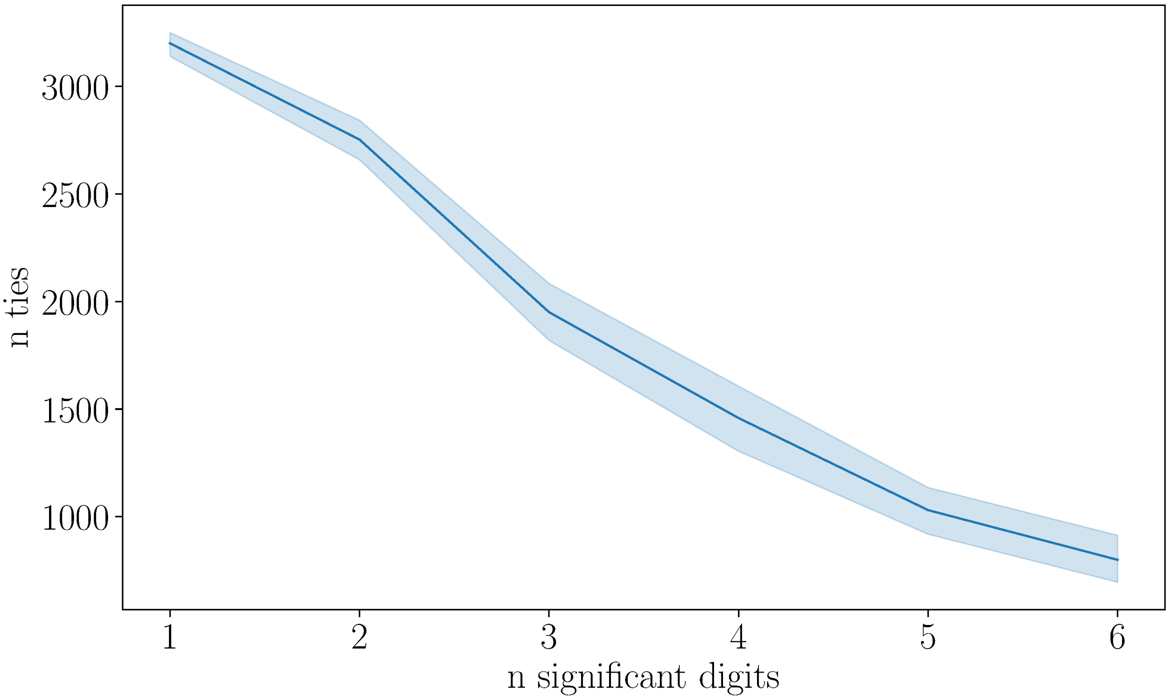


Fig. 8.5 The average number of ties observed between QPP methods, when the number of significant digits differs. Note that, even when using 6 significant digits, we can observe more than 500 ties on average.

8.5.2 Error measures

Given r_i^p, the rank observed for the query i in the ranked list sorted by QPP score, and r_i^e, the rank observed in the ranked list sorted by AP, four possible measures can be defined to quantify the distance from the optimal rank:

- sARE , as defined in Equation 8.2;
- sRE (scaled Rank Error) which uses the signed distance between the two ranks, scaled by the number of queries, and is defined as

$$\frac{r_i^p - r_i^e}{|Q|}.$$

For the case of no ties, or using the `first` or `average` rank strategy for ties, the sum over all queries would be 0, as it is equal to

$$\sum_Q \frac{r_i^p - r_i^e}{|Q|} = \frac{1}{|Q|} \sum_Q r_i^p - \sum_Q r_i^e.$$

This approach is not particularly useful for our needs, but may be useful for other studies.

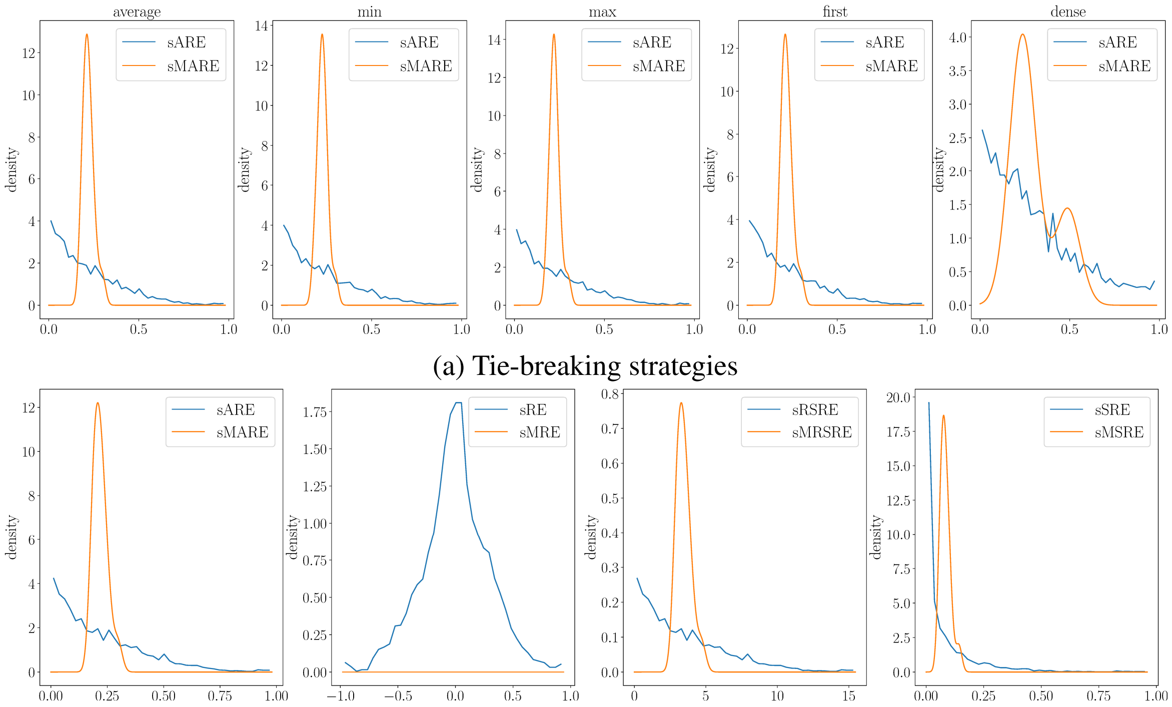
- sSRE (scaled Square Rank Error) is the square of the difference between the two ranks, normalized by the number of queries and is defined as

$$\left(\frac{r_i^p - r_i^p}{|Q|}\right)^2 .$$

- sRSRE (scaled Root Square Rank Error) the root of the squared difference:

$$\sqrt{\frac{(r_i^p - r_i^e)^2}{|Q|}}.$$

As shown in Equation 8.3, each of these measures can be aggregated by computing the mean of all queries for each predictor, to obtain a "mean" version.



(a) Tie-breaking strategies

(b) Ranking error computations

Fig. 8.6 The top panel (a) shows a comparison between multiple tie-breaking strategies (average, min, max, first and dense approaches, respectively) for both sARE and sMARE. The bottom panel (b) shows the different aggregation algorithms (s(M)ARE, s(M)RE, s(M)RSRE, S(M)SRE, respectively) using average tie-breaking, in term of score density distributions.

Figure 8.6 compares all of the tie-breaking strategies and formulations for the ranking error of the distribution of the scores for using one possible retrieval pipeline (`indri` stoplist and `porter` stemmer). Figure 8.6a shows the tie-breaking comparison. Note that, we have artificially inflated the number of ties by truncating the AP and QPP scores to 2 decimal points. Using higher precision scores, the tie-breaking strategies used are all nearly identical, due to proportionally fewer ties. For our tie-breaking strategy comparison, we show only the results observed using sARE – and its averaged version, sMARE – as deviation measure. All other measures discussed exhibit similar behavior. Figure 8.6b shows the comparison between the different approaches of computing the deviation of the QPP prediction from the ideal rank. For this comparison, ties-breaking uses the `average` strategy since our earlier experiment shows no appreciable differences between tie-breaking strategy when using our experimental data. For each possible setting, we compute the measure of interest for each topic-predictor pair, and plot the probability density distribution of such scores (blue lines). Furthermore, we compute the mean of the scores over all topics for each predictor (orange line). This statistical measure will be used later in our ANOVA experiments when we compare the QPP predictors.

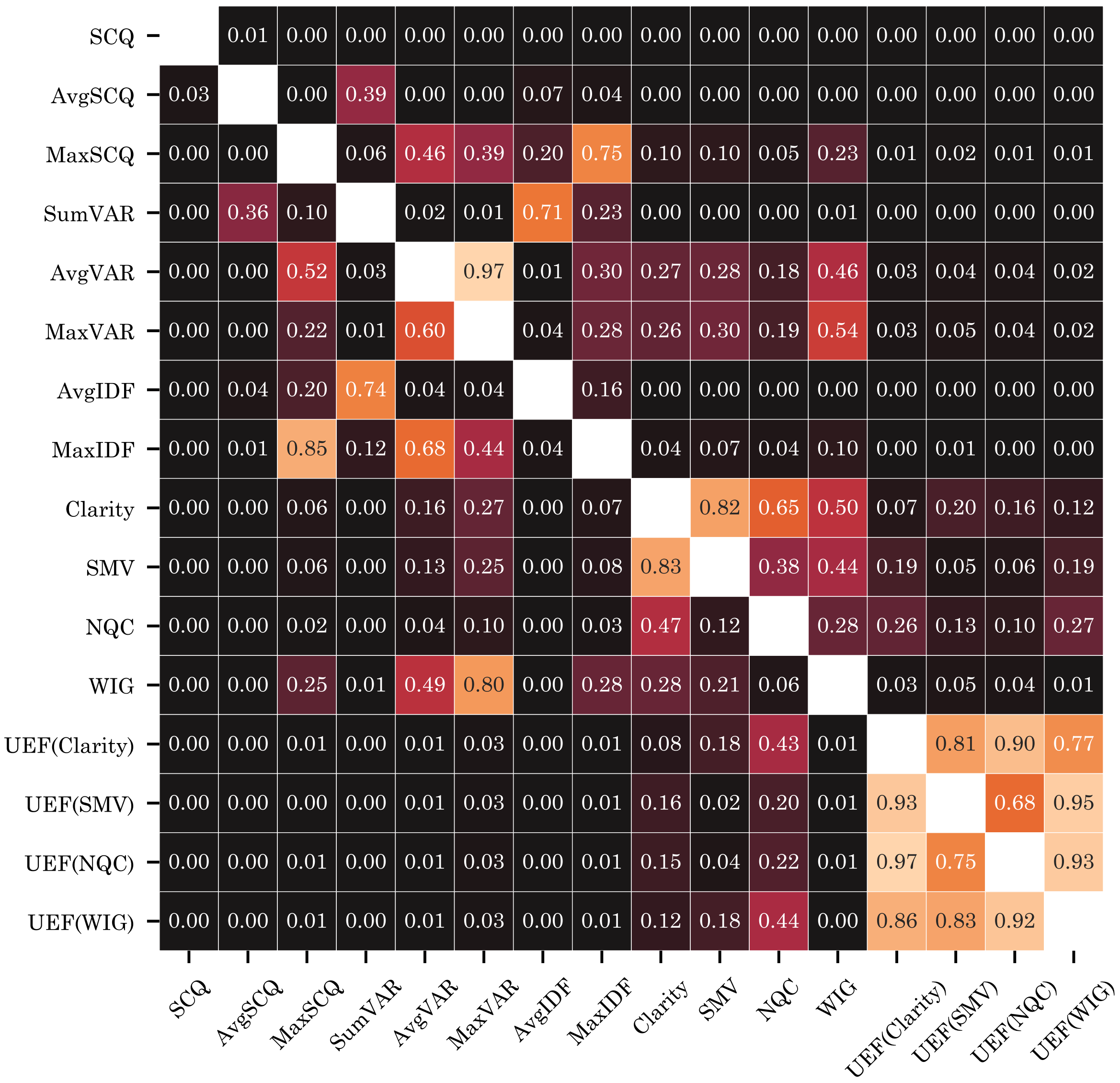
For tie-breaking strategies, we observe that the `average`, `min`, `max`, and `first` tie-breaking strategies all exhibit similar behavior with sARE, with the exception of `dense` tie-breaking which produces much more widely dispersed results. Observe, for example, the additional peak in the distribution when using the `dense` approach. This peak corresponds to the small peaks observed in the other tie breaking strategies, but is inflated in size, when compared to the others. The dense approach is strongly influenced by the number of ties present in the ranking list. This causes the results to be unpredictable since they depend on the randomly observed magnitude (the quantity of ties), which is not correlated with the magnitude of our goal – the performance of QPP. As a result, we recommend against using the `dense` approach, since it may overly inflate performance differences between systems.

Turning our attention to the `first` tie-breaking approach, even though it has a roughly similar distribution to the other strategies, it also introduces a bias as the queries are sorted in an arbitrary order. Such an order does not depend on the actual performance. This problem is particularly relevant when we have large number of ties. In general, if we have many queries and small groups of ties, then the bias does not heavily impact sARE. Nevertheless, we recommend against using it, in order to minimize any possible corner cases. Based on these experimental results, in the remainder of our experiments we will use the `average` tie-breaking method, as it is the most common method, and was the best performing method in our experimental analysis.

With respect to ranking error, we observe that both sARE (scaled Absolute Rank Error) and sRSRE (scaled Root Square Rank Error) have similar density distributions, but sARE is in the [0, 1] interval. Similarly, sSRE is bound by a [0, 1] interval. Overall, the shape of the distribution is quite similar to sARE for our collection, but has two distinguishing differences: it has lower values on average, and it has a smaller range of values. Since differences are squared, sSRE tends to be higher when there are large differences between predicted and observed ranks. Conversely, sARE is larger when there are many errors, even when many of them are small. The smaller values of sSRE when compared against sARE suggests that the QPP models tested tend to make many small errors, and not too many large errors. That is, sSRE is less *discriminative*.

To investigate this further, we compare the two approaches using sensitivity. Using the paired bootstrap test described by Sakai [141], the Achieved Significance Level (ASL) is computed for each pair of QPP methods using the title queries and the bootstrap with $10,000$ samples. The outcome of our pairwise comparisons is presented in Figure 8.7. While in general the patterns are similar, sARE does appear to be more sensitive, identifying 74/120 statistically significantly different pairs (61.7%), compared with 68/120 (56.7%) for sSRE. Note that when using this approach, both methods identify more pairs of predictors which are significantly different (where the significance level is $\alpha = 0.05$) than when using the Kendall's τ correlation measured with bootstrap resampling. Both lead to the SMV predictor being added to the cluster of best performing methods. As discussed previously, sMARE can be associated with Spearman's footrule distance, sMSRE (scaled Mean Squared Rank Error) on the other hand can be associated with Spearman's coefficient of association ρ. While both sARE and sSRE have valuable statistical properties [48], sARE appears to be more sensitive, and is more useful in our ANOVA analysis, as we want to perform a detailed comparative analysis of methods. The sRE (scaled Rank Error), despite being on a larger interval scale ([-1, 1]), is not useful when computing a mean, here called sMRE (scaled Mean Ranked Error), and is always equal to 0. This is easily explainable since the sum over the ranking errors (using the `average` and `first` tie-breaking strategies) will always be equal to 0. So, based on our desiderata, we have adopted the use of sARE /sMARE since: *i)* they are bounded between 0 and 1;[4] and *ii)* sMARE is not always equal to 0.

[4] The values of sSRE are bounded as well, and sMSRE $\in [0, \frac{1}{3})$, or $[0, \frac{1}{\sqrt{3}})$ if the squared root is applied on the mean.



	SCQ	AvgSCQ	MaxSCQ	SumVAR	AvgVAR	MaxVAR	AvgIDF	MaxIDF	Clarity	SMV	NQC	WIG	UEF(Clarity)	UEF(SMV)	UEF(NQC)	UEF(WIG)
SCQ		0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
AvgSCQ	0.03		0.00	0.39	0.00	0.00	0.07	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
MaxSCQ	0.00	0.00		0.06	0.46	0.39	0.20	0.75	0.10	0.10	0.05	0.23	0.01	0.02	0.01	0.01
SumVAR	0.00	0.36	0.10		0.02	0.01	0.71	0.23	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00
AvgVAR	0.00	0.00	0.52	0.03		0.97	0.01	0.30	0.27	0.28	0.18	0.46	0.03	0.04	0.04	0.02
MaxVAR	0.00	0.00	0.22	0.01	0.60		0.04	0.28	0.26	0.30	0.19	0.54	0.03	0.05	0.04	0.02
AvgIDF	0.00	0.04	0.20	0.74	0.04	0.04		0.16	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
MaxIDF	0.00	0.01	0.85	0.12	0.68	0.44	0.04		0.04	0.07	0.04	0.10	0.00	0.01	0.00	0.00
Clarity	0.00	0.00	0.06	0.00	0.16	0.27	0.00	0.07		0.82	0.65	0.50	0.07	0.20	0.16	0.12
SMV	0.00	0.00	0.06	0.00	0.13	0.25	0.00	0.08	0.83		0.38	0.44	0.19	0.05	0.06	0.19
NQC	0.00	0.00	0.02	0.00	0.04	0.10	0.00	0.03	0.47	0.12		0.28	0.26	0.13	0.10	0.27
WIG	0.00	0.00	0.25	0.01	0.49	0.80	0.00	0.28	0.28	0.21	0.06		0.03	0.05	0.04	0.01
UEF(Clarity)	0.00	0.00	0.01	0.00	0.01	0.03	0.00	0.01	0.08	0.18	0.43	0.01		0.81	0.90	0.77
UEF(SMV)	0.00	0.00	0.00	0.00	0.01	0.03	0.00	0.01	0.16	0.02	0.20	0.01	0.93		0.68	0.95
UEF(NQC)	0.00	0.00	0.01	0.00	0.01	0.03	0.00	0.01	0.15	0.04	0.22	0.01	0.97	0.75		0.93
UEF(WIG)	0.00	0.00	0.01	0.00	0.01	0.03	0.00	0.01	0.12	0.18	0.44	0.00	0.86	0.83	0.92	

Fig. 8.7 ASL value comparison showing the sensitivity of the sSRE and sARE deviation measures. Values above the diagonal show ASL values for sSRE and the ASL values for sARE are below the diagonal. Computing the sARE pairs result yields $ASL < 0.05 : 74/120$ (61.7%) and the sSRE pairs yield $ASL < 0.05 : 68/120$ (56.7%)

8.6 Breaking Down QPP Performance Using sMARE

We are now in a position to introduce our first ANOVA model which will enable a more comprehensive experimental analysis of the results:

$$y_{iqrs} = \mu + \tau_i + \gamma_q + \delta_r + \zeta_s + \varepsilon_{iqrs} \qquad (\text{MD4.1.0}_{micro})$$

where:

$y_{i...}$ is the performance (sARE-AP) on the i-th topic (using the specified QPP pipeline);

μ is the *grand mean*;

τ_i is the effect of the i-th topic (represented with the title query formulation); γ_q, δ_r, and ζ_s are the effect of the q-th stoplist, the r-th stemmer, and the s-th QPP model;

ε_{iqrs} is the error component.

This model mimics model MD4$_{mi}$ in the QPP domain. The main difference with model MD4$_{mi}$ relies on the fact that MD4.1.0$_{micro}$ replaces the IR model with the QPP one and does not include the interaction between different system's components.

Table 8.1 summarizes the ANOVA results of our first experiment. It can be seen that the stoplist, the stemmer, and the QPP model have a small effect size, while the topic effect is large, indicating that most of the performance of the QPP depends on the chosen topic.

Table 8.1 MD4.1.0$_{micro}$ ANOVA on the Robust 04 collection. Topics are represented with the title queries. SS: Sum of Squares; DF: Degrees of Freedom; MS: Mean Square; F: F statistics.

Source	**SS**	**DF**	**MS**	**F**	**p-value**	$\hat{\omega}^2_{\langle fact\rangle}$
Topic	876.524	248	3.534	168.136	<0.001	0.410
Stoplist	1.185	4	0.296	14.095	<0.001	0.001
Stemmer	5.218	2	2.609	124.108	<0.001	0.004
QPP model	46.569	15	3.105	147.691	<0.001	0.036
Error	1250.538	59490	0.021			
Total	2180.034	59759				

Based on these results, we next ran a Tukey's HSD post-hoc analysis to test for pairwise comparisons. Figure 8.8 shows the Tukey's HSD confidence intervals for sMARE-AP over the different QPP models. Comparing Figure 8.1 (orange bars) and Figure 8.8, we can

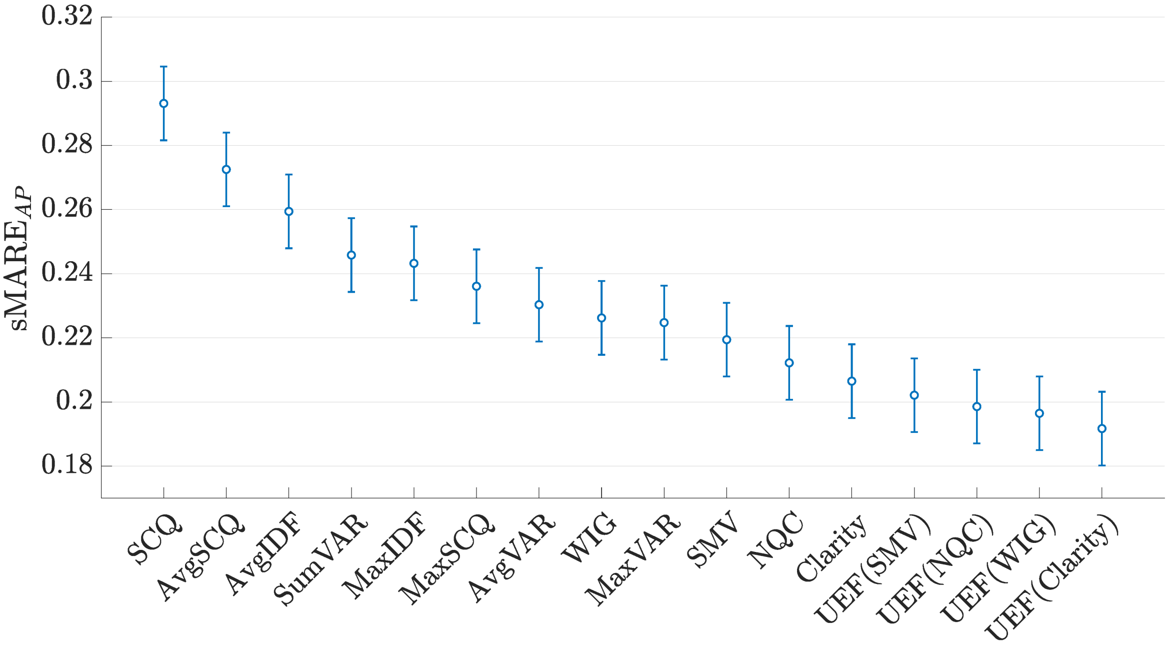
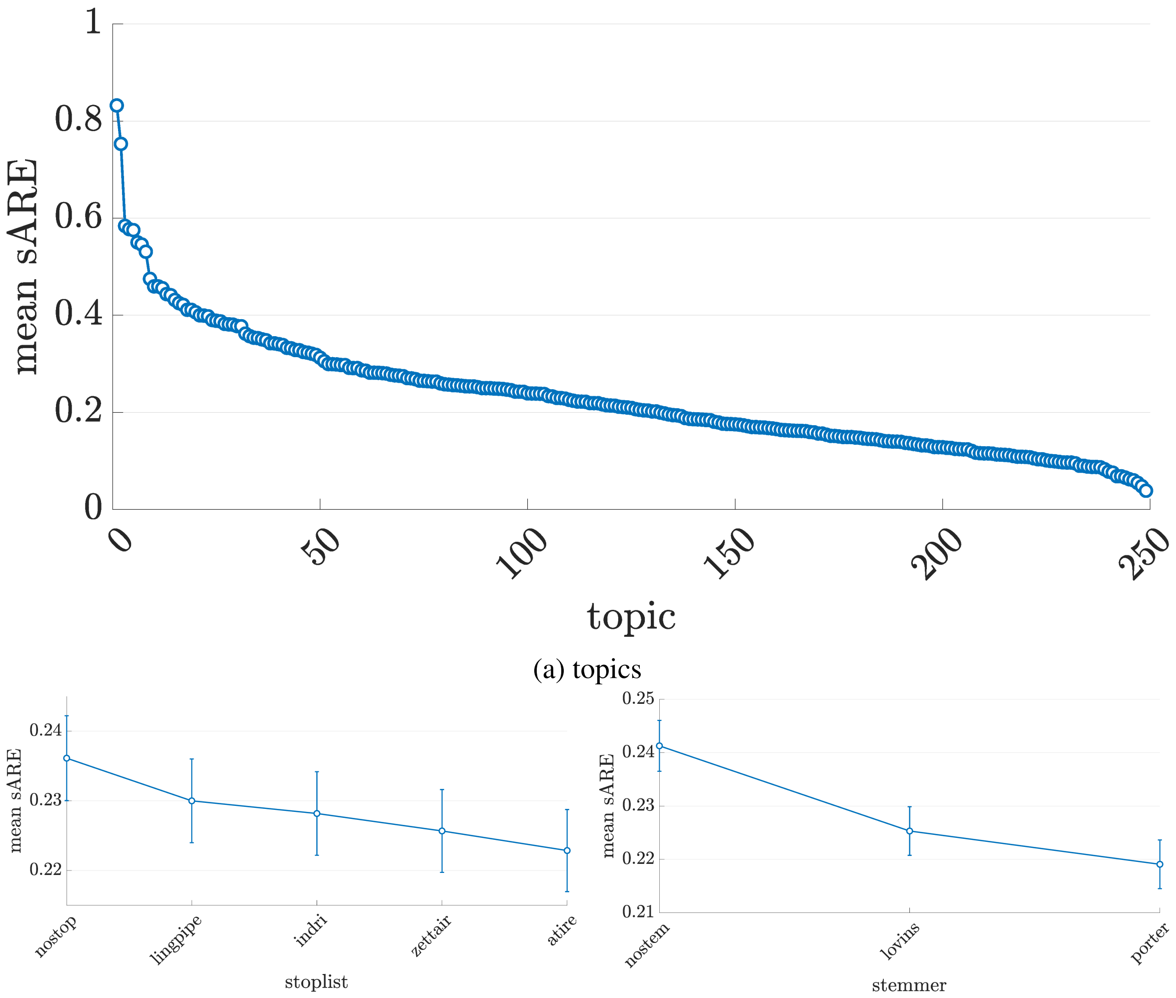


Fig. 8.8 Confidence Intervals of sMARE-AP from MD4.1.0$_{micro}$ for the QPP models on the Robust 04 title queries.

observe that there is less overlap between the CIs, in particular computing the p-values for the pairwise comparisons, out of 120 pairs of predictors, 96 of them are significantly different (80.0%). The outcomes observed when using the bootstrap-based approach resulted in 68.4% [5] more statistically significant differences between predictor pairs when compared against the original data, and the top performing cluster consists of UEF(WIG), UEF(SMV), UEF(NQC), and UEF(Clarity).

The "Topic" factor, as Table 8.1 suggests, is responsible for the largest part of the variance; this is in line with results from IR effectiveness evaluation (see for example Tague-Sutcliffe and Blustein [168]). Thus, the estimate of the performance for a specific QPP model can vary significantly as it is dependent on properties of the underlying collection (performance differences in topics/queries). By removing the contribution of the topics from the global variance, ANOVA removes any volatility in the underlying experimental data, therefore allowing the relative performance of predictors to be compared more precisely. When using only correlations aggregated across all topics, such information is lost, while an ANOVA analysis facilitates more discriminative performance comparisons between systems by systematically accounting for each factor separately.

[5] $(96-57)/57 = 0.684$, where 96 is the number of statistically significantly different pairs found now, and 57 pairs were found using the bootstrap based approach.



(a) topics

(b) stoplists

(c) stemmers

Fig. 8.9 Main effects for topics, stoplists and stemmers of sMARE-AP from MD4.1.0$_{micro}$ for the QPP models on the Robust 04 title queries. We also report the confidence interval for stoplists and stemmers. We do not report CI for the topics, for the sake of image readability.

Figure 8.9 shows the main effects observed for different factors and levels when using ANOVA with MD4.1.0$_{micro}$. From Figure 8.9a we can see that, in line with Table 8.1, the topic factor exhibits a very large variance. The topics `'356'` and `'679'` present a very large sARE (0.832 and 0.753 respectively). The title formulation for topic `'356'` is "`postmenopausal estrogen Britain`", while for topic `'679'` it is "`opening adoption records`".

Figure 8.9b shows the main effect for the different stoplists included in our analysis. It is interesting to observe that the variance over the different stoplists is very small – changing from the best stoplist (`atire`) to the worst (`nostop`) only leads to an increase of approximately 1.5%. Furthermore, post-hoc analysis shows that `atire` and `zettair` are not statistically significanlty different, while `indri`, `lingpipe` and `nostop` are statistically significantly worse then `atire`. Furthermore, all the stoplists help QPP models in predicting performance more accurately.

Figure 8.9c highlights the main effect for the stemmer component. Note that the stemmer selected has a bigger impact than the stoplist. Using the best stemmer allows us to predict the performance of the queries more easily. In more detail, we observe that Porter's stemmer performs best, followed by Lovins's stemmer. The worst approach is to not use stemming. All pairs of stemmers show a statistically significant difference in performance.

8.7 ANOVA Modeling of Multiple Queries and Interactions

To more fully explore the impact of the query formulations on the performance of QPP predictor, we use the ANOVA model MD4.1.0$_{micro}$ in a multiple query formulation setting. We randomly sample 5 formulations[6] to represent the topics. In total, 1,245 different queries were used. Then, we compute the sARE score for each query-predictor pairing. The ANOVA summary table computed using the model MD4.1.0$_{micro}$ of multiple formulations of topics is shown in Table 8.2. Comparing Table 8.2 to Table 8.1, we can see that the introduction of multiple query formulations and model MD4.1.0$_{micro}$ results in a reduced topic effect size, with the originally observed large-size effect becoming a medium-to-large sized effect. The introduction of the multiple formulations increases the variance of each topic, so the possible score differences between the topics tend to be smaller, smoothing the effect size. The QPP model factor effect is similar for both models. The large SS for the Error component indicates that this model is not suitable if we wish to study/explain any variance in the data. To do this, the model complexity must be increased in order to fit the data more tightly.

To help the model more fit the data more closely, one possible solution is to include a query Formulation factor in the ANOVA. This allows the partial modeling of the additional

[6] The topic with the minimal number of query formulations had 5 formulations.

Table 8.2 Summary table for ANOVA using model MD4.1.0$_{micro}$ and representing topics with multiple formulations.

Source	**SS**	**DF**	**MS**	**F**	**p-value**	$\hat{\omega}^2_{\langle fact \rangle}$
Topic	1653.019	248	6.665	214.777	<0.001	0.151
Stoplist	0.405	4	0.101	3.266	0.0110	<0.001
Stemmer	12.726	2	6.363	205.028	<0.001	0.001
QPP model	349.503	15	23.300	750.795	<0.001	0.036
Error	9264.609	298530	0.031			
Total	11280.263	298799				

variance due to the multiple formulations for each topic. Therefore, we now propose another alternative as an ANOVA model:

$$y_{ijqrs} = \mu + \tau_i + \nu_{j(i)} + \gamma_q + \delta_r + \zeta_s + \varepsilon_{ijqrs} \qquad (\text{MD4.1.0f}_{micro})$$

The model MD4.1.0f$_{micro}$ extends model MD4.1.0$_{micro}$ by including $\nu_{j(i)}$, the effect of the j-th formulation of the i-th topic. Note that Topic, Stoplist, Stemmer, and QPP model are crossed since each of them can be used in combination with all the others. This is not the case for the multiple formulations of a topic. A formulation can represent only the topic used to create it. Therefore, we cannot treat the formulation as a *crossed* factor, and so query formulations are *nested* for each Topic factor. This ensures the variance produced by different query formulations contribute only to the variance of the topic they represent.

Table 8.3 presents the results of the ANOVA when using model MD4.1.0f$_{micro}$. In order to differentiate the case where a Formulation factor is nested in a Topic from our previous models, we use the term “Formulation (Topic)”. When examining Table 8.3, observe that the effect of the performance of both the Topic and Formulation is a large-sized effect. For formulations of a topic, a good formulation can dramatically change the performance of a QPP model. The effect of the QPP model observed in Table 8.3 is still small-sized, but has a relative increase of 22.2% when compared against Table 8.2. Such observations highlight the importance of introducing query formulations into the analysis, both in our data and in the ANOVA model, allowing us to learn more about a predictor. Note that the model MD4.1.0f$_{micro}$ still results in a high SS error. This indicates that the model may benefit from further modification to model the data more tightly. However, this will require additional efficiency improvements to be made for running multi-factor ANOVA algorithms on large data collections. The current techniques used to compute the models in this work

Table 8.3 Summary table for ANOVA using model MD4.1.0f$_{micro}$ and representing topics with multiple formulations.

Source	**SS**	**DF**	**MS**	**F**	**p-value**	$\hat{\omega}^2_{\langle fact \rangle}$
Topic	1653.019	248	6.665	260.704	<0.001	0.177
Formulation(Topic)	1657.578	996	1.664	65.093	<0.001	0.176
Stoplist	0.405	4	0.101	3.965	0.0032	0.000
Stemmer	12.726	2	6.363	248.871	<0.001	0.002
QPP model	349.503	15	23.300	911.343	<0.001	0.044
Error	7607.031	297534	0.026			
Total	11280.263	298799				

already require substantial memory and computational resources, and successfully increasing the complexity of the model, or using additional data, is unlikely using any of the currently available hardware and software at either of our universities. We run the above-describe ANOVA via Matlab (version 2017b) on a server with 72 Intel(R) Xeon(R) Gold 6140M CPU 2.30GHz. The largest analysis occupied 250GB of RAM and it required approximately 200 hours to fit the whole model.

One of the most interesting aspects of our framework is the ability to compute the effect size for the interactions between factors. This is possible since the relative performance of a QPP model for each topic can be computed using sARE , and multiple query formulations were introduced as a nested factor. The resulting ANOVA model MD4.1.1$_{micro}$ includes component level interactions, and is defined as:

$$\begin{aligned} y_{ijqrs} = {} & \mu + \tau_i + \nu_{j(i)} + \gamma_q + \delta_r + \zeta_s + (\tau\gamma)_{iq} + (\tau\delta)_{ir} + (\tau\zeta)_{is} \\ & + (\nu\gamma)_{j(i)q} + (\nu\delta)_{j(i)r} + (\nu\zeta)_{j(i)s} + (\gamma\delta)_{qr} + (\gamma\zeta)_{qs} + (\delta\zeta)_{rs} + \varepsilon_{ijqrs} \end{aligned} \qquad \text{(MD4.1.1}_{micro}\text{)}$$

This model extends MD4.1.0f$_{micro}$ to include all possible two-way interactions. It can be seen as the exact replica of model MD4$_{mi}$ in the QPP domain.

Table 8.4 presents the ANOVA summary statistics for the model MD4.1.1$_{micro}$. The table empirically shows that the largest differences in QPP performance are due to the topics, and their formulations. While the importance of topics is a well-known phenomenon, our model is able to explicitly quantify the magnitude of this effect. The effect for the QPP factor is medium-sized (medium-sized effects are associated with ω^2 between 6% and 14%). It is important to note that the dimension of the effect is due to the wide variety of QPP models (and their performance) that are taken into account. For example, a practitioner wishing to

Table 8.4 MD4.1.1$_{micro}$ ANOVA applied on Robust 04 collection. ω^2 for non-significant factors is ill-defined and thus not reported. When compared against [55], a different set of random formulations for the topics is used: which leads small differences in the results – the Sum of the Squares being the largest. Nevertheless, the magnitude of the effects and the p-values, which are the focus in an ANOVA, are the same as those in [55].

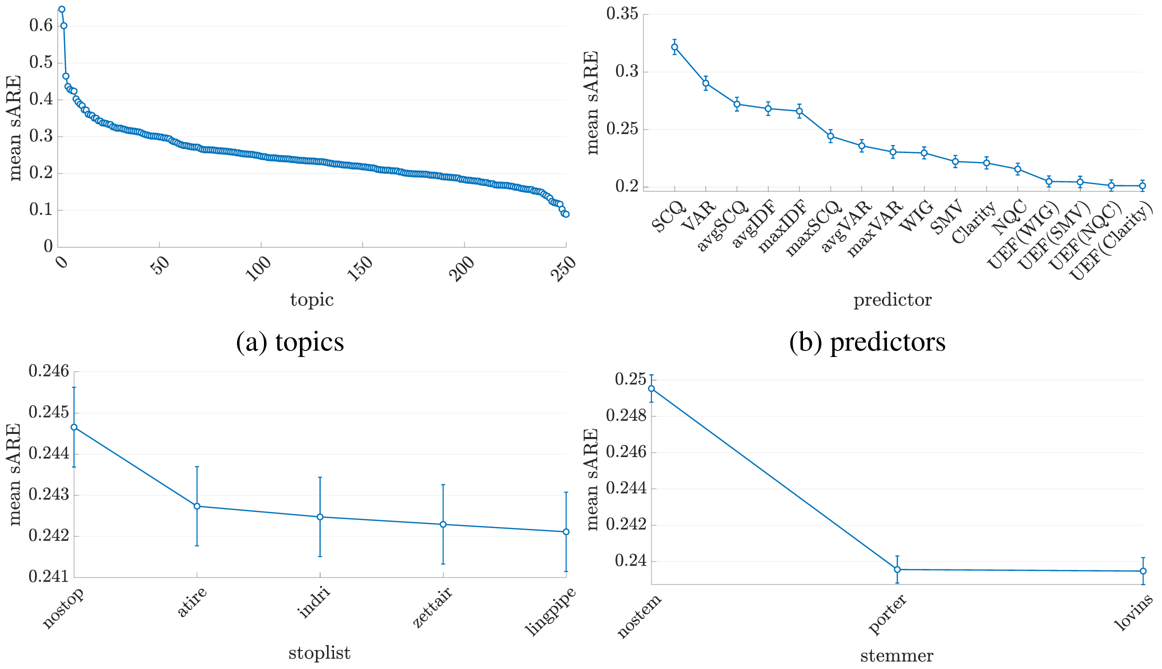
Source	**SS**	**DF**	**MS**	**F**	**p-value**	$\hat{\omega}^2_{\langle fact \rangle}$
Topic	1653.019	248	6.665	1186.233	<0.001	0.496
Formulation(Topic)	1657.578	996	1.664	296.182	<0.001	0.496
Stoplist	0.405	4	0.101	18.041	<0.001	0.001
Stemmer	12.726	2	6.363	1132.393	<0.001	0.008
QPP model	349.503	15	23.300	4146.715	<0.001	0.172
Topic*Stoplist	39.333	992	0.040	7.057	<0.001	0.020
Topic*Stemmer	147.087	496	0.297	52.776	<0.001	0.079
Topic*QPP model	2297.031	3720	0.617	109.892	<0.001	0.575
Frm.*Stoplist	85.596	3984	0.021	3.824	<0.001	0.036
Frm.*Stemmer	292.736	1992	0.147	26.154	<0.001	0.144
Frm.*QPP model	3215.366	14940	0.215	38.302	<0.001	0.651
Stoplist*Stemmer	0.041	8	0.005	0.918	0.5000	—
Stoplist*QPP model	0.840	60	0.014	2.492	<0.001	<0.001
Stemmer*QPP model	4.509	30	0.150	26.749	<0.001	0.003
Error	1524.492	271312	0.006			
Total	11280.263	298799				

evaluate new QPP models may observe a smaller ω^2 for the QPP model factor if the relative performance differences between the models being compared is less substantial.

The effect sizes of different stoplists and stemmers are both small, but still significant. This suggests that stemmers and stoplists may affect overall prediction quality, and practitioners should consider all possible factors when comparing and contrasting QPP performance for a corpus.

We are now in a position to explore the interaction between topics (and their query formulations) and the predictors. The large effect size indicates that important differences between QPP model performance exist within reformulations of a single topic. Identifying the QPP model where interactions are smallest is valuable in practice, as this corresponds to be choosing a model that is the most robust to query reformulation. Additionally, this approach enables a series of additional analyses, such as a failure analysis for topics to determine which QPP model has the largest interactions with another factor.

There are many additional factors that can influence the performance of the QPP method, beyond the ones tested in the current model. For example, other ranking algorithms or



(a) topics

(b) predictors

(c) stoplists

(d) stemmers

Fig. 8.10 Main effects observed using model MD4.1.1$_{micro}$ with multiple topic formulations.

evaluation measures can also be used with sMARE, and could provide new experimental evidence and insights into performance differences between various QPP models in the future.

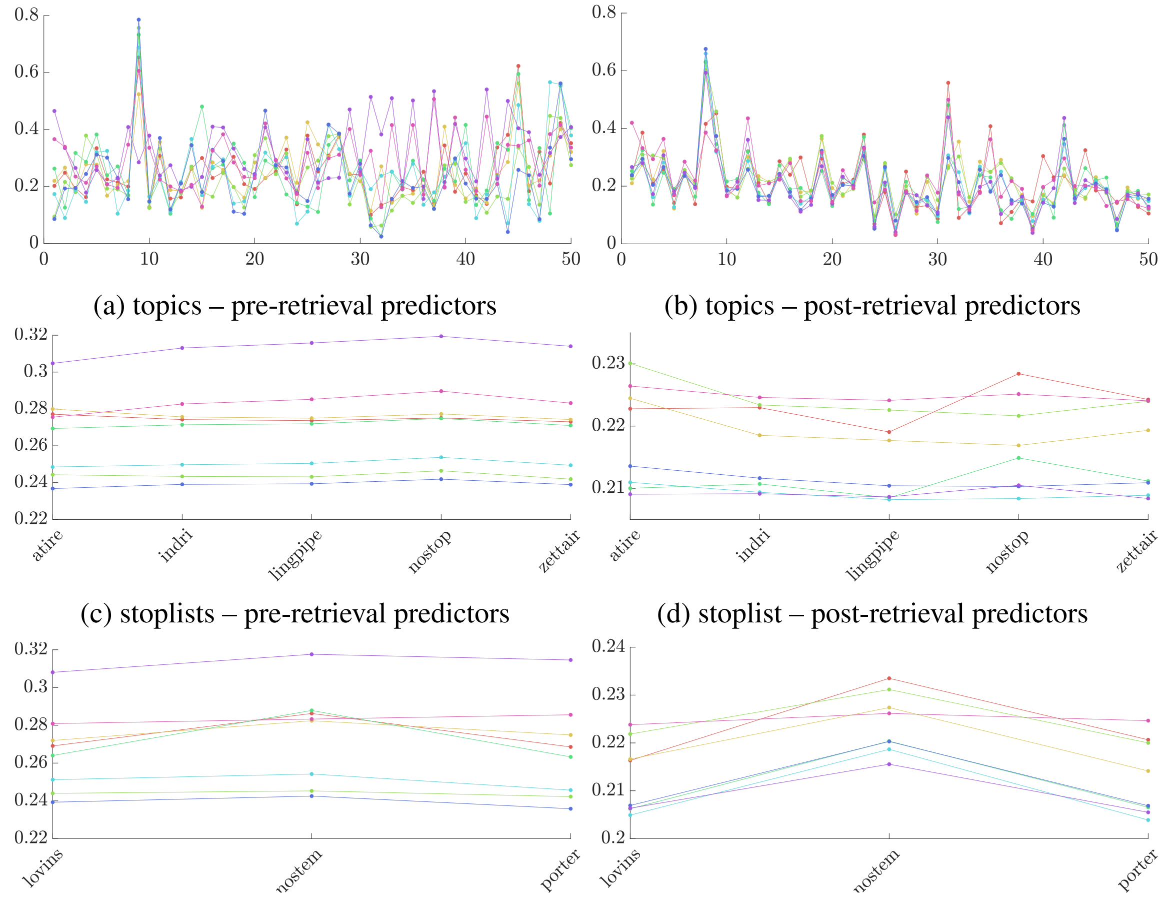
Figure 8.10 shows the main effects observed using the multiple formulations. Comparing the plot with Figure 8.9, we can see that overall, the results tend to be more uniform when multiple formulations are included. Formulations tend to have large performance variability: such variability is responsible for the flattening of relative predictor performance. Nevertheless, they give additional power to statistical techniques, allowing the obtaining of more precise results that better generalize to reality. Comparing Figure 8.10a to Figure 8.9a we observe that the main effects for the topics tend to be more stable, with a smaller variance. We still have two outliers – the biggest outlier is ‘356’, which also had the biggest effect in MD4.1.0$_{micro}$. The second is ‘344’, with the title formulation “Abuses of E-Mail”. These two topics have sMARE of 0.6463 and 0.6016, respectively. Observing that the topic ‘356’ remains particularly complex suggests that the problem is likely linked to the semantic gap between the topic formulations and the relevant documents for that topic. In contrast to what was observed in Figure 8.9a, the topic ‘679’ is not an outlier anymore. This corroborates what was observed in Table 8.4, showing the importance of the query formulations: different formulations might help the predictors to estimate the query difficulty.

Figure 8.10b shows an interesting pattern when compared to Figure 8.8. In particular, we observe that the distribution of the main effects contains much more evident steps if we include multiple query formulations. While the overall order of predictors is close to the one that we observed previously, using multiple formulations we are better able to distinguish between clusters of systems. In particular, SCQ performance suggests that it belongs to its own cluster of quality. VAR, avgSCQ, avgIDF and maxIDF form a distinct cluster, and so do maxSCQ, avgVAR and maxVAR. We then have two clusters of post-retrieval predictors: the first includes the original form of all the predictors, while the second includes the UEF version.

Figures 8.10c and 8.10d show the main effect for stoplists and stemmers respectively, when multiple formulations are included in the analysis. The post-hoc analysis shows that all the stoplists are statistically significantly different from the no-stop approach, indicating the importance of applying a stoplist in the QPP scenario. Nevertheless, they are all in the same equivalence class. This empirically suggests that what makes the difference in the QPP setting is either removing stopwords or not, but the stoplists are overall equivalent. Similar conclusions can be drawn for stemmers: in Figure 8.10d) both stemmers (Porter's and Lovins') are statistically better than the no-stemming approach, but the two stemming approaches do not differ statistically significantly.

Figure 8.11 shows the interaction plots for the model MD4.1.1$_{micro}$. We report the interaction between the predictors and topic, stoplist and stemmer factors. The predictors are further separated into pre- and post-retrieval approaches, shown one the left and right, respectively. Figures 8.11a and 8.11b describe the interaction between topics and predictors. Note that, to ease the readability, we report the interaction of the systems with 50 randomly sampled topics. Similar results where observed with different topic samplings. Both plots exemplify the strong interaction between the predictors and the topics, showing several cross-overs between lines and lines tending not to be parallel. This in general confirms what was observed in Table 8.4. Nevertheless, we observe that lines for the post-retrieval predictors (Figure 8.11b) are more stable (a similar conclusion can be reached also by looking at Figure 8.10b). This means that *i)* different post-retrieval predictors tend to perform more similarly than pre-retrieval predictors; and *ii)* the interaction between topics and post-retrieval predictors is lower compared to that between pre-retrieval predictors and topics.

Concerning the stoplist and stemmer components, Figures 8.11c and 8.11e illustrate how much they interact with the pre-retrieval predictors. In both cases, the interaction between the component and the predictor is light, with parallel lines overall. The only exception to this is avgIDF and maxIDF, which show a swift drop in performance when used in combination with the nostem approach. Figure 8.11d reports the interaction between the different post-



(a) topics – pre-retrieval predictors

(b) topics – post-retrieval predictors

(c) stoplists – pre-retrieval predictors

(d) stoplist – post-retrieval predictors

(e) stemmers – pre-retrieval predictors

(f) stemmers – post-retrieval predictors

Fig. 8.11 Interaction effects observed using model MD4.1.1$_{micro}$ with multiple topic formulations. We report interactions between pre-retrieval (left) and post-retrieval (right) models, with topics (top), stoplists (center) and stemmers (bottom).

retrieval QPP models with the stoplist component. The choice of stoplist does not interact particularly with the post-retrieval methods, as also shown in Table 8.4. The QPP approach most affected by the different stoplists is Clarity, both in its traditional and UEF versions. This indicates that, if the practitioner intends to use Clarity, it is important to validate its performance over different stoplists. On the other hand, the WIG model (both traditional and UEF versions) is the most stable. Concerning the choice of stemmer, Figure 8.11f shows that the stemmer interacts slightly with predictor performance, similar to what was observed for pre-retrieval QPP approaches. All of the QPP models appear to be overall stable across different stemmers, with small interaction with the stemmer used. In particular, most QPP models suffer when query terms are not stemmed. The traditional version of WIG is the most stable QPP approach for this: it does not benefit if the stemmer is used or not.

8.8 Final Remarks

In both Chapters 6 and 7, we observed intrinsic challenges in evaluating the performance of QPP frameworks. The setting in which a QPP model can be applied vary widely. Therefore, we can either use the traditional correlation based QPP evaluation – which presents some important expressiveness flaws, or define an ad-hoc evaluation, which limits our capability of comparing different approaches. Motivated by such observations, in this chapter we proposed a new approach, roughly based on Spearman’s foot rule, dubbed sMARE, that allows us to measure the quality of a QPP model as a whole and measure the performance of the QPP model topic by topic. This new approach allows for a more fine-grained analysis enabling us to understand what topics the QPP model fails. Nevertheless, it also allows a more powerful global comparison between different models since it allows for the computation of ANOVA models on the performance of the QPP models. This has the two-fold advantage of better discriminating statistically between systems and breaking down the performance on different factors of our experimental pipelines.

Chapter 9

Conclusions and Future Work

This manuscript concerned determining how to model the performance of an Information Retrieval (IR) system, to predict its performance before deployment. We, therefore, investigated two main areas: the modelling and the prediction of IR systems performance

Modeling Information Retrieval Performance The work concerning the modeling of IR systems performance mainly focuses on three main aspects: i) the role of the ANOVA as a tool to interpret and model the information retrieval systems performance; ii) the concept of topic difficulty and how to interpret it in light of different query formulations and collections; iii) how to improve the performance modeling via Generalized Linear Models (GLMs).

As a first analysis, in Chapter 3 we studied more in detail one of the most used tools in IR evaluation: ANOVA. In particular, we considered two recent ANOVA approaches - the traditional ANOVA and the bootstrap ANOVA (an ANOVA model based on the sampling with replacement of the residuals). We observed that, in general, the traditional ANOVA tend to be more stable, but also less powerful in terms of statistically significantly different pairs of systems identified, while the bootstrap ANOVA is more powerful, but also less stable.

Chapter 4 addresses the second point, where we tried to determine the impact of the collections and the query formulations on the performance of a system. We observed a high impact in terms of formulations and concerning the interaction between the formulations and the systems themselves. This hinted that the topic is likely not intrinsically complex, but rather it is not correctly expressed, considering the system at hand. We thus developed an experimental methodology to show that we can induce any sorting of the topics, showing that their complexity is not intrinsic but a consequence of poorly formulated queries.

Finally, in Chapter 5 we addressed the possibility of switching from Linear Models to Generalized Linear Models when evaluating IR systems, to better fit IR data. We studied how different links behave when it comes to evaluating IR systems. We observed that, if we switch from the traditional linear model to GLMs with non-linear links, we can better fit the experimental IR data in terms of deviance. Moreover, the inference attained by the GLMs is more powerful, identifying more pairs of statistically significantly different pairs of systems, while maintaining the same degree of stability compared to the traditional linear models.

Predicting Information Retrieval Performance Our investigation in terms of performance prediction started from two new QPP approaches: i) a predictor to recognize the best performing query variation when tackling the systematic review task; ii) a predictor capable of determining when the semantic gap afflicts a query severely, making it challenging from the semantic standpoint. In our analyses, we noticed that an important challenge was caused by the absence of a pointwise evaluation methodology in the QPP scenario. We, therefore, iii) analyzed the evaluation procedures used to assess the performance of QPP models and developed a new measure dubbed sMARE.

Our first analysis in the domain of query performance prediction concerned the identification of the best performing topic formulation when considering the medical systematic review task in which systems are measured in terms of recall. We addressed this in Chapter 6, where we extended a well-known approach to computing the gain provided by different topic formulations. The gain approach we developed for our specific setting allows us to identify the query most likely capable of achieving the best result over a set of topic reformulations.

The second aspect related to the query performance prediction addressed in this manuscript concerns the prediction of the prominence of the semantic gap on a given query. We analyzed it in Chapter 7. We first developed a procedure to label queries as semantically or lexically hard. Secondly, we identified a set of features of such query that correlates with the prominence of the semantic gap. Using such features, we trained a classifier that identifies beforehand whether lexical or semantic models perform better.

Given the challenges presented by the evaluation of the previous two predictors, for which we had to resort to an ad-hoc evaluation methodology, in Chapter 8 we revisited the evaluation methodology currently used in the QPP scenario. To address the challenges we noticed, mostly linked to using a single point estimation to describe the performance, we developed a measure, dubbed sARE, that allows modeling the performance achieved by the QPP model as a distribution. This new measure allows for more powerful break-down statistical analyses (e.g., ANOVA).

Future Work Our future analyses go in multiple directions. There is still much to understand about why specific formulations perform better than others. If we were able to understand what characteristics of a given formulation make it more suited for a certain system we could i) predict better the performance of such query; ii) select the most suited query over a set of formulations; iii) select the system that is likely to perform the best. In this sense, what learned in Chapters 4 and 7, can be the starting point of this research path.

Concerning the modeling of IR systems performance, we can expand our work knowledge on GLMs used in IR by considering more complex scenarios where we include multiple formulations and different systems components. Furthermore, we can consider changing the distribution used in the GLM to make our statistical models more suited to our empirical observations.

In Chapter 4 we showed the importance in terms of evaluation of multiple query formulations, with Chapters 6, 7, but also 8, we exhibited the power that reformulations can have in predicting the performance of a system or evaluating a QPP engine. We are motivated in pursuing such a research path, by developing QPP models capable of keeping into account the multiple query formulations, but also all the different aspects and features that we showed to have a strong impact on the performance of a system.

In recent years, the developments in Natural Language Processing (NLP) concerning contextualized embeddings and large language models, led to the development of new IR systems, such as dense retrieval and Neural IR approaches, that underlie different rationale and behave very differently from traditional ones. We are thus interested in exploiting the knowledge gained through our work to develop new QPP approaches that better suit such novel IR systems.

Finally, several relevant emerging research areas and retrieval tasks can also benefit from what we studied in this manuscript. For example, one domain where the need for QPP techniques is getting urgent is the conversational search: understanding if a user's query will perform good or if the system needs the user to reformulate it is of uttermost importance in this domain. Our objective is to understand whether the findings of this manuscript generalize to other IR tasks and how we can use our knowledge of IR systems, topics and performance.

List of acronyms

AA	Active Agreements	46
AAP	Average Average Precision	68
AD	Active Disagreements	46
ANOVA	ANalysis Of VAriance	xiii
AP	Average Precision	xiii
ASL	Achieved Significance Level	159
BH	Benjamini Hochberg	13
CDF	Cumulative Density Function	xx
CHS	Consumer Health Search	4
CI	Confidence Interval	27
DF	Degrees of Freedom	10
FDR	False Discovery Rate	13
FWER	Family-wise Error Rate	13
GLM	Generalized Linear Model	xx
GLMM	General Linear Mixed Model	7
GLiM	General Linear Model	15
GoP	Grid of Points	xiii
HSD	Honestly Significant Difference	xx
IDF	Inverse Document Frequency	16
IR	Information Retrieval	xvii
KDE	Kernel Density Estimation	39
KLD	Kullback-Leibler Divergence	66
LM	Language Model	31
MA	Mixed Agreement	46
MAP	Mean Average Precision	xiv
MD	Mixed Disagreements	46

MLE	Maximum Likelihood Estimation	39
MS	Mean Squares	10
nDCG	Normalized Discounted Cumulated Gain	18
NIST	National Institute of Standards and Technology	17
NLP	Natural Language Processing	175
OLS	Ordinary Least Squares	9
P	Precision	40
PA	Passive Agreements	46
PPA	Proportion of Active Agreements	47
PD	Passive Disagreements	xvii
PDF	Probability Density Function	39
PPA	Proportion of Passive Agreements	47
QE	Query Expansion	54
QPP	Query Performance Prediction	xv
RBP	Rank-Biased Precision	18
RM	Relevance Model	31
RMSE	Root Mean Square Error	27
RSS	Sum of Squares of Residuals	95
sARE	scaled Absolute Rank Error	xv
sARE-AP	AP induced scaled Absolute Rank Error	153
sMARE-AP	AP induced scaled Mean Absolute Rank Error	xv
SOA	Strength of Association	135
SS	Sum of Squares	10
SSD	Statistically Significantly Different	xvii
TF	Term Frequency	16
TREC	Text REtrieval Conference	17

References

[1] Agosti, M., Marchesin, S., and Silvello, G. (2020). Learning Unsupervised Knowledge-Enhanced Representations to Reduce the Semantic Gap in Information Retrieval. *ACM Transactions on Information Systems*, 38(4):38:1–38:48.

[2] Allan, J., Harman, D. K., Kanoulas, E., Li, D., Van Gysel, C., and Voorhees, E. (2018). TREC 2017 Common Core Track Overview. In *Proceedings of the 26th Text REtrieval Conference TREC*.

[3] Allan, J., Harman, D. K., Kanoulas, E., and Voorhees, E. (2019). TREC 2018 Common Core Track Overview. In *Proceedings of the 28th Text REtrieval Conference TREC*.

[4] Amati, G., Carpineto, C., and Romano, G. (2004). Query Difficulty, Robustness, and Selective Application of Query Expansion. In *Proceedings of the 26th European Conference on IR Research, ECIR*, pages 127–137.

[5] Amati, G. and van Rijsbergen, C. J. (2002). Probabilistic Models of Information Retrieval based on measuring the Divergence From Randomness. *ACM Transactions on Information Systems*, 20(4):357–389.

[6] Aslam, J. A. and Pavlu, V. (2007). Query Hardness Estimation Using Jensen-Shannon Divergence Among Multiple Scoring Functions. In *Proceedings of the 29th European Conference on IR Research, ECIR*, pages 198–209.

[7] Bailey, P., Moffat, A., Scholer, F., and Thomas, P. (2015). User Variability and IR System Evaluation. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development on Information Retrieval*, pages 625–634.

[8] Bailey, P., Moffat, A., Scholer, F., and Thomas, P. (2016). UQV100: A Test Collection with Query Variability. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development on Information Retrieval*, page 725–728.

[9] Bailey, P., Moffat, A., Scholer, F., and Thomas, P. (2017). Retrieval Consistency in the Presence of Query Variations. In *Proceedings of the 40th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 395–404.

[10] Balasubramanian, N. (2011). *Query-Dependent Selection of Retrieval Alternatives*. PhD thesis, University of Massachusetts Amherst.

[11] Balasubramanian, N. and Allan, J. (2010). Learning to Select Rankers. In *Proceedings of the 33rd International ACM SIGIR conference on Research and development in information retrieval*, pages 855–856.

[12] Banks, D., Over, P., and Zhang, N.-F. (1999). Blind Men and Elephants: Six Approaches to TREC Data. *Information Retrieval Journal*, 1(1):7–34.

[13] Belkin, N. J., Cool, C., Croft, W. B., and Callan, J. P. (1993). The effect of multiple query variations on information retrieval system performance. In *Proceedings of the 16th International ACM SIGIR conference on Research and development in information retrieval*, pages 339–346.

[14] Belkin, N. J., Kantor, P., Fox, E. A., and Shaw, J. A. (1995). Combining the evidence of multiple query representations for information retrieval. *Information Processing & Management*, 31(3):431–448.

[15] Benham, R. and Culpepper, J. S. (2017). Risk-Reward Trade-offs in Rank Fusion. In *Proceedings of 2017 Australasian Document Computing Symposium*, pages 1:1–1:8.

[16] Benham, R., Culpepper, J. S., Gallagher, L., Lu, X., and Mackenzie, J. (2018a). Towards efficient and effective query variant generation. In *Design of Experimental Search & Information REtrieval Systems (DESIRES)*, pages 62–67.

[17] Benham, R., Gallagher, L., Mackenzie, J., Damessie, T. T., Chen, R.-C., Scholer, F., Moffat, A., and Culpepper, J. S. (2017). RMIT at the 2017 TREC CORE track. In *Proceedings of the 26th Text REtrieval Conference TREC*.

[18] Benham, R., Gallagher, L., Mackenzie, J., Liu, B., Lu, X., Scholer, F., Moffat, A., and Culpepper, J. S. (2018b). RMIT at the 2018 TREC CORE track. In *Proceedings of the 27th Text REtrieval Conference TREC*.

[19] Benham, R., Mackenzie, J., Moffat, A., and Culpepper, J. S. (2019). Boosting Search Performance Using Query Variations. *ACM Transactions on Information Systems*, 37(4):1–25.

[20] Benjamini, Y. and Hochberg, Y. (1995). Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *J. Royal Stat. Soc.*, 57(1):289–300.

[21] Berto, A., Mizzaro, S., and Robertson, S. (2013). On Using Fewer Topics in Information Retrieval Evaluations. In *Proceedings of the 2nd International Conference on Advances in Information Retrieval Theory*, page 30–37.

[22] Bodoff, D. and Li, P. (2007). Test theory for assessing ir test collections. In *Proceedings of the 30th International ACM SIGIR Conference on Research and Development on Information Retrieval*, pages 367–374.

[23] Brennan, R. L. (2001). *Generalizability Theory*. Springer-Verlag, New York, USA.

[24] Buckley, C., Salton, G., Allan, J., and Singhal, A. (1995). Automatic query expansion using SMART: TREC 3. In *Proceedings of the 4th Text REtrieval Conference TREC*, pages 69–69.

[25] Buckley, C. and Voorhees, E. (2005). Retrieval System Evaluation. In *Proceedings of the 14th Text REtrieval Conference TREC*, pages 53–78.

[26] Burgin, R. (1992). Variations in relevance judgments and the evaluation of retrieval performance. *Information Processing & Management*, 28(5):619–627.

[27] Burnham, K. P. and Anderson, D. R. (2002). *Model Selection and Multimodel Inference. A Practical Information-Theoretic Approach.* Springer-Verlag, Heidelberg, Germany, 2nd edition.

[28] Carmel, D. and Yom-Tov, E. (2010). Estimating the Query Difficulty for Information Retrieval. In *Proceeding of the 33rd International ACM SIGIR conference on Research and development in information retrieval*, pages 911–911.

[29] Carmel, D., Yom-Tov, E., Darlow, A., and Pelleg, D. (2006). What Makes a Query Difficult? In *Proceedings of the 29th International ACM SIGIR Conference on Research and Development on Information Retrieval*, page 390–397.

[30] Carterette, B. (2011). Model-Based Inference about IR Systems. In *Proceedings of the Third International Conference on Advances in Information Retrieval Theory*, page 101–112.

[31] Carterette, B. (2012). Multiple Testing in Statistical Analysis of Systems-based Information Retrieval Experiments. *ACM Transactions on Information Systems*, 30(1):1–34.

[32] Carterette, B. (2015). Bayesian Inference for Information Retrieval Evaluation. In *Proceedings of the 7th International Conference on Advances in Information Retrieval Theory*, pages 31–40.

[33] Carterette, B. (2017). But Is It Statistically Significant? Statistical Significance in IR Research, 1995-2014. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1125–1128.

[34] Carterette, B., Pavlu, V., Fang, H., and Kanoulas, E. (2009). Million query track 2009 overview. In *Proceedings of the 18th Text REtrieval Conference TREC*.

[35] Chifu, A.-G., Laporte, L. é. a., Mothe, J., and Ullah, M. Z. (2018). Query Performance Prediction Focused on Summarized Letor Features. In *Proceedings of the 41st International ACM SIGIR Conference on Research and Development on Information Retrieval*, pages 1177–1180.

[36] Clancy, R., Ferro, N., Hauff, C., Lin, J., Sakai, T., and Wu, Z. Z. (2019). The SIGIR 2019 Open-Source IR Replicability Challenge (OSIRRC 2019). In *Proceedings of the 42nd ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1432–1434.

[37] Cormack, G. V. and Lynam, T. R. (2006). Statistical Precision of Information Retrieval Evaluation. In *Proceedings of the 29th International ACM SIGIR Conference on Research and Development on Information Retrieval*, pages 533–540.

[38] Cormack, G. V. and Lynam, T. R. (2007). Validity and power of t-test for comparing map and gmap. In *Proceedings of the 30th International ACM SIGIR conference on Research and development in information retrieval*, pages 753–754.

[39] Croft, W. B., Metzler, D., and Strohman, T. (2009). *Search Engines: Information Retrieval in Practice*. Addison-Wesley, Reading (MA), USA.

[40] Cronen-Townsend, S. and Croft, W. B. (2002). Quantifying query ambiguity. In *Proceeding of the ACM HLT Conference on Human Language Technology*, pages 104–109.

[41] Cronen-Townsend, S., Zhou, Y., and Croft, W. B. (2002). Predicting Query Performance. In *Proceedings of the 25th International ACM SIGIR conference on Research and development in information retrieval*, pages 299–306.

[42] Cronen-Townsend, S., Zhou, Y., and Croft, W. B. (2004). A Language Modeling Framework for Selective Query Expansion. Technical report, Center for Intelligent Information Retrieval, University of Massachusetts.

[43] Culpepper, J. S., Faggioli, G., Ferro, N., and Kurland, O. (2021). Topic difficulty: Collection and Query Formulation Effects. *ACM Transactions on Information Systems*, 40(1):1–36.

[44] Cummins, R. (2014). Document Score Distribution Models for Query Performance Inference and Prediction. *ACM Transactions on Information Systems*, 32(1):2:1–2:28.

[45] Dammesie, T., Scholer, F., and Culpepper, J. S. (2016). The influence of topic difficulty, relevance level, and document ordering on relevance judging. In *Proceedings of the 21st Australasian Document Computing Symposium*, pages 41–48.

[46] Dang, V., Bendersky, M., and Croft, W. B. (2010). Learning to Rank Query Reformulations. In *Proceedings of the 33rd International ACM SIGIR conference on Research and development in information retrieval*, page 807–808.

[47] Di Nunzio, G. M. and Faggioli, G. (2021). A study of a gain based approach for query aspects in recall oriented tasks. *Applied Sciences*, 11(19):9075.

[48] Diaconis, P. and Graham, R. L. (1977). Spearman's Footrule as a Measure of Disarray. *J. Royal Stat. Soc.*, 39(2):262–268.

[49] Diaz, F. (2007). Performance Prediction Using Spatial Autocorrelation. In *Proceedings of the 30th International ACM SIGIR Conference on Research and Development on Information Retrieval*, page 583–590.

[50] Edinger, T., Cohen, A. M., Bedrick, S., Ambert, K. H., and Hersh, W. R. (2012). Barriers to Retrieving Patient Information from Electronic Health Record Data: Failure Analysis from the TREC Medical Records Track. In *AMIA 2012, American Medical Informatics Association Annual Symposium*, pages 180–188.

[51] Efron, B. and Tibshirani, R. J. (1994). *An Introduction to the Bootstrap*. Chapman and Hall/CRC, USA.

[52] Faggioli, G., Ferrante, M., Ferro, N., Perego, R., and Tonellotto, N. (2022). A Dependency-Aware Utterances Permutation Strategy to Improve Conversational Evaluation. In *Proceedings of the 44th European Conference on IR Research, ECIR*, pages 184–198.

[53] Faggioli, G. and Ferro, N. (2021). System Effect Estimation by Sharding: A Comparison Between ANOVA Approaches to Detect Significant Differences. In *Proceedings of the 43rd European Conference on IR Research, ECIR*, pages 33–46.

[54] Faggioli, G. and Marchesin, S. (2021). What makes a query semantically hard? In *Proceedings of the 2nd International Conference on Design of Experimental Search & Information REtrieval Systems*, DESIRES '21.

[55] Faggioli, G., Zendel, O., Culpepper, J. S., Ferro, N., and Scholer, F. (2021). An enhanced evaluation framework for query performance prediction. In *Lecture Notes in Computer Science*, pages 115–129.

[56] Ferrari Dacrema, M., Boglio, S., Cremonesi, P., and Jannach, D. (2019). A Troubling Analysis of Reproducibility and Progress in Recommender Systems Research. *User Modeling and User-Adapted Interaction*, pages 1 – 49.

[57] Ferro, N. (2018). IMS @ TREC 2017 Core Track. In *Proceedings of the 26th Text REtrieval Conference TREC.*

[58] Ferro, N. and Harman, D. (2010). CLEF 2009: Grid@CLEF Pilot Track Overview.

[59] Ferro, N., Kim, Y., and Sanderson, M. (2019). Using Collection Shards to Study Retrieval Performance Effect Sizes. *ACM Transactions On Information Systems (TOIS)*, 5(44):59.

[60] Ferro, N., Maistro, M., Sakai, T., and Soboroff, I. (2018). Overview of centre@ clef 2018: a first tale in the systematic reproducibility realm. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 239–246.

[61] Ferro, N. and Sanderson, M. (2017). Sub-corpora Impact on System Effectiveness. In *Proceedings of the 40th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 901–904.

[62] Ferro, N. and Sanderson, M. (2019). Improving the Accuracy of System Performance Estimation by Using Shards. In *Proceedings of the 42th International ACM SIGIR Conference on Research and Development on Information Retrieval*, pages 805–814.

[63] Ferro, N. and Sanderson, M. (2022). How do you Test a Test? A Multifaceted Examination of Significance Tests. In *Proceedings of the 15th ACM International Conference on Web Search and Data Mining*, pages 280–288.

[64] Ferro, N. and Silvello, G. (2016). A General Linear Mixed Models Approach to Study System Component Effects. In *Proceedings of the 39th International ACM SIGIR conference on Research and development in information retrieval*, pages 25–34.

[65] Ferro, N. and Silvello, G. (2018). Toward an anatomy of IR system component performances. *jasist*, 69(2):187–200.

[66] Fuhr, N. (2017). Some Common Mistakes In IR Evaluation, And How They Can Be Avoided. *SIGIR Forum*, 51(3):32–41.

[67] Gibbons, J. D. and Chakraborti, S. (2011). *Nonparametric Statistical Inference*. Chapman & Hall/CRC, Taylor and Francis Group, Boca Raton (FL), USA, 5th edition.

[68] Griffiths, A., Luckhurst, H. C., and Willett, P. (1986). Using Interdocument Similarity Information in Document Retrieval Systems. *J. Am. Soc. Inf. Sci.*, 37(1):3–11.

[69] Grossman, M. R., Cormack, G. V., and Roegiest, A. (2016). TREC 2016 Total Recall Track Overview. In *Proceedings of the 25th Text REtrieval Conference TREC*.

[70] Harman, D. (1993). Overview of the first trec conference. In *Proceedings of the 16thACM SIGIR International Conference on Theory of Information Retrieval*, page 36–47.

[71] Harman, D. K. (1994). Overview of the Third Text REtrieval Conference (TREC-3). In *Proceedings of the 3rd Text REtrieval Conference TREC*, pages 1–19.

[72] Hauff, C., Azzopardi, L., and Hiemstra, D. (2009). The Combination and Evaluation of Query Performance Prediction Methods. In *Proceedings of the 31st European Conference on IR Research, ECIR*, pages 301–312.

[73] Hauff, C., Hiemstra, D., and de Jong, F. (2008). A Survey of Pre-Retrieval Query Performance Predictors. In *Proceedings of the 17th ACM International conference on Information and knowledge management - CIKM*, pages 1419–1420.

[74] Hauff, C., Kelly, D., and Azzopardi, L. (2010). A comparison of user and system query performance predictions. In *Proceedings of the 19th ACM International conference on Information and knowledge management - CIKM*, pages 979–988.

[75] He, B. and Ounis, I. (2004a). A Query-based Pre-retrieval Model Selection Approach to Information Retrieval. In *Proceedings of Computer-Assisted Information Retrieval (Recherche d'Information et ses Applications) - RIAO*, pages 706–719.

[76] He, B. and Ounis, I. (2004b). Inferring Query Performance Using Pre-retrieval Predictors. In *Proceedings of the String Processing and Information Retrieval, 11th International Conference, SPIRE*, pages 43–54.

[77] Hersh, W., Buckley, C., Leone, T. J., and Hickam, D. (1994). Ohsumed: An interactive retrieval evaluation and new large test collection for research. In *Proceedings of the 17th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 192–201.

[78] Hsu, J. C. (1996). *Multiple Comparisons. Theory and methods*. Chapman and Hall/CRC, USA.

[79] Hull, D. A. (1993). Using Statistical Testing in the Evaluation of Retrieval Experiments. In *Proceedings of the 16th International ACM SIGIR Conference on Research and Development on Information Retrieval*, pages 329–338.

[80] Ito, P. K. (1980). Robustness of ANOVA and MANOVA test procedures. In *Handbook of Statistics – Analysis of Variance*, pages 199–236.

[81] Järvelin, K. and Kekäläinen, J. (2002). Cumulated Gain-Based Evaluation of IR Techniques. *ACM Transactions on Information Systems*, 20(4):422–446.

[82] Jimmy, Zuccon, G., Palotti, J. R. M., Goeuriot, L., and Kelly, L. (2018). Overview of the CLEF 2018 Consumer Health Search Task. In *Working Notes of CLEF 2018 - Conference and Labs of the Evaluation Forum, Avignon, France, September 10-14, 2018*, CEUR Workshop Proceedings, page 2125.

[83] Jones, T., Turpin, A., Mizzaro, S., Scholer, F., and Sanderson, M. (2014). Size and Source Matter: Understanding Inconsistencies in Test Collection-Based Evaluation. In *Proceedings of the 23rd ACM International conference on Information and knowledge management - CIKM*, pages 1843–1846.

[84] Kastner, M., Straus, S., and Goldsmith, C. H. (2007). Estimating the Horizon of articles to decide when to stop searching in systematic reviews: an example using a systematic review of RCTs evaluating osteoporosis clinical decision support tools. *Annual Symposium proceedings. AMIA Symposium*, 2007:389–393.

[85] Kempthorne, O. and Doerfler, T. (1969). The behaviour of some significance tests under experimental randomization. *Biometrika*, 56(2):231–248.

[86] Kendall, M. G. (1938). A New Measure of Rank Correlation. *Biometrika*, 30(1/2):81–93.

[87] Kendall, M. G. (1945). The Treatment of Ties in Ranking Problems. *Biometrika*, 33(3):239–251.

[88] Kendall, M. G. (1948). *Rank correlation methods*. Griffin.

[89] Kocabaş, İ., Dinçer, B. T., and Karaoğlan, B. (2014). A nonparametric term weighting method for information retrieval based on measuring the divergence from independence. *Information Retrieval Journal*, 17(2):153–176.

[90] Koopman, B. and Zuccon, G. (2014). Why Assessing Relevance in Medical IR is Demanding. In *Proceedings of the Medical Information Retrieval Workshop at SIGIR co-located with the 37th annual International ACM SIGIR conference (ACM SIGIR 2014)*, volume 1276 of *CEUR Workshop Proceedings*, pages 16–19.

[91] Koopman, B., Zuccon, G., Bruza, P., Sitbon, L., and Lawley, M. (2016). Information retrieval as semantic inference: a Graph Inference model applied to medical search. *Inf. Retr. Journal*, 19(1-2):6–37.

[92] Kullback, S. and Leibler, R. A. (1951). On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86.

[93] Kutner, M. H., Nachtsheim, C. J., Neter, J., and Li, W. (2005). *Applied Linear Statistical Models*. McGraw-Hill/Irwin, New York, USA, 5th edition.

[94] Kwok, K. L., Grunfeld, L., Sun, H. L., Deng, P., and Dinstl, N. (2004). Trec 2004 robust track experiments using pircs.

[95] Laurikkala, J., Juhola, M., Kentala, E., Lavrac, N., Miksch, S., and Kavsek, B. (2000). Informal Identification of Outliers in Medical Data. In *Fifth International Workshop on Intelligent Data Analysis in Medicine and Pharmacology*, volume 1, pages 20–24.

[96] Lavrenko, V. and Croft, W. B. (2001). Relevance-based Language Models. In *Proceedings of the 24th International ACM SIGIR Conference on Research and Development on Information Retrieval*, pages 120–127.

[97] Lehmann, E. L. and Romano, J. P. (2005). Generalizations of the familywise error rate. *The Annals of Statistics*, 33(3):719–735.

[98] Levi, O., Raiber, F., Kurland, O., and Guy, I. (2016). Selective Cluster-Based Document Retrieval. In *Proceedings of the 25th ACM International Conference on Information and Knowledge Management, CIKM*, pages 1473–1482.

[99] Li, H. and Xu, J. (2014). Semantic Matching in Search. *Found. Trends Inf. Retr.*, 7(5):343–469.

[100] Lin, J., Mackenzie, J., Kamphuis, C., Macdonald, C., Mallia, A., Siedlaczek, M., Trotman, A., and de Vries, A. (2020). Supporting Interoperability Between Open-Source Search Engines with the Common Index File Format. In *Proceedings of the 43rd ACM SIGIR Conference on Research and Development on Information Retrieval*, pages 2149–2152.

[101] Liu, B., Craswell, N., Lu, X., Kurland, O., and Culpepper, J. S. (2019). A Comparative Analysis of Human and Automatic Query Variants. In *Proceedings of the 42nd ACM SIGIR International Conference on Theory of Information Retrieval*, pages 47–50.

[102] Liu, X. and Croft, W. B. (2006). Experiments on retrieval of optimal clusters. Technical report, University of Massachusetts Amherst.

[103] Liu, X., Nie, J. Y., and Sordoni, A. (2016). Constraining Word Embeddings by Prior Knowledge - Application to Medical Information Retrieval. In *Proceedings of the 12th Asia Information Retrieval Societies Conference, AIRS*, pages 155–167.

[104] Lu, X., Kurland, O., Culpepper, J. S., Craswell, N., and Rom, O. (2019). Relevance Modeling with Multiple Query Variations. In *Proceedings of the 42nd ACM SIGIR International Conference on Theory of Information Retrieval*, pages 27–34.

[105] Madsen, H. and Thyregod, P. (2010). *Introduction to General and Generalized Linear Models*. CRC Press.

[106] Marchesin, S., Purpura, A., and Silvello, G. (2020). Focal elements of neural information retrieval models. an outlook through a reproducibility Study. *Information Processing & Management*, 57(6):102–109.

[107] Maxwell, S. and Delaney, H. D. (2004). *Designing Experiments and Analyzing Data. A Model Comparison Perspective*. Lawrence Erlbaum Associates, Mahwah (NJ), USA, 2nd edition.

[108] Mendenhall, W. and Sincich, T. (2012). *A Second Course in Statistics. Regression Analysis*. Prentice Hall, USA, 7th edition.

[109] Meng, X. L., Rosenthal, R., and Rubin, D. B. (1992). Comparing Correlated Correlation Coefficients. *Psychological Bulletin*, 111(1):172–175.

[110] Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. In *Proceedings of the 1st International Conference on Learning Representations, ICLR*.

[111] Mitra, B., Craswell, N., et al. (2018). An introduction to neural information retrieval. *Foundations and Trends® in Information Retrieval*, 13(1):1–126.

[112] Mizzaro, S. (2008). The Good, the Bad, the Difficult, and the Easy: Something Wrong with Information Retrieval Evaluation? In *Lecture Notes in Computer Science*, pages 642–646.

[113] Mizzaro, S. and Robertson, S. (2007). Hits hits trec: exploring IR evaluation results with network analysis. In *Proceedings of the 30th International ACM SIGIR conference on Research and development in information retrieval*, pages 479–486.

[114] Moffat, A., Scholer, F., and Thomas, P. (2012). Models and Metrics: IR Evaluation as a User Process. In *Proceedings of 2012 Australasian Document Computing Symposium*, pages 47–54.

[115] Moffat, A. and Zobel, J. (2008). Rank-biased Precision for Measurement of Retrieval Effectiveness. *ACM Transactions on Information Systems*, 27(1):2:1–2:27.

[116] Mothe, J. and Tanguy, L. (2005). Linguistic Features to Predict Query Difficulty. In *Proceedings of the Predicting query difficulty-methods and applications workshop, co-located with the ACM Conference on research and Development in Information Retrieval, SIGIR 2005*, pages 7–10.

[117] Myers, R. H., Montgomery, D. C., Vining, G. G., and Robinson, T. J. (2012). *Generalized Linear Models: with Applications in Engineering and the Sciences*, volume 791. John Wiley & Sons.

[118] Nelder, J. A. and Wedderburn, R. W. M. (1972). Generalized Linear Models. *Journal of the Royal Statistical Society. Series A (General)*, 135(3):370–384.

[119] Newman, D. (1939). The distribution of range in samples from a normal population, expressed in terms of an independent estimate of standard deviation. *Biometrika*, 31(1/2):20–30.

[120] Olejnik, S. and Algina, J. (2003). Generalized Eta and Omega Squared Statistics: Measures of Effect Size for Some Common Research Designs. *Psychological Methods*, 8(4):434–447.

[121] Parapar, J., Losada, D. E., and Barreiro, A. (2021). Testing the tests: Simulation of rankings to compare statistical significance tests in information retrieval evaluation. In *Proceedings of the 36th Annual ACM Symposium on Applied Computing*, page 655–664.

[122] Parapar, J., Losada, D. E., Presedo-Quindimil, M. A., and Barreiro, A. (2020). Using score distributions to compare statistical significance tests for information retrieval evaluation. *Journal of the Association for Information Science and Technology*, 71(1):98–113.

[123] Pehcevski, J., Thom, J. A., Vercoustre, A.-M., and Naumovski, V. (2010). Entity ranking in Wikipedia: utilising categories, links and topic difficulty prediction. *Information Retrieval Journal*, 13(5):568–600.

[124] Peng, J., Macdonald, C., He, B., Plachouras, V., and Ounis, I. (2007). Incorporating Term Dependency In The DFR Framework. In *Proceedings of the 30th International ACM SIGIR conference on Research and development in information retrieval*, pages 843–844.

[125] Pérez-Iglesias, J. and Araujo, L. (2010). Standard Deviation as a Query Hardness Estimator. In *String Processing and Information Retrieval*, pages 207–212.

[126] Ponte, J. and Croft, W. B. (1998). A language modeling approach to information retrieval. In *Proceedings of the 21st ACM SIGIR Conference on Research and Development on Information Retrieval*, pages 275–281.

[127] Raiber, F. and Kurland, O. (2014). Query-performance prediction: setting the expectations straight. In *Proceedings of the 37th International ACM SIGIR conference on Research and development in information retrieval*, pages 13–22.

[128] Robertson, S. (2006). On GMAP: and Other Transformations. In *Proceedings of the 15th ACM International conference on Information and knowledge management - CIKM*, pages 78–83.

[129] Robertson, S. (2012). On Smoothing Average Precision. In *Advances in Information Retrieval*, pages 158–169.

[130] Robertson, S. and Kanoulas, E. (2012). On Per-topic Variance in IR Evaluation. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development on Information Retrieval*, pages 891–900.

[131] Robertson, S. and Zaragoza, U. (2009). The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389.

[132] Robertson, S. E. (1977). The probability ranking principle in ir. *Journal of documentation*.

[133] Rocchio, J. J. (1971). *The SMART retrieval system: Experiments in automatic document processing*. Prentice Hall, Englewood Cliffs, NJ.

[134] Roegiest, A., Cormack, G. V., Grossman, M. R., and Clarke, C. L. (2015). TREC 2015 Total Recall Track Overview. In *Proceedings of the 24th Text REtrieval Conference TREC*.

[135] Roitero, K., Maddalena, E., and Mizzaro, S. (2017). Do easy topics predict effectiveness better than difficult topics? In *Lecture Notes in Computer Science*, pages 605–611.

[136] Roitman, H. (2017). An Enhanced Approach to Query Performance Prediction Using Reference Lists. In *Proceedings of the 40th International ACM SIGIR conference on Research and development in information retrieval*, pages 869–872.

[137] Roitman, H. (2018a). An Extended Query Performance Prediction Framework Utilizing Passage-Level Information. In *Proceedings of the 41th International ACM SIGIR Conference on Research and Development on Information Retrieval*, page 35–42.

[138] Roitman, H. (2018b). Query Performance Prediction using Passage Information. In *Proceedings of the 41st International ACM SIGIR conference on Research and development in information retrieval*, pages 893–896.

[139] Roitman, H. (2020). ICTIR Tutorial: Modern Query Performance Prediction: Theory and Practice. In *Proceedings of the 43th International ACM SIGIR Conference on Research and Development on Information Retrieval*, pages 195–196.

[140] Rutherford, A. (2011). *ANOVA and ANCOVA. A GLM Approach.* John Wiley & Sons, New York, USA, 2nd edition.

[141] Sakai, T. (2006). Evaluating Evaluation Metrics based on the Bootstrap. In *Proceedings of the 29th International ACM SIGIR Conference on Research and Development on Information Retrieval*, pages 525–532.

[142] Sakai, T. (2014). Metrics, Statistics, Tests. In *Bridging Between Information Retrieval and Databases - PROMISE Winter School 2013, Revised Tutorial Lectures*, pages 116–163.

[143] Sakai, T. (2016). Statistical Significance, Power, and Sample Sizes: A Systematic Review of SIGIR and TOIS, 2006-2015. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 5–14.

[144] Sakai, T. (2020). On Fuhr's Guideline for IR Evaluation. *SIGIR Forum*, 54(1):p14:1–p14:8.

[145] Salton, G. (1968). *Automatic Information Organization and Retrieval.* McGraw Hill Text.

[146] Salton, G. and McGill, M. J. (1983). *Introduction to modern information retrieval.* mcgraw-hill.

[147] Sanderson, M., Turpin, A., Zhang, Y., and Scholer, F. (2012). Differences in effectiveness across sub-collections. In *Proceedings of the 21st ACM International conference on Information and knowledge management - CIKM*, pages 1965–1969.

[148] Sanderson, M. and Zobel, J. (2005). Information retrieval system evaluation: effort, sensitivity, and reliability. In *Proceedings of the 28th International ACM SIGIR conference on Research and development in information retrieval*, pages 162–169.

[149] Savoy, J. (1997). Statistical Inference in Retrieval Effectiveness Evaluation. *Information Processing & Management*, 33(44):495–512.

[150] Scariano, S. M. and Davenport, J. M. (1987). The Effects of Violations of Independence Assumptions in the One-Way ANOVA. *The American Statistician*, 41(2):123–129.

[151] Scells, H., Azzopardi, L., Zuccon, G., and Koopman, B. (2018). Query Variation Performance Prediction for Systematic Reviews. In *Proceedings of the The 41st International ACM SIGIR conference on Research and development in information retrieval*, page 1089–1092.

[152] Scheffé, H. (1953). A Method For Judging All Contrasts In The Analysis Of Variance. *Biometrika*, 40(1-2):87–110.

[153] Scholer, F. and Garcia, S. (2009). A Case for Improved Evaluation of Query Difficulty Prediction. In *Proceedings of the 32nd ACM SIGIR Conference on Research and Development on Information Retrieval*, page 640–641.

[154] Scholer, F., Kelly, D., Wu, W.-C., Lee, H. S., and Webber, W. (2013). The effect of threshold priming and need for cognition on relevance calibration and assessment. In *Proceedings of the 36th International ACM SIGIR conference on Research and development in information retrieval*, pages 623–632.

[155] Scholer, F., Williams, H. E., and Turpin, A. (2004). Query Association Surrogates for Web Search. *J. Assoc. Inf. Sci. Technol.*, 55(7):637–650.

[156] Sedgwick, P. (2012). Multiple significance tests: the bonferroni correction. *Bmj*, 344.

[157] Shavelson, R. J. and Webb, N. M. (1991). *Generalizability Theory. A Primer*. SAGE Publishing, USA.

[158] Sheldon, D., Shokouhi, M., Szummer, M., and Craswell, N. (2011). LambdaMerge: Merging the results of query reformulations. In *Proceedings of the fourth ACM International conference on Web search and data mining - WSDM*, pages 795–804.

[159] Shtok, A., Kurland, O., and Carmel, D. (2010). Using Statistical Decision Theory and Relevance Models for Query-Performance Prediction. In *Proceedings of the 33rd ACM SIGIR Conference on Research and Development on Information Retrieval*, pages 259–266.

[160] Shtok, A., Kurland, O., and Carmel, D. (2016). Query Performance Prediction Using Reference Lists. *ACM Transactions on Information Systems*, 34(4):19:1–19:34.

[161] Shtok, A., Kurland, O., Carmel, D., Raiber, F., and Markovits, G. (2012a). Predicting Query Performance by Query-Drift Estimation. *ACM Transactions on Information Systems*, 30(2):11.

[162] Shtok, A., Kurland, O., Carmel, D., Raiber, F., and Markovits, G. (2012b). Predicting Query Performance by Query-Drift Estimation. *ACM Transactions on Information Systems*, 30(2):1–35.

[163] Siegel, S. (1957). Nonparametric statistics. *The American Statistician*, 11(3):13–19.

[164] Smucker, M. D., Allan, J., and Carterette, B. (2009). Agreement among statistical significance tests for information retrieval evaluation at varying sample sizes. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, page 630–631.

[165] Smucker, M. D., Allan, J., and Carterette, B. A. (2007). A Comparison of Statistical Significance Tests for Information Retrieval Evaluation. In *Proceedings of the 16th ACM International conference on Information and knowledge management - CIKM*, pages 623–632.

[166] Spärck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*.

[167] Student (1908). The probable error of a mean. *Biometrika*, pages 1–25.

[168] Tague-Sutcliffe, J. M. and Blustein, J. (1994). A Statistical Analysis of the TREC-3 Data. In *Proceedings of the 3rd Text REtrieval Conference TREC*, pages 385–398.

[169] Tao, Y. and Wu, S. (2014). Query Performance Prediction By Considering Score Magnitude and Variance Together. In *Proceedings of the 23rd ACM International conference on Information and knowledge management - CIKM*, page 1891–1894.

[170] Thomas, P., Scholer, F., Bailey, P., and Moffat, A. (2017). Tasks, Queries, and Rankers in Pre-Retrieval Performance Prediction. In *Proceedings of 2017 Australasian Document Computing Symposium*, pages 1–4.

[171] Tombros, A., Villa, R., and van Rijsbergen, C. J. (2002). The Effectiveness of Query-Specific Hierarchic Clustering in Information Retrieval. *Inf. Process. Manag.*, 38(4):559–582.

[172] Tsukuda, K., Sakai, T., Dou, Z., and Tanaka, K. (2013). Estimating Intent Types for Search Result Diversification. In *Information Retrieval Technology*, pages 25–37.

[173] Tukey, J. W. (1949). Comparing Individual Means in the Analysis of Variance. *Biometrics*, 5(2):99–114.

[174] Umemoto, K., Yamamoto, T., and Tanaka, K. (2016). ScentBar: A Query Suggestion Interface Visualizing the Amount of Missed Relevant Information for Intrinsically Diverse Search. In *Proceedings of the 39th International ACM SIGIR conference on Research and development in information retrieval*, pages 405–414.

[175] Urbano, J. (2016). Test Collection Reliability: a Study of Bias and Robustness to Statistical Assumptions via Stochastic Simulation. *Information Retrieval Journal*, 19(3):313–350.

[176] Urbano, J., Lima, H., and Hanjalic, A. (2019). Statistical significance testing in information retrieval: An empirical analysis of type i, type ii and type iii errors. In *Proceedings of the 42nd International ACM SIGIR conference on Research and development in information retrieval*, page 505–514.

[177] Urbano, J., Marrero, M., and Martín, D. (2013). A Comparison of the Optimality of Statistical Significance Tests for Information Retrieval Evaluation. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development on Information Retrieval*, pages 925–928.

[178] Urbano, J. and Nagler, T. (2018). Stochastic Simulation of Test Collections: Evaluation Scores. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, page 695–704.

[179] Van Gysel, C., De Rijke, M., and Kanoulas, E. (2018). Neural vector spaces for unsupervised information retrieval. *ACM Transactions on Information Systems*, 36(4):1–25.

[180] Voorhees, E. (2004). Overview of the TREC 2004 Robust Retrieval Track. In *Proceedings of The 13th Text REtrieval Conference*.

[181] Voorhees, E. (2005). Overview of the trec 2005 robust retrieval track. In *Proceedings of the 14th Text REtrieval Conference TREC*.

[182] Voorhees, E., Alam, T., Bedrick, S., Demner-Fushman, D., Hersh, W. R., Lo, K., Roberts, K., Soboroff, I., and Wang, L. L. (2021). TREC-COVID: Constructing a Pandemic Information Retrieval Test Collection. *SIGIR Forum*, 54(1).

[183] Voorhees, E. and Harman, D. K. (1999). Overview of the Eigth Text REtrieval Conference (TREC-8). In *Proceedings of the 8th Text REtrieval Conference TREC*, pages 1–24.

[184] Voorhees, E., Samarov, D., and Soboroff, I. (2017). Using Replicates in Information Retrieval Evaluation. *ACM Transactions on Information Systems*, 36(2):12:1–12:21.

[185] Vulić, I. and Moens, M. F. (2015). Monolingual and Cross-Lingual Information Retrieval Models Based on (Bilingual) Word Embeddings. In *Proceeding of the 38th International ACM SIGIR conference on Research and development in information retrieval*, pages 363–372.

[186] W, C. and Cleverdon (1962). Report on the testing and analysis of an investigation into the comparative efficiency of indexing systems. *Aslib Cranfield Research Project, College of Aeronautics, Cranfield, UK.*

[187] W, C. and Cleverdon (1997). The cranfield tests on index languages devices. *Readings in Information Retrieval*, page 47–60.

[188] Wackerly, D., Mendenhall, W., and Scheaffer, R. L. (2014). *Mathematical statistics with applications*. Cengage Learning.

[189] Wand, M. P. and Jones, M. C. (1995). *Kernel Smoothing*. Chapman and Hall/CRC, USA.

[190] Wang, L. L., Lo, K., Chandrasekhar, Y., Reas, R., Yang, J., Burdick, D., Eide, D., Funk, K., Katsis, Y., Kinney, R., Li, Y., Liu, Z., Merrill, W., Mooney, P., Murdick, D., Rishi, D., Sheehan, J., Shen, Z., Stilson, B., Wade, A., Wang, K., Wang, N. X. R., Wilhelm, C., Xie, B., Raymond, D., Weld, D. S., Etzioni, O., and Kohlmeier, S. (2020). Cord-19: The covid-19 open research dataset.

[191] Webber, W., Moffat, A., and Zobel, J. (2008). Score Standardization for Inter-Collection Comparison of Retrieval Systems. In *Proceedings of the 31st International ACM SIGIR Conference on Research and Development in Information Retrieval*, page 51–58.

[192] Wilbur, W. J. (1994). Non-parametric Significance Tests of Retrieval Performance Comparisons. *Journal of Information Science*, 20(4):270–284.

[193] Xiong, L., Xiong, C., Li, Y., Tang, K.-F., Liu, J., Bennett, P., Ahmed, J., and Overwijk, A. (2020). Approximate nearest neighbor negative contrastive learning for dense text retrieval. *ArXiv*.

[194] Xu, J. and Croft, W. B. (1996). Query expansion using local and global document analysis. In *Proceedings of the 19th International ACM SIGIR Conference on Research and Development on Information Retrieval*, pages 4–11.

[195] Yilmaz, E., Aslam, J. A., and Robertson, S. (2008). A New Rank Correlation Coefficient for Information Retrieval. In *Proceedings of the 31st ACM SIGIR Conference on Research and Development on Information Retrieval*, pages 587–594.

[196] Yom-Tov, E., Fine, S., Carmel, D., and Darlow, A. (2005). Learning to estimate query difficulty: Including applications to missing content detection and distributed information retrieval. In *Proceedings of the 28th International ACM SIGIR conference on Research and development in information retrieval*, pages 512–519.

[197] Zamani, H., Croft, W. B., and Culpepper, J. S. (2018). Neural Query Performance Prediction Using Weak Supervision from Multiple Signals. In *Proceedings of the 41th International ACM SIGIR Conference on Research and Development on Information Retrieval*, pages 105–114.

[198] Zampieri, F., Roitero, K., Culpepper, J. S., Kurland, O., and Mizzaro, S. (2019). On Topic Difficulty in IR Evaluation: The Effect of Systems, Corpora, and System Components. In *Proceedings of the 42nd International ACM SIGIR conference on Research and development in information retrieval*, pages 909–912.

[199] Zendel, O., Culpepper, J. S., and Scholer, F. (2021). Is Query Performance Prediction With Multiple Query Variations Harder Than Topic Performance Prediction? In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development on Information Retrieval*, pages 1713–1717.

[200] Zendel, O., Shtok, A., Raiber, F., Kurland, O., and Culpepper, J. S. (2019). Information Needs, Queries, and Query Performance Prediction. In *Proceedings of the 42nd ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 395–404.

[201] Zhai, C. (2008). Statistical Language Models for Information Retrieval. A Critical Review. *Foundoundations and Trends in Information Retrieval*, 2(3):137–213.

[202] Zhai, C. and Lafferty, J. (2001). A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval. In *Proceedings of the 24th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 334–342.

[203] Zhao, R. and Grosky, W. I. (2002). Narrowing the semantic gap - improved text-based web document retrieval using visual features. *IEEE Transansactions on Multimedia*, 4(2):189–200.

[204] Zhao, Y., Scholer, F., and Tsegay, Y. (2008). Effective Pre-retrieval Query Performance Prediction Using Similarity and Variability Evidence. In *Proceedings of the 30th European Conference on IR Research, ECIR*, pages 52–64.

[205] Zhou, Y. and Croft, W. B. (2006). Ranking Robustness: A Novel Framework to Predict Query Performance. In *Proceedings of the 15th ACM International conference on Information and knowledge management - CIKM*, page 567–574.

[206] Zhou, Y. and Croft, W. B. (2007). Query Performance Prediction in Web Search Environments. In *Proceedings of the 30th International ACM SIGIR Conference on Research and Development on Information Retrieval*, page 543–550.

[207] Zobel, J. (1998). How Reliable Are the Tesults of Large-Scale Information Retrieval Experiments? In *Proceedings of the 21st International ACM SIGIR conference on Research and development in information retrieval*, pages 307–314.