Analytics Methods to Understand Information Retrieval Effectiveness-A Survey

Josiane Mothe

mathematics

MDPI

Article

Analytics Methods to Understand Information Retrieval Effectiveness—A Survey

Josiane Mothe

INSPE, IRIT UMR5505 CNRS, Université Toulouse Jean-Jaurès, 118 Rte de Narbonne, F-31400 Toulouse, France; Josiane.Mothe@irit.fr; Tel.: +33-5-61556444

Abstract: Information retrieval aims to retrieve the documents that answer users' queries. A typical search process consists of different phases for which a variety of components have been defined in the literature; each one having a set of hyper-parameters to tune. Different studies focused on how and how much the components and their hyper-parameters affect the system performance in terms of effectiveness, others on the query factor. The aim of these studies is to better understand information retrieval system effectiveness. This paper reviews the literature of this domain. It depicts how data analytics has been used in IR to gain a better understanding of system effectiveness. This review concludes that we lack a full understanding of system effectiveness related to the context which the system is in, though it has been possible to adapt the query processing to some contexts successfully. This review also concludes that, even if it is possible to distinguish effective from non-effective systems for a query set, neither the system component analysis nor the query features analysis were successful in explaining when and why a particular system fails on a particular query.

Keywords: information systems; information retrieval; system effectiveness; search engine; IR system analysis; data analytics; query processing chain

MSC: 94A16; 68T20; 94A16

Citation: Mothe, J. Analytics Methods to Understand Information Retrieval Effectiveness—A Survey. *Mathematics* **2022**, *10*, 2135. https://doi.org/10.3390/math10122135

Academic Editors: Cornelia Caragea and Zhao Kang

Received: 31 January 2022
Accepted: 18 May 2022
Published: 19 June 2022
Corrected: 19 September 2022

1. Introduction

Information retrieval (IR) aims to retrieve the documents that answer users' queries. It is a core functionality of many digital systems, such as web search engines and e-commerce recommender systems. The effectiveness of an IR system is not the same on all the queries it processes, this is the *query factor* of effectiveness variability.

In IR, documents are indexed to build document representations that will be used for the online query-document matching. The online process used to answer a user's query consists of different phases and aims to choose the documents to deliver to the user as well as their order. A typical online process consists of the following phases: query pre-processing, optional automatic query reformulation or expansion, search for documents matching the query, and retrieved document ordering (see Figure 1).

A variety of components have been defined in the literature for each phase. For example, the searching/matching, also called weighting model (it is called this because in this component each document will receive a score with regard to a given query), can be achieved by the Salton's Vector Space Model [1], the Roberston and Spark Jones' BM25 probabilistic model [2,3], the Ponte and Croft' Language Modelling [4], and others.

Defining an information search process chain implies we decide which component will be used in each phase. In addition to the wide choice of possible components, each one has a set of hyper-parameters that need to be set. For example, the number of terms to add to the query in the automatic query expansion phase is one of the parameters to be decided. A query processing chain A will not result in the same retrieved documents, and, thus, not the same effectiveness, than a query processing chain B, even when considering

the same query or queries, the same collection of documents, and the same effectiveness measure. This variability in effectiveness due to the system is also named as the system factor of variability in effectiveness.

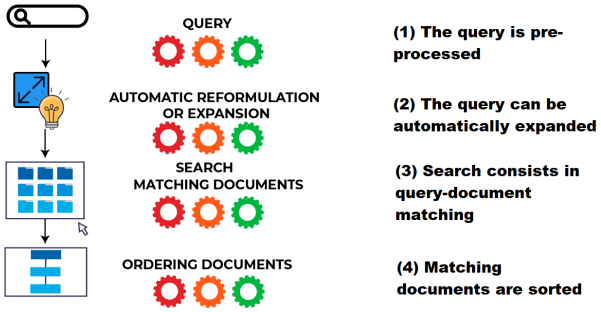


Figure 1. An online search process consists of four main phases to retrieve an ordered list of documents that answer the user's query. The component used at each phase has various hyper-parameters to tune.

Common practice in information retrieval is to tune the search process components' parameters once for all the forthcoming queries.

For example, Terrier, an open source search engine that implements state-of-the-art indexing and retrieval functionalities adapted to use on TREC-like collections [5], provides implementations of many weighting and query expansion models, along with default values of their hyper-parameters (http://terrier.org/docs/v4.0/configure_retrieval.html accessed on 15 May 2022). The optimal values of components hyper-parameters are obtained in an empiric way and different methods have been used for that, grid search is one of the most popular [6]. Optimal hyper-parameter values have been shown to be collection dependant.

Some studies have shown that the optimal hyper-parameters are not only collection dependent but query dependent as well. For example, considering the query expansion component, Ayter et al. reported that for the query 442 from the TREC7 reference collection, it is best to add 2 to 5 query terms to the initial query when expanding it, while for query 432, it is best to add 20 terms [7]. Note that evaluation forums distinguish between topics and queries. In evaluation frameworks, a topic consists of a title which is generally used as the query to submit to the search engine, a description of one or two sentences that explains the title, and a narrative that depicts what a relevant document is and what information will not be relevant. Because queries are usually formed from topic title, topic and query are often used interchangeably. The relationship between search components and system performance is worth studying in order to better understand the impact of the choice one makes when designing a query processing chain.

With regard to a better understanding of IR thanks to the system features, pioneer studies were based on the results from the participants at IR evaluation forums, such as TREC[5]. The strength of such evaluation campaigns is that a series of systems, each with different query processing chains, are using the same queries, the same document sets to search in, and the same evaluation measures. Because the results are obtained on the same basis, they are comparable and analyses are made possible [8–15]. The material which is used in such studies is the participants' results to shared tasks, that is to say from 30 to 100 systems (called *runs* in shared tasks). These studies suffer, however, from the lack of structured and easily exploitable descriptions of the query processing chains that the participants used. The components used in the indexing and query processing, as well as the values of their hyper-parameters, are described in papers and working notes both in verbose ways and with different levels of detail.

Further studies considered generated query processing chains [7,16,17]. Here, the data collections from shared tasks are also used, but rather than considering real participants' systems, a huge number of systems, from a few hundred to several thousands, are generated

thanks to some tools [18,19]. The generated systems differ by the components used or their hyper-parameters. Such chains have the advantage of being deterministic in the sense that the components they used are fully described and known, and, thus, allowed deeper analyses. The effect of individual components or hyper-parameters can be studied.

Finally, some studies developed models that aim at predicting the performance of a given system on a query for a document collection [20–22]. The predictive models can help in understanding the system effectiveness. If they are transparent, it is possible to know what the most important features are, what deep learning approaches seldom do.

The challenge for IR we are considering in this paper is:

- Can we understand better the IR system effectiveness, that is to say successes and failures of systems, using data analytics methods?

The sub-objectives targeted here are:

- Did the literature allow conclusions to be drawn from the analysis of international evaluation campaigns and the analysis of the participants' results?
- Did data driven analysis, based on thorough examination of IR components and hyper-parameters, lead to different or better conclusions?
- Did we learn from query performance prediction?

The more long-term challenge is:

- Can system effectiveness understanding be used in a comprehensive way in IR to solve system failures and to design more effective systems? Can we design a transparent model in terms of its performance on a query?

This paper reviews the literature of this domain. It covers analyses on the system factor, that is to say the effects of components and hyper-parameters on the system effectiveness. It also covers the query factor through studies that analyse the variability due to the queries. Cross-effects are also mentioned. This paper does not cover the question of relevance, although it is in essence related to the effectiveness calculation. It does not cover query performance prediction either.

Rather, this paper question the understanding we have of information retrieval thanks to data analytic methods and provides an overview on which methods have been used in relation to which angle of effectiveness understanding the studies focused on.

The rest of this paper is organised as follows: Section 2 presents the related work. Section 3 presents the material and methods. Section 4 reports on the results of analyses conducted on participants obtained at evaluation campaigns. Section 5 covers the system factor and analyses of results obtained with systematically generated query processing chains. Section 6 is about the analyses on the query effect and cross effects. Section 7 discusses the reported work in terms of its potential impact for IR and concludes this paper.

2. Related Work

To the best of our knowledge, there is no survey published on this specific challenge. Related work mainly consists of surveys that study a particular IR component. Other related studies are of relevance in IR, query difficulty and query performance prediction, and fairness and transparency in IR.

2.1. Surveys on a Specific IR Component

Probably because of its long-standing history in IR and the large number of techniques that have been developed, several surveys focused on the **query expansion component**. Carpineto and Romano' survey [23] includes the different applications of query expansion, as well as the different techniques. They suggested a classification of QE approaches that Azad and Deepak [24] completed with a four-level taxonomy. To analyse the different methods, Carpineto and Romano did not use any data analytics, rather they used both a classification with various criteria and a comparison of method effectiveness. More precisely, the criteria they used are as follows: the data source used in the expansion

(e.g., Wordnet, top ranked documents, ...), candidate feature extraction method, feature selection method, and the expanded query representation. With regard to effectiveness, they report mean average precision on TREC collections (sparse results). Mean average precision is the average of average precision on a query set. Average precision is one of the main evaluation measure in IR. It is the area under the precision–recall curve which, in practice, is replaced with an approximate based on precision at every position in the ranked sequence of documents, more at https://jonathan-hui.medium.com/map-mean-average-precision-for-object-detection-45c121a31173 accessed on 15 May 2022. The authors concluded that for query expansion, linguistics techniques are considered as less effective than statistic-based methods. In particular, local analysis seems to perform better than corpus based. The authors also mentioned that the methods seem to be complementary and that this should be exploited more. Their final conclusion is that the best choice depends on many factors among which the type of collection being queried, the availability and features of the external data, and the type of queries. The authors did not detail the link between these features and the choices of a query expansion mechanism.

Moral et al. [25] focuses on **stemming algorithms** applied in the indexing and query pre-processing and their effect. They considered mainly rule-based stemmers and classified the stemmers according to their features, such as their strength, the aggressiveness with which the stemmer clears the terminations of the terms, the number of rules and suffixes considered, their use of recoding phase, partial-matching, and constraint rules. They also compared the algorithms according to their conflation rate or index compression factor. The authors did not compare the algorithms in terms of effectiveness but rather refer to other papers for this aspect.

We can also mention the study by Kamphuis et al. in which they considered 8 variants of the **BM25 scoring function** [26]. The authors considered 3 TREC collections and used average precision at 30 documents. Precision at 30 documents is the precision, the proportion of relevant document within the retrieved document list where this list is considered up to the 30th retrieved document. They show that there is no significant effectiveness difference between the different implantation of BM25.

These analyses focus on a single component and do not analyse the results obtained strictly speaking but rather compare them using typical report means (mainly tables of effectiveness measures averaged over queries) as presented in Section 2.3.

2.2. Effectiveness and Relevance

System effectiveness is closely related to the notion of **relevance**.

Mizzaro [27] studied different kinds of relevance in IR, for which he defined several dimensions. He concluded that common practice to evaluate IR is to consider: (a) the surrogate, a representation of a document; (b) the query, the way the user expresses their perceive information need; and (c) the topic, that refers to the subject area the user is interested in. He also mentioned that this is the *lowest* level of relevance consideration in that it does not consider the real user's information need nor the perceived information need, nor the information the user creates or receives when reading a document. Ruthven [28] studied how various types of TREC data can be used to better understand relevance and found that factors, such as familiarity, interest, and strictness of relevance criteria, may affect the TREC relevance assessments.

Although relevance and the way relevance assessments are collected and considered can be a factor of IR system effectiveness, in this paper, we do not discuss the relevance point and consider effectiveness in its most common meaning in the IR field, as mentioned in [27].

2.3. Typical Evaluation Report in IR Literature

With regard to **hyper-parameters**, we should mention that it is a common practice nowadays in IR experimental evaluation (https://www.sigir.org/sigir2012/paper-guidelines.php accessed on 15 May 2022 is an example of paper guideline to write IR

papers.) to analyse the hyper-parameters of the method one developed. Analysing the results is generally performed by comparing the results in terms of effectiveness in tables or graphs that show the effectiveness for different values of the hyper-parameters (see Figure 2 that represent typical reports on comparison of methods and hyper-parameters in IR papers). In these figures and tables, the parameter values change and either different effectiveness measures or different evaluation collections, or both are reported.

The purpose here is to emphasise that, even if extensive experimental evaluation is reported in IR papers, the reports are mainly under the form or tables and curves, which are low level data analysis representations that we do not discuss in the rest of this paper.

Training Type	Encoder	L#	Batch Size	TREC-DL'19 nDCG@10	TREC-DL'19 MRR@10	TREC-DL'19 R@1K	TREC-DL'20 nDCG@10	TREC-DL'20 MRR@10	TREC-DL'20 R@1K	MSMARCO DEV nDCG@10	MSMARCO DEV MRR@10	MSMARCO DEV R@1K
BM25	–	–	–	.501	.689	.739	.475	.649	.806	.241	.194	.868
ANCE	BERT-Base	12	32	.648	–	–	–	–	–	–	.330	.959
LTRe				.661	–	–	–	–	–	–	.329	.955
ANCE + LTRe				.675	–	–	–	–	–	–	.341	.962
RocketQA	ERNIE-Base	12	4,000	–	–	–	–	–	–	–	**.364**	–
			128	–	–	–	–	–	–	–	.309	–
TCT	BERT-Base	12	96	.670	–	.720	–	–	–	–	.335	.964
TCT (ours)	DistilBERT	6	32	$.680^{b}$	$.857^{b}$	.745	$.631^{b}$	$.773^{b}$	.792	$.372^{b}$	$.315^{b}$	$.951^{b}$
Margin-MSE	DistilBERT	6	32	.697	.868	.769	–	–	–	.381	.323	.957
Margin-MSE (ours)				$.687^{b}$	$.851^{b}$	.767	$.654^{b}$	$.812^{b}$	.801	$.385^{bt}$	$.326^{bt}$	$.958^{bt}$
TAS-Balanced	DistilBERT	6	32	$.712^{b}$	$.892^{b}$	$\mathbf{.845}^{btm}$	$\mathbf{.693}^{btm}$	$\mathbf{.843}^{b}$	$.865^{btm}$	$.402^{btm}$	$.340^{btm}$	$.975^{btm}$
			96	$\mathbf{.722}^{btm}$	$\mathbf{.895}^{b}$	$.842^{tm}$	$.692^{btm}$	$.841^{b}$	$.864^{btm}$	$.406^{btm}$	$.343^{btm}$	$.976^{btm}$
			256	$.717^{btm}$	$.883^{b}$	$.843^{tm}$	$.686^{btm}$	$.843^{b}$	$\mathbf{.875}^{btm}$	$\mathbf{.410}^{btm39}$	$.347^{btm39}$	$\mathbf{.978}^{btm3}$

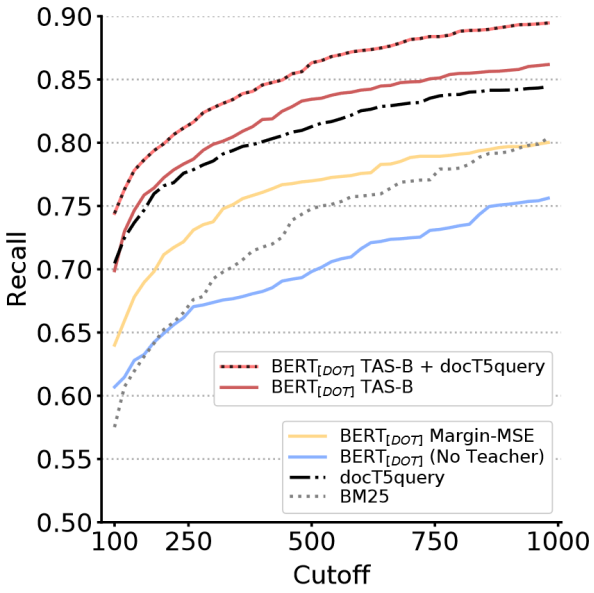


Figure 2. A common practice in IR literature is to analyse the effect of hyper-parameters on the overall system effectiveness and to present the results under the form of tables or graphs. The top part of this figure is a typical table that represents hyper-parameters or comparison of methods. Here, a deep learning-based model was used and comparisons are reported on the different training types, encoders, and batch sizes; using different effectiveness measures (nDCG@10, MRR@10, and R@1K), on different collections (here TREC DL'19, TREC-DL'20, and MSMARCO DEV). The best results are highlighted in bold font. the bottom part is a typical graph to compare different variants or hyper-parameters effect on effectiveness. Here, the lines represent different combination of hyper-parameters, effectiveness is measured in terms of recall (Y-axis) for different cut-off of the retrieved document list. Table and Figure adapted with permission from [29], Copyright 2021, Sebastian Hofstätter et al.

3. Materials and Methods

The analysis of effectiveness for better comprehensive understanding of IR relies on data analysis methods and analysable data that we describe in this section.

3.1. Data Analysis Methods

System effectiveness analyses rely on different statistical analysis methods, including, but not limited to, machine learning.

Boxplot is a graphical representation of a series of numerical values that shows their locality, spread, and skewness based on their quartiles. Whiskers extend the Q1–Q3 box, indicating variability outside the upper and lower quartiles. Beyond the whiskers, outliers that differ significantly from the rest of the dataset are plotted as individual points. Effectiveness under different conditions (different queries, different values of a component parameter) is a typical series that can be represented under the form of a boxplot.

Correlation is a family of analysis that measures the relationship between two variables, its strength and direction. Correlation calculation results in a value that ranges between -1 (strong negative correlation) and 1 (strong positive correlation); 0 indicating that the two variables are not correlated. The p-value indicates the confidence or risk of error in rejecting the hypothesis that the two variables are independent. The most familiar measure of correlation is the Pearson product-moment correlation coefficient which is a normalised form of the covariance. Covariance between two random variables measures their join distance to their expected values which can be the distance to the mean for numerical data. Pearson ρ assumes linear relationship between the two variables. Spearman's correlation (r) considers the ranks rather than the values and measures how far from each other variable ranks are. r is similar to Pearson on ranks. Spearman's assumes monotonic relationship between the two variables. Kendall correlation measures the correlation on ranks, that is the similarity of the orderings of data when ranked by each of the variable values. It is affected by whether the ranks between observations are the same or not without considering how far they are as opposed to r. It is thus considered as more appropriate for discrete variables. With regard to system effectiveness, correlation is used in query performance prediction to evaluate the accuracy of the prediction: the two analysed variables are the predictor (either a single predictor or a complex one) and the observed effectiveness.

Analysis of variance (ANOVA) encompasses different statistical models and estimation procedures used to highlight differences or dependencies between several statistical groups. It is used to analyse the difference between the means of more than two groups. In ANOVA, the observed variance in a particular variable is partitioned into components that are attributable to different sources of variation. A one-way ANOVA uses one independent variable, while a two-way ANOVA uses two independent variables. The General Linear Mixed Model [30] extends the General Linear Model [31] so that the linear predictor contains random effects in addition to the usual fixed effects.

Factorial analysis is used to describe variability among observed, correlated variables; it uses factors, here combinations of initial variables, to represent individuals or data in a space of a lower dimension. It uses singular value decomposition and is appropriate to visualise the link between elements (individuals) that are initially represented in a high dimensional space (variables). Two variants of factorial analysis are used in the context of IR system performance analysis. Factorial analysis is also the core model used in the Latent Semantic Indexing model [32] where documents are considered in the high dimensional space of words. It is also linked to the matrix factorisation principle used in recommender systems for example [33]. Principal Component Analysis (PCA) and Correspondence Analysis [34] which differ on the pre-treatment applied to the initial analysed matrix and on the distance used to find the links between variables and individuals. Although PCA reduces the dimensionality of the data by considering the most important dimensions as determined by the eigen values of the variance/covariance matrix using Euclidian distance, CA uses the χ^2 distance on contingency matrices. Factorial analysis results on visual representations which can be manually interpreted. Among others, one interesting

property of CA compared to PCA is that individuals and features can be observed all together in the same projected space. Factorial analyses are used in the context of IR effectiveness analysis.

Clustering methods is a family of methods that aims to group together similar objects or individuals. Under this group falls agglomerative clustering and k-means. In agglomerative clustering, each individual corresponds to a cluster; at each processing step, the two closest clusters are merged; the process ends when there is a single cluster. The minimum value of the error sum of squares is used as the ward criterion to choose the pair of clusters to merge [35]. The resulting dendogramme (tree-like structure) can be cut at any level to produce a partition of objects. Depending on its level, the cut will result in either numerous but homogeneously-composed clusters or few but heterogeneously-composed clusters. Another popular clustering method is k-means where a number of seeds, corresponding to the desired number of clusters, are chosen. Objects are associated to the closest seed. Objects can then be re-allocated to a different cluster if it is closer to the centroid of an other cluster. For system effectiveness analysis, clustering can be used to group queries, systems or even measures.

Although the previous methods are usually considered as descriptive ones, the two other groups of methods are predictive methods. That means they are used to predict either a class (e.g., for a qualitative variable) or a value (e.g., for a continuous variable).

Regression methods aim to approach the value of a dependent variable (the variable to be predicted) considering one or several independent variables (the variables or features that are used to predict). The regression is based on a function model with one or more parameters (e.g., linear function in the linear regression; polynomial, ...). Logistic regression is for the case the variable to explain is binary (e.g., the individual belongs to a class or not). It is used, for example, in query performance prediction.

Decision trees show a family of non-parametric supervised learning methods that are used for classification and regression. The resulting model is able to predict the value of a target variable by learning simple decision rules inferred from the data features. CART [36] and random forests [37] are the most popular among these methods. They have been shown as very competitive methods. The extra advantage is that the model can combine both quantitative and qualitative variables. In addition, the obtained models are explainable. For system effectiveness analysis, the target variable is effectiveness measurement or class of query difficulty (easy, hard, medium, for example). The system hyperparameters or query features are used to infer the rules.

In this study, we do not consider **deep learning** methods as means to analyse and understand information retrieval effectiveness. Deep learning is more and more popular in IR but still these models lack interpretability. The artificial intelligence community is re-investigating the explainability and interpretability challenge of neural network based models [38]. For example, a recent review focused on explainable recommendation systems [39]. Still, model explanability is mainly based on model interpretability and prominent interpretable models are more conventional machine learning ones, such as regression models and decision tree models [39].

3.2. Data and Data Structures for System Effectiveness Analysis

There are different international challenges in IR where participants use the same data collections to answer shared tasks and, thus, that can be used to deeply analyse system effectiveness, its factors and parameters. In this paper, the studied papers focused on the pioneering TREC challenge. TREC considered many different languages, but when it began and nowadays it is mainly focused on English. TREC encompasses various tasks; the most popular and running from the largest number of years is ad hoc retrieval where the task is to retrieve the relevant documents, given a query. It was also the first and unique task introduced in TREC in 1992 [40]. This paper focuses on ad hoc retrieval.

System performance analyses (presented in Sections 4–6) share the same type of data structures, namely matrices.

In general, the participants' results consists in measurements across three dimensions (system, topic, measure). As a result of a challenge like TREC, we can thus build 3D matrices (see Figure 3) that report values for different systems, different topics, different effectiveness measures.

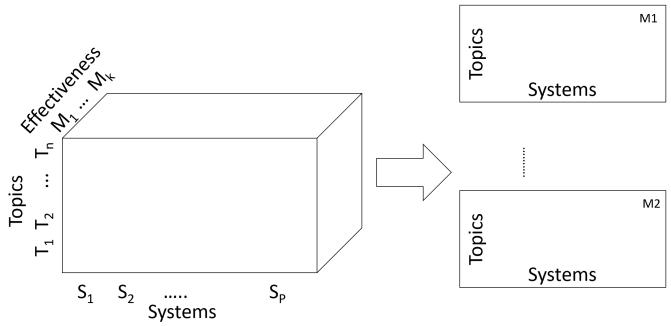


Figure 3. The 3D matrices obtained from participants' results to shared ad hoc information retrieval tasks that report effectiveness measurements for systems, topics, effectiveness measures can be transformed into 2D matrices that fit many data analysis methods.

Such a 3D matrix can be transformed into a 2D one for a given effectiveness measure where the two remaining dimensions are the systems and the topics. The resulting matrix can then be used as an input to many of the data analysis methods we presented in the previous sub-section where individuals are the systems represented according to the topics, or using the transposed matrix, individuals are topics represented according to the systems.

We can also have more information at our disposal on systems or on topics or on both. In that case, the data structures can be more complex. For example, if we consider a given effectiveness measure, systems can be represented by different features (e.g., the components that are used, their hyperparameters, ...). In the same way, topics can come with various features (e.g., linguistic or statistical features, ...) (see Figure 4).

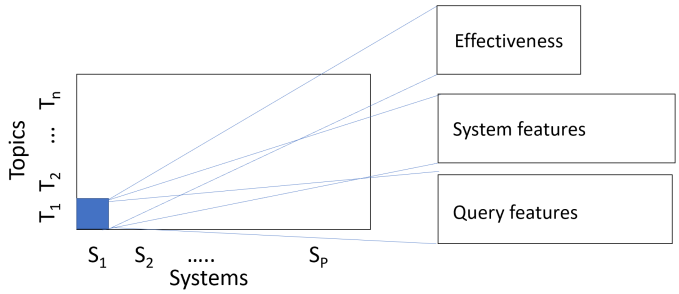


Figure 4. More complex data structures can be used that integrate features on topics, on systems or on both.

Finally, some studies consider aggregated values. For example, rather than considering each query individually, we can consider aggregated value across queries; this is commonly used to compare systems and methods at a upper level. On the other hand, it is possible not to consider each system individually but aggregate the results across systems (see Figure 5).

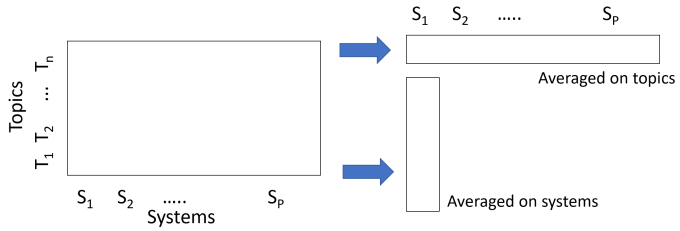


Figure 5. Aggregated values can be considered, either on the queries or systems.

4. System Performance Analysis Based on Their Participation to Evaluation Challenges

International evaluation forums such as TREC. TREC started in 1992 and supports research in the information retrieval and provides the framework for large-scale evaluation of information retrieval have provided shared collections consisting of queries, documents, and relevance judgements, but also consisting of participants' results. For a given collection, we have different systems that ran the queries over the same document set and are evaluated the same way. Indeed, each participant to TREC tracks receives a detailed report where their runs or systems are scored according to various effectiveness measures obtained by the open evaluation tool trec_eval described in https://github.com/topics/trec-eval accessed on 15 May 2022. This opened the opportunity to mine these results.

The first to analyse the results were the organisers. In the first TREC report, the results of the different participants were compared mainly through recall precision curves [41].

Additional analyses were made a few years later. One search engine (SMART system of Cornell University) which has been one of the most effective in TREC ad hoc task, was run in its versions used in each of the first seven TREC conferences on each of the first seven ad hoc test collections [42]. This longitudinal analysis of a system version shows that effectiveness increases over time, whatever the system variant used.

The reliable information access workshop was dedicated to the analysis of the participants' results. The organisers reported that a system can perform very well on a query A, very bad on a query B, while an other system will do the opposite [12]. At that time, it was not possible to understand the reasons of the variability in results; it was stated as being dependant on three factors: the query, the relationship between the query and the documents, and the system parameters; it was considered as a difficult problem to understand [14] and this difficulty remains.

Tague-Sutcliffe and Blustein [8] also reported such variability and showed that the variance due to queries was much greater than the one due to systems. This finding has encouraged the further study on the link between queries and system performance.

Banks et al. [9] considered a matrix in which rows and columns represent systems and topics/queries, and cells correspond to average precision (like the one presented in Figure 6). They also considered the retrieved document lists of each participant. Then, they applied six data analysis methods, trying to extract knowledge, relationships, clusters, ...from these data that would help to understand better the structure or derive general conclusions from the participants' results. Among them, the authors considered the analysis of variance to look for deviations and variability in retrieval performance that could be explained by the systems, topics, or both. They also tried to extract clusters of topics and systems. On document lists, they analysed the correlations on document orders and tried to extract the sub-patterns in the sequence of retrieved documents. The authors concluded "None of the work we have done using the six approaches discussed has provided the sort of insights we were seeking, and the prospects that additional work along these lines will yield significantly better results vary but are not generally promising." [9].

	APL985L	APL985LC	APL985SC	AntHoc01	Brkly24	Brkly25
T351	0.2257	0.2261	0.1655	0.2933	0.2987	0.3137
T352	0.0229	0.0321	0.0594	0.0277	0.0379	0.0097
T353	0.3271	0.3052	0.2852	0.2091	0.374	0.264
T354	0.1119	0.1496	0.0908	0.0139	0.0192	0.1084
T355	0.0973	0.0688	0.0327	0.1365	0.0987	0.183
T356	0.052	0.0593	0.0462	0.0091	0.0128	0.0452
T357	0.1358	0.1803	0.1391	0.0984	0,3284	0.3277
T358	0.0994	0.0988	0.0489	0.1514	0,2078	0.3887
T359	0.0378	0.0337	0.0146	0.0223	0,0319	0.0357
T360	0.39	0.3825	0.4096	0.0404	0,3275	0.036

Figure 6. A 2D matrix representing the effectiveness of different systems (X axis) on different topics (Y axis). This matrix is an extract of the one representing the AP (effectiveness measure) for TREC 7 ad hoc participants on the topic set of that track.

Analyses were produced on web track overviews. On Web track 2009 [43], the organisers reported the plot representing the effectiveness of participants' system considering two evaluation measures, the mean subtopic recall and the mean precision (see Figure 7). This analysis showed that the two measures correlate, which means that a system A that is better than a system B on one of the two measures is also better when considering the second measure. When effective systems are effective, the measure that is used does not matter.

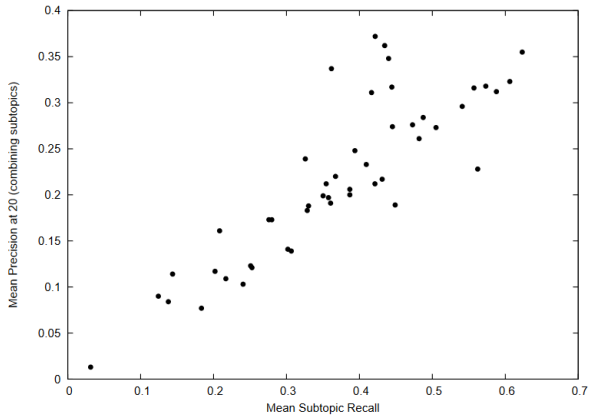


Figure 7. Effective systems are effective whatever the measure used. Web track 09 participants' results considering mean subtopic recall (X-axis) and mean precision (Y-axis); each dot is a participant systems. Figure reprinted with permission from [43], Copyright 2009, Charles Clarke et al.

In web track 2014 [44], the authors provided a different type of analysis with box plots that show the dispersion of the effectiveness measurements for each of the topics, across participants' systems (see Figure 8). This type of view informs on the impact of the system factor on the results. The smaller the box, the smaller the importance of the system factor is. Both some easy queries, for which the median effectiveness is high, and hard queries, for which the median effectiveness is low, have packed results (e.g., easy 285 topic in Figure 8 and hard 255 topic—not presented here). Both types have also dispersed results (e.g., easy 285 topic on Figure 8 and hard 269 topic—not presented here).

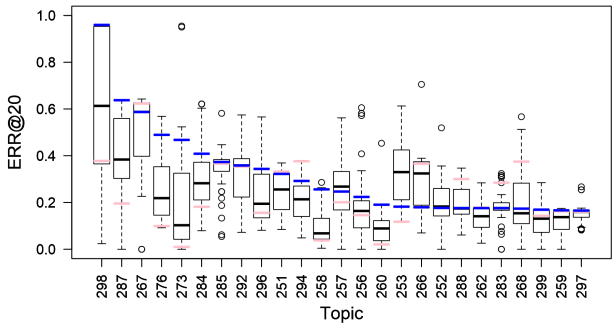


Figure 8. In the easiest topics according to the median effectiveness on the participants' results, there are both topics with very diverse system effectiveness results (e.g., 298) and very similar ones (e.g., topic 285)—Web track 2014—topics are ordered according to decreasing the err@20 of the best system. Figure reprinted with permission from [44], Copyright 2014, Kevyn Collins-Thompson et al.

On the same type of matrices than the ones Banks et al. used (see Figure 6), Dinçer et al. [10], Mothe et al. [11], and Bigot et al. [15] applied factorial analyses more successfully. These studies showed that topics and systems are indeed linked. Principal Component Analysis (PCA) and Correspondence Analysis were used.

On Figure 9 we can see PCA applied on a matrix that reports average precision for different queries (variables, matrix columns) by different systems (individuals, matrix rows) at TREC 12 Robust track. We can see on the left bottom part systems that behave similarly on the same queries (they fail on the same queries, succeed on the same queries) and that behave differently from the other systems. We can see another group of systems on the top left corner of the figure. Similar results where reported in [11] where PCA was applied on a matrix that studied recall at TREC Novelty track. In both studies, the results showed that there is not just two groups of systems, thus emphasising that systems behave differently on different queries but that some systems have similar profiles (behave similarly on the same queries).

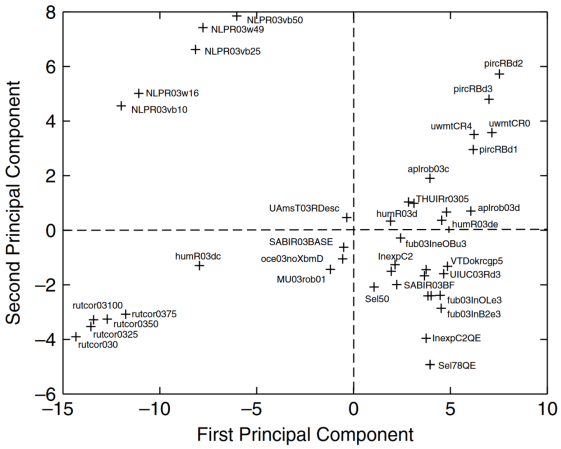


Figure 9. System failure and effectiveness depend on queries—not all systems succeed or fail on the same queries. The visualisation shows the two first principal components of a Principal Component Analysis, where the data of the system effectiveness is obtained for each topic by each participants' run. MAP measure of TREC 12 Robust Track participants' runs. Figure reprinted with permission from [10], Copyright 2007, John Wiley and Sons.

These results are complementary from Mizzaro and Roberston' findings which show that "easy" queries, the ones that on average systems can answer pretty well, are best to distinguish good systems from bad systems [13].

Kompaore et al. [45] showed that queries can be clustered in a more sophisticated way, based on their linguistic features. The authors extracted 13 linguistic features that they used

to cluster queries—based on agglomerative clustering—and obtained three query clusters. Then, they ranked the TREC ad hoc task participants according to the result they obtained on mean average precision over the entire set of queries, and over each of the clusters. They showed that for each query cluster the best system is not the same (see Figure 10). For example, while ETHme1 was ranked first when considering the mean average precision on the entire set of queries and the query cluster 1, it is ranked the 10th on the query cluster 2. For that query cluster, the best performing system is uwgcx0, and it is LNaDesc1 for query cluster 3. This shows that there is also system profiles that can be extracted, considering topic difficulty levels.

Although these studies analysed TREC participants' systems, other studies have generated different combinations of search components and hyperparameters [15,16,18,45–47] and thus went a step further in understanding the factors of variability.

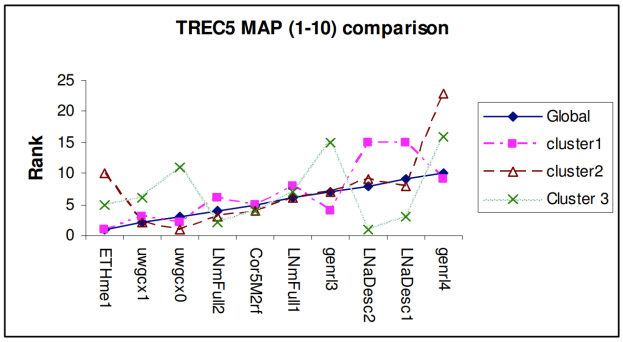


Figure 10. The first ranked system differs according to the query clusters. The rank of the system is on the Y-axis and the system is on X-axis. Blue diamonds correspond to the ranks of the systems when considering all the queries, pink squares when considering the query cluster 1, brown triangles are for query cluster 2, and green crosses for cluster 3. Systems on the X-axis are ordered according to decreasing effectiveness on average on the query set. Figure reprinted with permission from [45], Copyright 2007, Kompaore et al.

5. Analyses Based on Systems That Were Generated for the Study—The System Factor

The system factor is the factor that has been mentioned the first in shared tasks: systems do not perform identically. Thanks to the results the participants' system obtained in shared tasks, it has been possible to identify which techniques or systems work better in average over queries, but because the description of those systems was not enough, it has not been possible to study the important factors within the systems. This is what some studies aimed to analyse.

Two research groups deeply studied this specific point: the Information Management Systems Research Group of the University of Padua in Italy, starting from 2011 [48] and the Information System Research Group of the University of Toulouse in France, starting from 2011 [49]. Google Scholar was used to find what were the first pieces of work related to automatically generated IR chains in the objective to analyse the component effects. Although the two cited works did not obtain many citations, they mark the starting point of this new research track.

The analyses based on synthetic data are in line with the idea developed in 2010 in [46] for an evaluation framework where components would be run locally and where intermediate output would be upload so that component effects could be analysed deeper; evaluation as a service has further developed the same idea [50].

One of the first implementations of the automatic generation of a large series of query processing chains was the one in Louedec et al. [18]; in line with the ideas in [46,51] also implemented in [19]. It was made possible because of the valuable work that has been performed in Glasgow on Terrier [5] to implement IR components from the literature. Other platforms can also serve this purpose, such as Lemur/Indri (https://www.lemurproject.org/ accessed on 15 May 2022); https://sourceforge.net/p/lemur/wiki/

Indri%20Retrieval%20Model/ accessed on 15 May 2022 although more centred on language models or Cherche (https://github.com/raphaelsty/cherche accessed on 15 May 2022) for neural models.

Compared to using participants' systems, generated query processing chains gives the ability to know the exact components and hyper-parameters used and thus make deeper analysis possible.

One of the pioneer studies that analysed a huge number of automatically generated systems is Ayter et al. [7]. They used about 80,000 combinations of components in query processing chains with: 4 stemmers, 7 weighting models, 6 query processing, 7 query expansion models, and various numbers of query terms and documents to consider in query expansion. They used TREC 7 and 8 ad hoc collection (100 topics in total querying the same document collection) and average precision as the effectiveness measure. Among their findings, the authors concluded that the choice of the stemmer component had little to no influence, while the weighting model had an impact on the results (see Figure 11). Other findings were that dirichletLM is the weaker search model among the 7 studied, while BB2 is among the best; this when considering also all the other parameters. Their analyses also confirmed that systems behave differently and that the choice of the components at each phase of the retrieving process, as well as the component hyper-parameters, are an important part of system successes and failures.

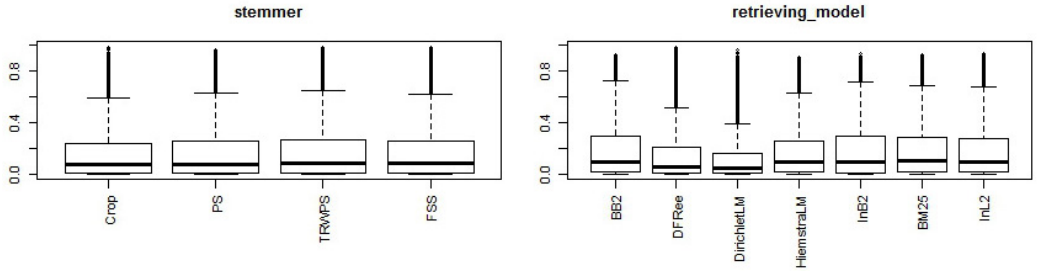


Figure 11. The choice of the weighting model has more impact than the stemmer used. Individual boxplots represent average precision on the TREC 7 and 8 topics when a given component is used in a query processing chain—80,000 query processing chains or component combinations were used. Figure reprinted with permission from [7], Copyright 2015, J.UCS.

Another important study is the one from Ferro and Silvello [16] followed up by [17] from the same authors. On TREC 8, they considered the cross effect of the component choice in the query processing chain. They considered three components: the stop list (5 variants), the stemmer (5 variants), and the weighting model (16 variants), for a total of 400 possible different combinations. As an effectiveness measure, they also used average precision. They show that variants of the stop word list used during indexing does not have a huge impact, but the system should use one (see Figure 12, subfigures A, B where the blue—no stopword—curve is systematically below the others and C where the starting point of the curve—no stopword—is lower than the rest of the curves). It also showed that, given a stopword list is used, the weighing model has the strongest impact among the three studied components (subfigures B and D, where waves show that the systems have different effectiveness).

Additionally, related to these studies, CLAIRE [52] is a system to explore IR results. When analysing TREC ad hoc and web tracks, the findings are consistent with previous analyses: dirichletLM is the weaker weighting model among the studied ones, IR weighting models suffer from the absence of a stop list and of stemmer. By such exploration, the authors were able to show which is the most promising combination of components for a system on a collection (e.g., jskls model equipped with a snowball stop list and a porter stemmer).

These data analytics studies allowed to understand the influence of the components and their possible cross-effect. They are using the results obtained on different collection; they do not aim to predict results.

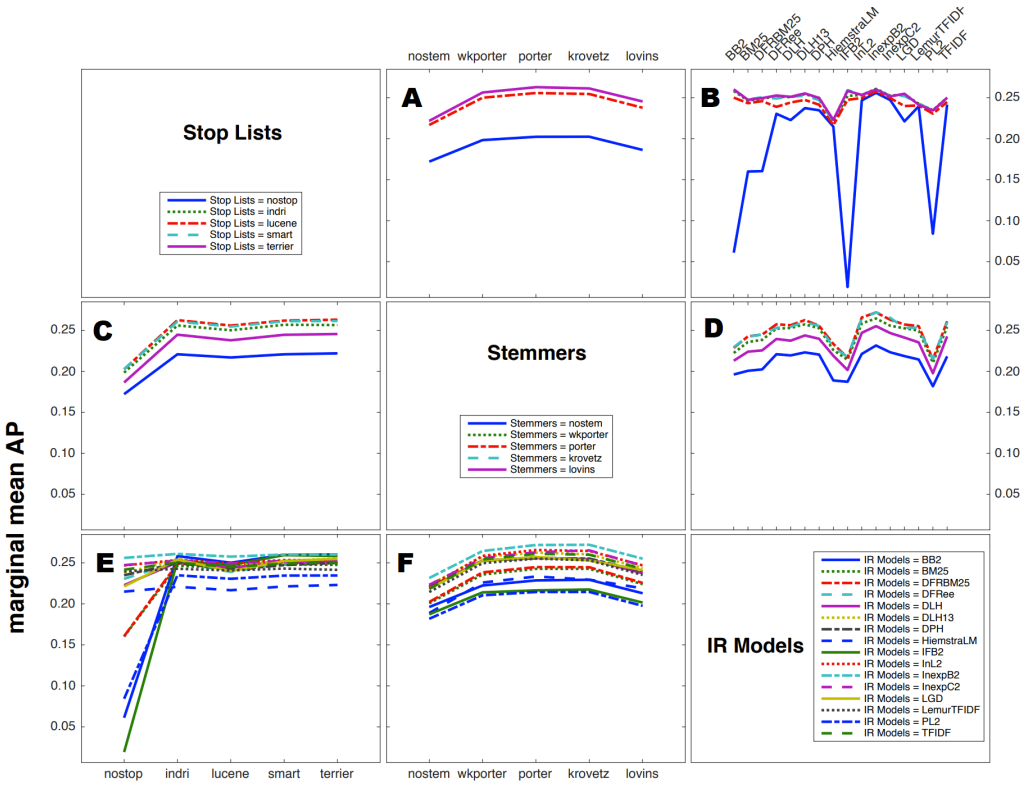


Figure 12. Interaction between component choices. The curves used in this representation are somehow misleading since the variables are not continuous but nevertheless can be understood, we thus kept the original Figures from [16]) where we added letters in each sub-figure for clarity. On the first row, the stop list effect is shown for different stemmers (**A**) and different weighting models (**B**). On the second row, the effect of stemmers is reported for different stop lists (**C**) and different weighting models (**D**). On the latest row, the weighting model effect is reported, for different stop lists (**E**) and different stemmers (**F**). Figure adapted with permission from [16], Copyright 2016, Nicola Ferro et al.

6. The Query Factor

6.1. Considering the Queries and Their Pre- and Post-Retrieval Features

The query is the main factor that explains variability in effectiveness [7,8,13,53]. The query factor, thus, attracted much research attention mainly under the query difficulty or query performance prediction research topics. Because this paper focuses on how data analytics can help IR effectiveness understanding, we do not survey query performance predictors but rather what we have learnt from the studies on the different types of query features used for effectiveness prediction.

The explanation of system effectiveness failure, which is strongly related to query difficulty in IR terminology, by the query it-self was studied considering different types of features:

- Linguistic features extracted from the query only [21,54–56];
- Other pre-retrieval features that use information on the document collection [57–61];
- Post retrieval features that consider the retrieved documents for that query [22,57,62–73].

Mothe and Tanguy [21] considered 16 linguistic query features. They observed the correlation of these features with average recall and precision obtained by TREC participant systems. Morphological, syntactical and semantical features were considered. Syntactic

links span (distance between syntactically linked words) was shown as inversely correlated to precision while polysemy value, the number of semantic classes a given word belongs to in WordNet was shown to be inversely correlated to recall.

Molina [74] developed a resource where 258 features have been extracted from queries from Robust, WT10G and GOV2 collections. Among the features extracted from the query only, are aggregations (minimum, maximum, mean, quartiles, standard deviation, and total) over query terms on the number of synonyms, hyponyms, meronyms and sister terms from WordNet. The authors did not provide a deep analysis on how much each of these features correlate with the system performance.

Hauff et al. [60] surveyed 22 pre-retrieval features from the literature at that time, from which some use the query only to be computed. The authors categorised these features into four groups, depending on the underlying heuristic: specificity, ambiguity, term relatedness, and ranking sensitivity. Intuitively, the higher the term specificity the easier the query. Inversely, the higher the term ambiguity, the more difficult the query is. When analysed on three TREC collections, the authors found weak correlation between the system performance and features based on the query only. Among them, the features related to the term ambiguity where the most correlated to system performance, in line with [21]. Features that consider information on the documents were found more correlated to performance than the ones based on the query only.

The query effect was also studied in link with the document collection that is searched by considering information resulting from the document indexing. These query features are grounded on the same idea as term weighting is for indexation: terms are not equivalent and their frequency in documents matters. Inverse document frequency, based on the number of documents in which the term occurs has specifically been used, but other features were also developed [59,61,75].

Finally, the query effect was also studied considering post-retrieval features [57,62–67,69–73]. Post-retrieval predictors are categorised into clarity-based, score-based, and robustness-based approaches [22,68]. They imply that a first document retrieval is performed using the query before the features can be calculated. Post-retrieval features mainly used the document scores.

Considered individually, post-retrieval features have been consistently shown as better predictors than pre-retrieval ones. It appeared, however, that an individual feature, either pre- or post-retrieval, is not enough to predict whether the system is going to fail or not or to predict its effectiveness. That mean that none of these individual features "explained" the system performance.

Indeed, many studies have reported weak correlation values for individual features [60,71,76] with the actual system effectiveness. When considering a single feature, the correlation values differ from one collection to another and from one feature to another. Moreover, they are weak. For example, Hauff et al. [60] report 396 correlation values among which 13 only are over 0.5. Hashemi et al. [77] reported 216 correlation values including the ones obtained with a new neural network-based predictor, with a maximum value of 0.58, a median of 0.17. Chifu et al. [71] reported 312 values, none of which above 0.50. In the same way, Khodabakhsh and Bagheri report 952 correlation values, none of which are above 0.46 [73]. When correlation are low it is even likelier that there is either very weak or no correlation at all between the predicted value (here effectiveness) and the feature used to predict. Table 1 and Figure 13 illustrate this. For this illustration, we took IDF_{Max} and IDF_{AVG} which are considered as the best pre-retrieval features [60,69,78], as well as BM25, a post-retrieval feature. We can see that with a correlation of 0.29 for BM25 or 0.24 for IDF (see Table 1), there is no correlation between the two variables as depicted on the scatter plots (see Figure 13).

Papers on query performance prediction seldom plot the predicted and actual values which is however an appropriate mean to check whether the correlation exists or not. As a counter example of this, we should here recall that the Anscombe's quartet [79] effect on the Pearson correlation illustrates that even a Pearson correlation up to 0.816 can be obtained with no correlation between the two studied variables (see Figure 14).

Table 1. Correlation between query features and ndcg. WT10G TREC collection. * marks the usual <0.05 *p*-Value significance.

	Feature			
Measure	**BM25_MAX**	**BM25_STD**	**IDF_MAX**	**IDF_AVG**
Pearson ρ	0.294 *	0.232 *	0.095	0.127
p-Value	0.0034	0.0224	0.3531	0.2125
Spearman *r*	0.260 *	0.348 *	0.236 *	0.196
p-Value	0.0100	<0.001	0.0202	0.0544
Kendall τ	0.172 *	0.230 *	0.159 *	0.136 *
p-Value	0.0128	<0.001	0.0215	0.0485

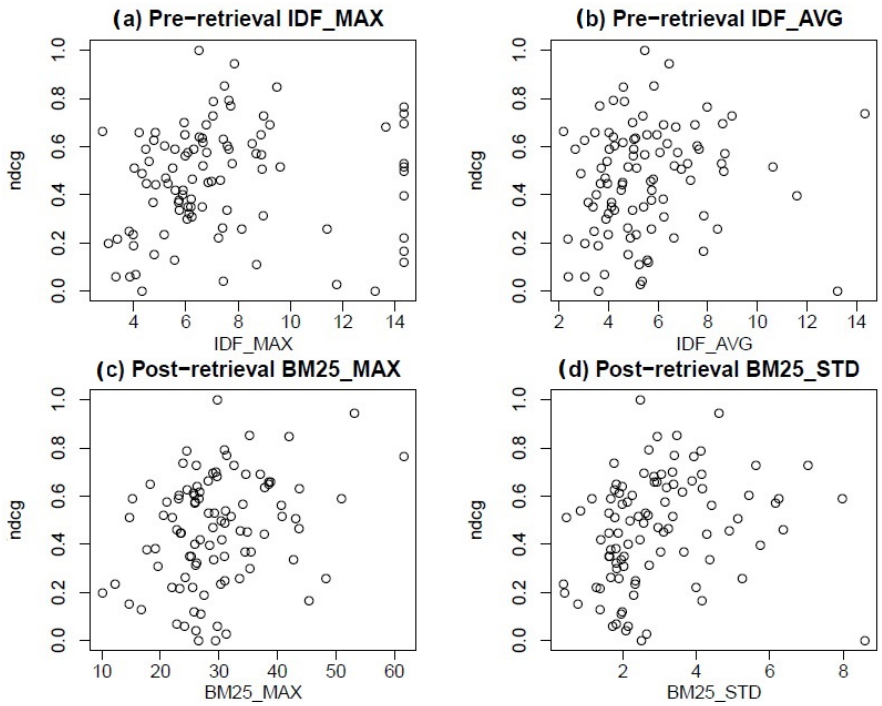


Figure 13. No correlation using pre- or post-predictors with the actual effectiveness—IDF pre-retrieval predictor and BM25 post-retrieval predictor (X-axis) and ndcg (Y-axis) values on WT10G TREC collection. Although the correlation values are up to 0.35, there is no correlation.

A single query feature cannot explain a single system effectiveness, but:

1. Combination of query features might;
2. It may explain that systems will fail in general.

Regarding (1), a series of studies have been produced to combine query features into models [78,80–82]. Grivolla et al. [80] grouped queries according to the predicted performance. They trained a model that combined various features, including linguistic, pre- and post- retrieval features, and used decision tree and SVM. The experiments they reported on TREC-8 participants' results showed that the model was not robust across systems: for some of them the prediction was accurate while it was not for some others. Raiber et al. combined various features in a Markov Random Fields model and reported a maximum correlation of 0.695 and a median of 0.32 [81]. Chifu et al. [71] combined Letor features using a linear model and reported a maximum correlation of 0.45. The only study we found that both reported positive results and plotted predicted vs actual effectiveness is Roy et al. [78]. They proposed a linear combination of a word embedding based pre-retrieval feature which measures the ambiguity of each query term, with the post-retrieval NQC feature (see Figure 15). However, the model performs well for easy

queries only. However, we know from Mizzaro and Robertson [13] that easy queries do not help to distinguish effective systems from non-effective ones [13].

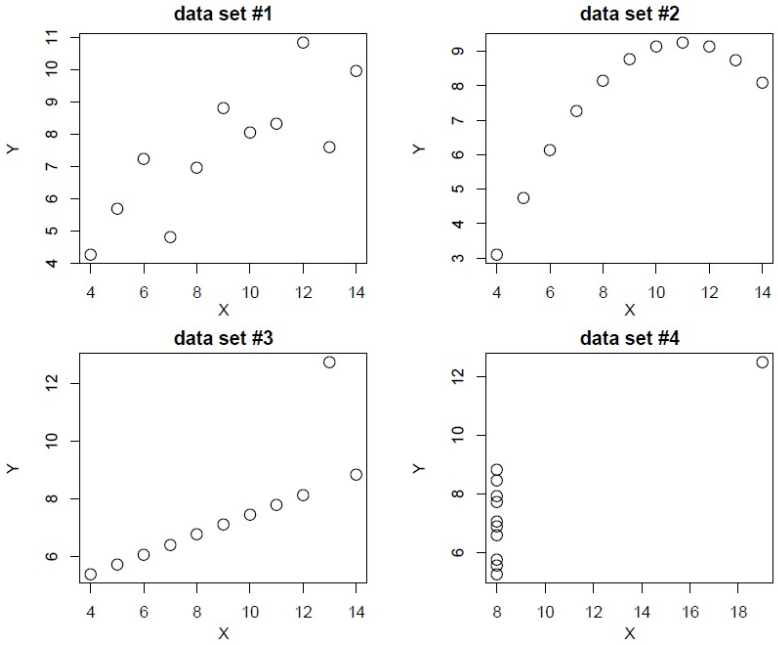


Figure 14. Pearson correlation value higher than 0.8 does not mean the two variables are correlated. The Anscombe's quartet presents four datasets that share the same mean, same number of values, same Pearson correlation value (ρ = 0.816) but for which this latter value does not always means the two variables X and Y are correlated. X and Y are not correlated on #4 despite high ρ value. #2 X and Y are perfectly correlated but not in a linear way (Pearson cannot measure other than linear correlations) #1 and #3 illustrates two cases of linear correlation. Figures generated from the data in [79].

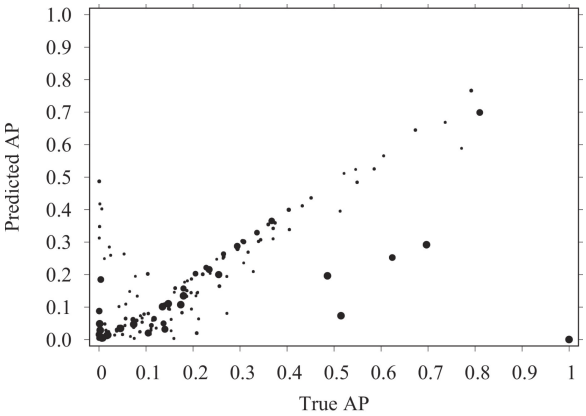
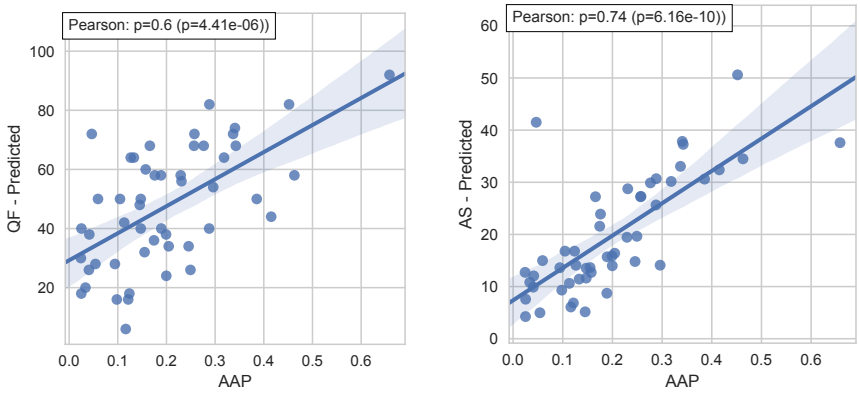


Figure 15. Predicted AP is correlated to actual AP for easy queries (the ones on the right part of the plot), although there are sparse. Figure reprinted with permission from Roy et al. [78], Copyright 2019, Elsevier.

With regard to (2), rather than considering individual system performance, Mizzaro et al. [82] focused on the average of average precision values over systems, this to detect the queries for which systems will fail in general. The authors showed that the correlation is more obvious between the predictor and the average system effectiveness than it was in other studies between the predictor and a single system (see Figure 16).

This call also for the need to try to understand the relationship between the query factor and the system factor.



(**a**) (**b**)

Figure 16. AS feature [83] is correlated to the average effectiveness of a set of systems. TREC7 Adhoc collection. Pearson correlation between AAP and (**a**) QF [66], (**b**) AS [83]. Dots correspond to actual and predicted AAP for individual topics; the cones represent the confidence intervals. Figure reprinted with permission from [82], Copyright 2018, Mizzaro et al.

6.2. Relationship between the Query Factor and the System Factor

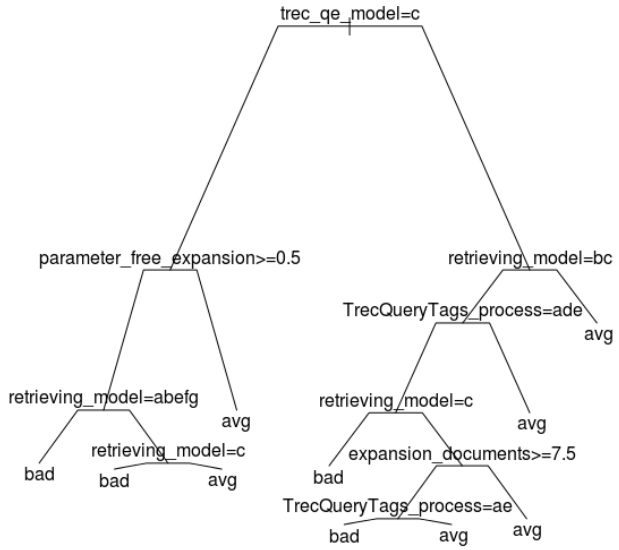
In Section 5, we reported studies on system variabilities and the effect of the components used at each search phase on the results when averaged over query sets, not considering deeply the query effect.

Here, we consider studies that were conducted at a finer grain. These pieces of work tried to understand the very link between the level of query difficulty (or level of system effectiveness) and the system components or hyperparameters.

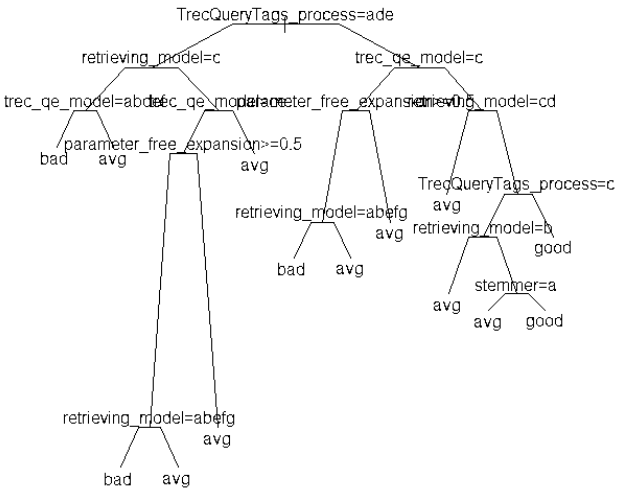
It was concretely used in Ayter et al. where 80,000 combinations of components and hyper-parameters were evaluated on the 100 queries of TREC 7 and 8 ad hoc track topics. The combinations differed on the stemmer used to index the documents (4 different stemmers were used), the topic tags used to build the query, the weighting model (7 different models), the query expansion model (6 different models) and the query expansion hyper-parameters which take different values as well. The authors showed that the most important parameters, the ones that influence the results the most, depend on the difficulty of the query, that is to say whether relevant documents will be easily found by the search engine or not [7] (see Figure 17).

In Figure 17a where the easy queries only are analysed, we can see that the most influential parameter is the query expansion model used because this is the one where the tree first split, here for the value c, which corresponds to the Info query expansion model. The retrieving or matching model is the second most influential parameter. For hard queries however, the most influential parameter is the topic part used for the query. In that research the authors either used the title only, or the other topic fields, narrative and descriptive, that provide more information on the users' need related to the topic. The leaves of the tree is whether the decision for a query is "easy" (good performance), "average" or "hard" (bad performance) when following a branch from the root to the leave. The main overall conclusion is that the influential parameters are not the same for easy and hard queries; giving the intuition that obtaining the best performance cannot be by applying the same process whatever the queries are. This was further analysed in [53], where more TREC collections were studied with the same conclusions.

These results are in favour of considering the system component parameters, not at a global level like search grid or other optimisation methods do in IR, but rather at a finer grain.



(**a**) Easy queries



(**b**) Difficult queries

Figure 17. The parameters that affect retrieval effectiveness the most depend on the query difficulty. On (**a**), for easy queries, the most important parameter for search effectiveness optimisation is the choice of the query expansion component; on (**b**), for hard queries, the most important parameter is the topic parts used for building the query, then the weighting component and in third the query expansion model. Figure reprinted with permission from [7], Copyright 2015, J.UCS.

7. Discussion and Conclusions

Understanding information retrieval effectiveness involves considering several dimensions. In this paper, we focused on the system and the query, while the document collection and the effectiveness measures were in the background.

From **evaluation forums and shared tasks**, although participants provide some detailed description of the systems they designed, the information is not enough structured or detailed to draw conclusions, except in broad strokes. The main conclusions from the analyses of shared tasks results are:

- C1: it is possible to distinguish between effective and non-effective systems on average over a query set;
- C2: effectiveness of systems has increased over years thanks to the effort put in the domain;

- C3: some queries are easy for all the systems while others are hard for all (see Figure 18, left-side part) but systems do not always fail or succeed on the same queries (see Figure 18, right side part). Some systems have a similar profile, they fail/success on the same queries.

However, it was not possible to understand system successes and failures.

Regarding C1, we also considered the participants' results from the first 7 years of TREC ad hoc for a total of 496 systems (or runs) and considered the 130 effectiveness measures from trec_eval that evaluate (system, query) pairs, such as in [84]. Correlation when considering pairs of measurements for a given topic and a given system are high, which means it is possible to distinguish between effective and non effective systems, it does not depend on the measure used (see Figure 19).

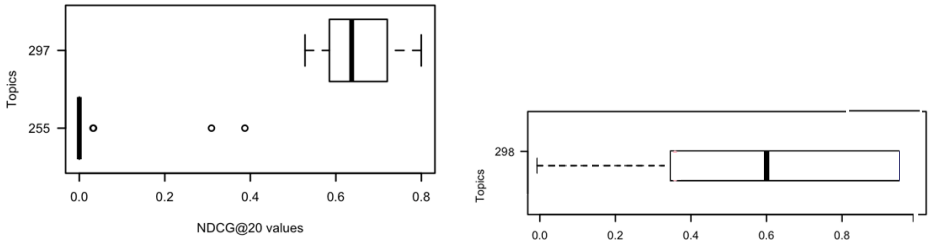


Figure 18. Some queries are easy for all the systems, some are hard for all, other depends on the system. On the TREC topic 297, all the analysed systems obtained at least 0.5 as NDCG@20, half of them obtained more than 0.65 and some obtained 0.8, which is high. For topic 255, all the systems failed but 3, only one obtained more than 0.3. The right part boxplot, as opposed to the left side ones, shows that for topic 298, the system effectiveness have a large span from 0 to almost 1.

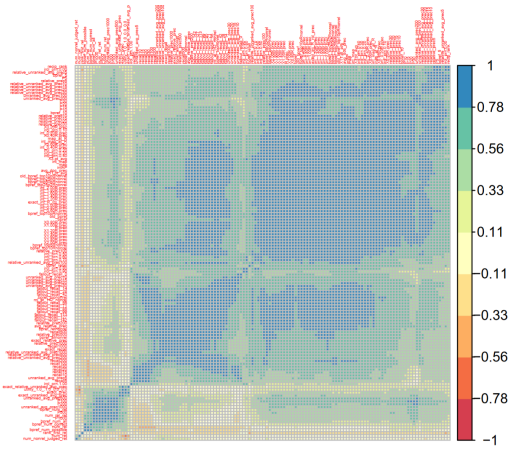


Figure 19. When considering a given system and a given query, the effectiveness measure used to compare the systems does not matter much: all are strongly correlated. Pearson correlation values between two effectiveness measurements on two measures for a given (system, query) pair. Correlations are represented using a divergent palette (a central colour, yellow, and 2 shades depending on whether the values go for negative—red—or positive values—blue).

Regarding C2, although it might be overgeneralisation of a single phenomenon, results from SMART system are convincing: the performances almost double when considering the results on the first participation to TREC ad hoc compared to the ones obtained 6 years later (see details Section 4).

Regarding C3, many studies agreed on this observation (see Figure 18). Moreover, systems with the same profiles belong generally to the same research group which means

that they may be just small variants one from the other (e.g., different hyper-parameters but basically the same query processing chain, using the same components).

Different attempts to extract more information from official runs were not conclusive.

From **automatically generated query processing chains**, we have a deep knowledge on the systems, we know exactly what components are used with which hyper-parameters. From the analyses that used these data, we can conclude:

- C4: some components and hyper-parameters are more influential than others and informed choices can be made;
- C5: the choice of the most appropriate components depends on the query level of difficulty.

Regarding C4: for some components or their hyperparameters, their choice will have a huge impact on the system effectiveness, for some others, there is no or little impact on effectiveness. This means that if one wants to tune or decide on not all the parameters but a few they should start by the more influential ones. Moreover, we know what is the best decision on some components: a stoplist should be used, a stemmer should be used, but the choice of the stemmer does not matter much; considering the weighting models that are implemented in Terrier, dirichletLM should be avoid, BB2 is a much better option.

Regarding C5: the choice of the most appropriate query processing chain is in relation with the query level of difficulty. In other words, different queries need different processing chains. This means that if we want to increase system effectiveness in the future, we should not just tune the system on a per collection basis: grid searching or any other more sophisticated version of parameter optimisation is not enough. What we need rather is to adapt the processing chain to the query.

From **query analyses**, we can conclude:

- C6: a single query feature or a combination of features have not been proven to explain system effectiveness;
- C7: query features can explain somehow system effectiveness.

Despite their apparent contradiction, C6 and C7 are in fact complementary. Some query features or combination of them seems to be accurate to predict not individual systems but average system effectiveness; there is also some success on predicting easy queries. Systems are, however, more easily distinguishable based on the difficult queries, not the easy ones for which they are more homogeneous in their successes. Up to now, the accuracy of features or feature combinations has not demonstrated that they can explain system effectiveness; correlation values that are reported are seldom over 0.5 and more tricky studies do not report scatter plots.

Although we do not yet understand well the factors of system effectiveness, the studies show that not a single system, while effective in average on a query set, is able to answer all the queries well (mainly C5 in addition to C3 and C4). Advanced IR techniques can be grounded on this knowledge. Selective query expansions (SQE) for example, where a meta-system has two alternative component combinations which differ on using or omitting the automatic query reformulation phase, made use of the fact that some queries benefit from being expanded while other do not [85–87]. SQE has not been proven to be very effective, certainly due to both the limited number of configurations used at that time (two different query processing chains) and the relatively poor learning techniques used at that time. Selective query processing expand SQE concept, where the system decides which one, from a possibly large number of component combinations, should be used for each individual query [10,15,88]. Here, the results were more conclusive. For example, Bigot et al. [89] developed a model that learns the best query processing chain to use for a query based on subsets of documents. Although this makes the method applicable for repeated queries only, it can be an appropriate approach for real world web search engines. Deveaud et al. [90] learn to rank the query processing chain for each new query. They used 20,000 different query processing chains. However, this very large number of combinations makes it difficult to use in real world systems. Arslan and Dinçer [47] developed a meta-model

that uses eight term-weighting models that could be chosen among for any new query. The meta-model has to be train, as well as the term-weighting models; this is performed by a grid search optimisation which limit the usability. Mothe and Ullah [91] present an approach to optimise the set of query processing chains that can be chosen among a selective query processing strategy. It is based on a risk-sensitive function that optimises the possible gain in considering a specific query processing chain. The authors show that 20 query processing chains is a good trade-off between the cost of maintaining different query processing chains and the gain on effectiveness. Still they do not explain the successes and failures.

Thus, to the question "Can we design a transparent model in terms of its performance on a query", I am tempted to answer: "No, at this stage of IR knowledge; further analyses are needed". I am convinced that data analytics methods can further been investigated to analyse the amount of data that have been generated by the community, both in shared tasks and in labs while tuning systems.

The robustness of the finding across collections would also worth investigating in the future.

Funding: This research received no external funding.

Conflicts of Interest: The author declares no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AP	Average Precision
CA	Correspondence Analysis
CIKM	Conference on Information and Knowledge Management
CLEF	Conference and Labs of the Evaluation Forum
IR	Information Retrieval
MAP	Mean Average Precision
PCA	Principal Component Analysis
QE	Query Expansion
QPP	Query Performance Prediction
SIGIR	Conference of the Association for Computing Machinery Special Interest Group in Information Retrieval
SQE	Selective Query Expansion
TREC	Text Retrieval Conference

References

1. Salton, G.; Wong, A.; Yang, C.S. A vector space model for automatic indexing. *Commun. ACM* **1975**, *18*, 613–620. [CrossRef]
2. Robertson, S.E.; Jones, K.S. Relevance weighting of search terms. *J. Am. Soc. Inf. Sci.* **1976**, *27*, 129–146. [CrossRef]
3. Robertson, S.; Zaragoza, H. *The Probabilistic Relevance Framework: BM25 and Beyond*; Now Publishers Inc.: Delft, The Netherlands, 2009; pp. 333–389.
4. Ponte, J.M.; Croft, W.B. A Language Modeling Approach to Information Retrieval. In Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '98, Melbourne, Australia, 24–28 August 1998; ACM: New York, NY, USA, 1998; pp. 275–281. [CrossRef]
5. Ounis, I.; Amati, G.; Plachouras, V.; He, B.; Macdonald, C.; Johnson, D. Terrier information retrieval platform. In *European Conference on Information Retrieval*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 517–519.
6. Taylor, M.; Zaragoza, H.; Craswell, N.; Robertson, S.; Burges, C. Optimisation methods for ranking functions with multiple parameters. In Proceedings of the 15th ACM International Conference on Information and Knowledge Management, Arlington, VA, USA, 6–11 November 2006; pp. 585–593.
7. Ayter, J.; Chifu, A.; Déjean, S.; Desclaux, C.; Mothe, J. Statistical analysis to establish the importance of information retrieval parameters. *J. Univers. Comput. Sci.* **2015**, *21*, 1767–1789.
8. Tague-Sutcliffe, J.; Blustein, J. *A Statistical Analysis of the TREC-3 Data*; NIST Special Publication SP: Washington, DC, USA, 1995; p. 385.
9. Banks, D.; Over, P.; Zhang, N.F. Blind men and elephants: Six approaches to TREC data. *Inf. Retr.* **1999**, *1*, 7–34. [CrossRef]
10. Dinçer, B.T. Statistical principal components analysis for retrieval experiments. *J. Am. Soc. Inf. Sci. Technol.* **2007**, *58*, 560–574. [CrossRef]

11. Mothe, J.; Tanguy, L. Linguistic analysis of users' queries: Towards an adaptive information retrieval system. In Proceedings of the 2007 Third International IEEE Conference on Signal-Image Technologies and Internet-Based System, Shanghai, China, 16–18 December 2007; pp. 77–84.
12. Harman, D.; Buckley, C. The NRRC reliable information access (RIA) workshop. In Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Sheffield, UK, 25–29 July 2004; pp. 528–529.
13. Mizzaro, S.; Robertson, S. Hits hits trec: Exploring ir evaluation results with network analysis. In Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Amsterdam, The Netherlands, 23–27 July 2007; pp. 479–486.
14. Harman, D.; Buckley, C. Overview of the reliable information access workshop. *Inf. Retr.* **2009**, *12*, 615. [CrossRef]
15. Bigot, A.; Chrisment, C.; Dkaki, T.; Hubert, G.; Mothe, J. Fusing different information retrieval systems according to query-topics: A study based on correlation in information retrieval systems and TREC topics. *Inf. Retr.* **2011**, *14*, 617. [CrossRef]
16. Ferro, N.; Silvello, G. 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, Pisa, Italy, 17–21 July 2016; pp. 25–34.
17. Ferro, N.; Silvello, G. Toward an anatomy of IR system component performances. *J. Assoc. Inf. Sci. Technol.* **2018**, *69*, 187–200. [CrossRef]
18. Louedec, J.; Mothe, J. A massive generation of ir runs: Demonstration paper. In Proceedings of the IEEE 7th International Conference on Research Challenges in Information Science (RCIS), Paris, France, 29–31 May 2013; pp. 1–2.
19. Wilhelm, T.; Kürsten, J.; Eibl, M. A tool for comparative ir evaluation on component level. In Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, Beijing, China, 24–28 July 2011; pp. 1291–1292.
20. Carmel, D.; Yom-Tov, E.; Darlow, A.; Pelleg, D. What makes a query difficult? In Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle,WA, USA, 6–11 August 2006; pp. 390–397.
21. Mothe, J.; Tanguy, L. Linguistic features to predict query difficulty. In *ACM Conference on Research and Development in Information Retrieval, SIGIR, Predicting Query Difficulty-Methods and Applications Workshop*; ACM: New York, NY, USA, 2005; pp. 7–10.
22. Zamani, H.; Croft, W.B.; Culpepper, J.S. Neural query performance prediction using weak supervision from multiple signals. In Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, Ann Arbor, MI, USA, 8–12 July 2018; pp. 105–114.
23. Carpineto, C.; Romano, G. A survey of automatic query expansion in information retrieval. *ACM Comput. Surv. (CSUR)* **2012**, *44*, 1–50. [CrossRef]
24. Azad, H.K.; Deepak, A. Query expansion techniques for information retrieval: A survey. *Inf. Process. Manag.* **2019**, *56*, 1698–1735. [CrossRef]
25. Moral, C.; de Antonio, A.; Imbert, R.; Ramírez, J. A survey of stemming algorithms in information retrieval. *Inf. Res. Int. Electron. J.* **2014**, *19*, n1.
26. Kamphuis, C.; de Vries, A.P.; Boytsov, L.; Lin, J. Which BM25 Do You Mean? A Large-Scale Reproducibility Study of Scoring Variants. In *Advances in Information Retrieval*; Jose, J.M., Yilmaz, E., Magalhães, J., Castells, P., Ferro, N., Silva, M.J., Martins, F., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 28–34.
27. Mizzaro, S. How many relevances in information retrieval? *Interact. Comput.* **1998**, *10*, 303–320. [CrossRef]
28. Ruthven, I. Relevance behaviour in TREC. *J. Doc.* **2014**, *70*, 1098–1117. [CrossRef]
29. Hofstätter, S.; Lin, S.C.; Yang, J.H.; Lin, J.; Hanbury, A. Efficiently teaching an effective dense retriever with balanced topic aware sampling. In Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, 11–15 July 2021; pp. 113–122.
30. Breslow, N.E.; Clayton, D.G. Approximate inference in generalized linear mixed models. *J. Am. Stat. Assoc.* **1993**, *88*, 9–25.
31. McCullagh, P.; Nelder, J.A. *Generalized Linear Models*, 2nd ed.; Chapman and Hall: London, UK, 1989.
32. Dumais, S.T. LSA and information retrieval: Getting back to basics. *Handb. Latent Semant. Anal.* **2007**, *293*, 322.
33. Sarwar, B.; Karypis, G.; Konstan, J.; Riedl, J. *Application of Dimensionality Reduction in Recommender System—A Case Study*; Technical Report; Department of Computer Science and Engineering, University of Minnesota: Minneapolis, MN, USA, 2000.
34. Benzécri, J.P. Statistical analysis as a tool to make patterns emerge from data. In *Methodologies of Pattern Recognition*; Elsevier: Amsterdam, The Netherlands, 1969; pp. 35–74.
35. Ward, J.H., Jr. Hierarchical grouping to optimize an objective function. *J. Am. Stat. Assoc.* **1963**, *58*, 236–244. [CrossRef]
36. Li, B.; Friedman, J.; Olshen, R.; Stone, C. Classification and regression trees (CART). *Biometrics* **1984**, *40*, 358–361.
37. Ho, T.K. Random decision forests. In Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, QC, Canada, 14–16 August 1995; Volume 1, pp. 278–282.
38. Gunning, D. *Explainable Artificial Intelligence*; Defense Advanced Research Projects Agency (DARPA): Arlington, VA, USA, 2017; p. 2.
39. Zhang, Y.; Chen, X. Explainable recommendation: A survey and new perspectives. *Found. Trends® Inf. Retr.* **2020**, *14*, 1–101. [CrossRef]
40. Harman, D. *Overview of the First Text Retrieval Conference (trec-1)*; NIST Special Publication SP: Washington, DC, USA, 1992; pp. 1–532.
41. Harman, D. Overview of the first TREC conference. In Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Pittsburgh, PA, USA, 27 June–1 July 1993; pp. 36–47.

42. Buckley, C.; Mitra, M.; Walz, J.A.; Cardie, C. *SMART high precision: TREC 7*; NIST Special Publication SP: Washington, DC, USA, 1999; pp. 285–298.
43. Clarke, C.L.; Craswell, N.; Soboroff, I. *Overview of the Trec 2009 Web Track*; Technical Report; University of Waterloo: Waterloo, ON, Canada, 2009.
44. Collins-Thompson, K.; Macdonald, C.; Bennett, P.; Diaz, F.; Voorhees, E.M. *TREC 2014 Web Track Overview*; Technical Report; University of Michigan: Ann Arbor, MI, USA, 2015.
45. Kompaore, D.; Mothe, J.; Baccini, A.; Dejean, S. Query clustering and IR system detection. Experiments on TREC data. In Proceedings of the ACM International Workshop for Ph. D. Students in Information and Knowledge Management (ACM PIKM 2007), Lisboa, Portugal, 5–10 November 2007.
46. Hanbury, A.; Müller, H. Automated component–level evaluation: Present and future. In *International Conference of the Cross-Language Evaluation Forum for European Languages*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 124–135.
47. Arslan, A.; Dinçer, B.T. A selective approach to index term weighting for robust information retrieval based on the frequency distributions of query terms. *Inf. Retr. J.* **2019**, *22*, 543–569. [CrossRef]
48. Di Buccio, E.; Dussin, M.; Ferro, N.; Masiero, I.; Santucci, G.; Tino, G. Interactive Analysis and Exploration of Experimental Evaluation Results. In *European Workshop on Human-Computer Interaction and Information Retrieval EuroHCIR*; Citeseer: Nijmegen, The Netherlands, 2011; pp. 11–14.
49. Compaoré, J.; Déjean, S.; Gueye, A.M.; Mothe, J.; Randriamparany, J. Mining information retrieval results: Significant IR parameters. In Proceedings of the First International Conference on Advances in Information Mining and Management, Barcelona, Spain, 23–29 October 2011; Volume 74.
50. Hopfgartner, F.; Hanbury, A.; Müller, H.; Eggel, I.; Balog, K.; Brodt, T.; Cormack, G.V.; Lin, J.; Kalpathy-Cramer, J.; Kando, N.; et al. Evaluation-as-a-service for the computational sciences: Overview and outlook. *J. Data Inf. Qual. (JDIQ)* **2018**, *10*, 1–32. [CrossRef]
51. Kürsten, J.; Eibl, M. A large-scale system evaluation on component-level. In *European Conference on Information Retrieval*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 679–682.
52. Angelini, M.; Fazzini, V.; Ferro, N.; Santucci, G.; Silvello, G. CLAIRE: A combinatorial visual analytics system for information retrieval evaluation. *Inf. Process. Manag.* **2018**, *54*, 1077–1100. [CrossRef]
53. Dejean, S.; Mothe, J.; Ullah, M.Z. Studying the variability of system setting effectiveness by data analytics and visualization. In *International Conference of the Cross-Language Evaluation Forum for European Languages*; Springer: Cham, Switzerland, 2019; pp. 62–74.
54. De Loupy, C.; Bellot, P. Evaluation of document retrieval systems and query difficulty. In Proceedings of the Second International Conference on Language Resources and Evaluation (LREC 2000) Workshop, Athens, Greece, 31 May–2 June 2000; pp. 32–39.
55. Banerjee, S.; Pedersen, T. Extended gloss overlaps as a measure of semantic relatedness. In Proceedings of the IJCAI 2003, Acapulco, Mexico, 9–15 August 2003; pp. 805–810.
56. Patwardhan, S.; Pedersen, T. Using WordNet-based context vectors to estimate the semantic relatedness of concepts. In Proceedings of the Workshop on Making Sense of Sense: Bringing Psycholinguistics and Computational Linguistics Together, Trento, Italy, 4 April 2006.
57. Cronen-Townsend, S.; Zhou, Y.; Croft, W.B. Predicting query performance. In Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Tampere, Finland, 11–15 August 2002; pp. 299–306.
58. Scholer, F.; Williams, H.E.; Turpin, A. Query association surrogates for web search. *J. Am. Soc. Inf. Sci. Technol.* **2004**, *55*, 637–650. [CrossRef]
59. He, B.; Ounis, I. Inferring query performance using pre-retrieval predictors. In *International Symposium on String Processing and Information Retrieval*; Springer: Berlin/Heidelberg, Germany, 2004; pp. 43–54.
60. Hauff, C.; Hiemstra, D.; de Jong, F. A survey of pre-retrieval query performance predictors. In Proceedings of the 17th ACM Conference on Information and Knowledge Management, Napa Valley, CA, USA, 26–30 October 2008; pp. 1419–1420.
61. Zhao, Y.; Scholer, F.; Tsegay, Y. Effective pre-retrieval query performance prediction using similarity and variability evidence. In *European Conference on Information Retrieval*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 52–64.
62. Sehgal, A.K.; Srinivasan, P. Predicting performance for gene queries. In Proceedings of the ACM SIGIR 2005 Workshop on Predicting Query Difficulty-Methods and Applications. Available online: http://www.haifa.il.ibm.com/sigir05-qp (accessed on 15 May 2022).
63. Zhou, Y.; Croft, W.B. Ranking robustness: A novel framework to predict query performance. In Proceedings of the 15th ACM International Conference on Information and Knowledge Management, Arlington, VA, USA, 6–11 November 2006; pp. 567–574.
64. Vinay, V.; Cox, I.J.; Milic-Frayling, N.; Wood, K. On ranking the effectiveness of searches. In Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle,WA, USA, 6–11 August 2006; pp. 398–404.
65. Aslam, J.A.; Pavlu, V. Query hardness estimation using Jensen-Shannon divergence among multiple scoring functions. In *European Conference on Information Retrieval*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 198–209.
66. Zhou, Y.; Croft, W.B. Query performance prediction in web search environments. In Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Amsterdam, The Netherlands, 23–27 July 2007; pp. 543–550.

67. Shtok, A.; Kurland, O.; Carmel, D. Predicting query performance by query-drift estimation. In *Conference on the Theory of Information Retrieval*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 305–312.
68. Carmel, D.; Yom-Tov, E. Estimating the query difficulty for information retrieval. *Synth. Lect. Inf. Concepts Retr. Serv.* **2010**, *2*, 1–89.
69. Cummins, R.; Jose, J.; O'Riordan, C. Improved query performance prediction using standard deviation. In Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, Beijing, China, 24–28 July 2011; pp. 1089–1090.
70. Roitman, H.; Erera, S.; Weiner, B. Robust standard deviation estimation for query performance prediction. In Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval, Amsterdam, The Netherlands, 1–4 October 2017; pp. 245–248.
71. Chifu, A.G.; Laporte, L.; Mothe, J.; Ullah, M.Z. Query performance prediction focused on summarized letor features. In Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, Ann Arbor, MI, USA, 8–12 July 2018; pp. 1177–1180.
72. Zhang, Z.; Chen, J.; Wu, S. Query performance prediction and classification for information search systems. In *Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data*; Springer: Cham, Switzerland, 2018; pp. 277–285.
73. Khodabakhsh, M.; Bagheri, E. Semantics-enabled query performance prediction for ad hoc table retrieval. *Inf. Process. Manag.* **2021**, *58*, 102399. [CrossRef]
74. Molina, S.; Mothe, J.; Roques, D.; Tanguy, L.; Ullah, M.Z. IRIT-QFR: IRIT query feature resource. In *International Conference of the Cross-Language Evaluation Forum for European Languages*; Springer: Cham, Switzerland, 2017; pp. 69–81.
75. Macdonald, C.; He, B.; Ounis, I. Predicting query performance in intranet search. In Proceedings of the SIGIR 2005 Query Prediction Workshop, Salvador, Brazil, 15–19 August 2005.
76. Faggioli, G.; Zendel, O.; Culpepper, J.S.; Ferro, N.; Scholer, F. sMARE: A new paradigm to evaluate and understand query performance prediction methods. *Inf. Retr. J.* **2022**, *25*, 94–122. [CrossRef]
77. Hashemi, H.; Zamani, H.; Croft, W.B. Performance Prediction for Non-Factoid Question Answering. In Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval, Paris, France, 21–25 July 2019; pp. 55–58.
78. Roy, D.; Ganguly, D.; Mitra, M.; Jones, G.J. Estimating Gaussian mixture models in the local neighbourhood of embedded word vectors for query performance prediction. *Inf. Process. Manag.* **2019**, *56*, 1026–1045. [CrossRef]
79. Anscombe, F. American Statistical Association, Taylor & Francis, Ltd. are collaborating with JSTOR to. *Am. Stat.* **1973**, *27*, 17–21.
80. Grivolla, J.; Jourlin, P.; de Mori, R. *Automatic Classification of Queries by Expected Retrieval Performance*; SIGIR: Salvador, Brazil, 2005.
81. Raiber, F.; Kurland, O. Query-performance prediction: Setting the expectations straight. In Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval, Gold Coast, Australia, 6–11 July 2014; pp. 13–22.
82. Mizzaro, S.; Mothe, J.; Roitero, K.; Ullah, M.Z. Query performance prediction and effectiveness evaluation without relevance judgments: Two sides of the same coin. In Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, Ann Arbor, MI, USA, 8–12 July 2018; pp. 1233–1236.
83. Aslam, J.A.; Savell, R. On the Effectiveness of Evaluating Retrieval Systems in the Absence of Relevance Judgments. In Proceedings of the 26th ACM SIGIR, Toronto, ON, Canada, 28 July–1 August 2003; pp. 361–362.
84. Baccini, A.; Déjean, S.; Lafage, L.; Mothe, J. How many performance measures to evaluate information retrieval systems? *Knowl. Inf. Syst.* **2012**, *30*, 693–713. [CrossRef]
85. Amati, G.; Carpineto, C.; Romano, G. Query difficulty, robustness, and selective application of query expansion. In *European Conference on Information Retrieval*; Springer: Berlin/Heidelberg, Germany, 2004; pp. 127–137.
86. Cronen-Townsend, S.; Zhou, Y.; Croft, W.B. A framework for selective query expansion. In Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management, Washington, DC, USA, 8–13 November 2004; pp. 236–237.
87. Zhao, L.; Callan, J. Automatic term mismatch diagnosis for selective query expansion. In Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval, Portland, OR, USA, 12–16 August 2012; pp. 515–524.
88. Deveaud, R.; Mothe, J.; Ullah, M.Z.; Nie, J.Y. Learning to Adaptively Rank Document Retrieval System Configurations. *ACM Trans. Inf. Syst. (TOIS)* **2018**, *37*, 3. [CrossRef]
89. Bigot, A.; Déjean, S.; Mothe, J. Learning to Choose the Best System Configuration in Information Retrieval: The Case of Repeated Queries. *J. Univers. Comput. Sci.* **2015**, *21*, 1726–1745.
90. Deveaud, R.; Mothe, J.; Nia, J.Y. Learning to Rank System Configurations. In Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, CIKM '16, Indianapolis, IN, USA, 24–28 October 2016; ACM: New York, NY, USA, 2016; pp. 2001–2004.
91. Mothe, J.; Ullah, M.Z. Defining an Optimal Configuration Set for Selective Search Strategy-A Risk-Sensitive Approach. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management, Online, 1–5 November 2021; pp. 1335–1345.