PROMISE

Participative Research labOratory for Multimedia and Multilingual Information Systems Evaluation

FP7 ICT 2009.4.3, Intelligent Information Management

Deliverable 4.5

Rank Analysis Techniques for Interactive Environments

Version 1.00, 24 September 2013

Network of Excellence co-funded by the 7th Framework Program of the European Commission, grant agreement no. 258191

Document Information

Deliverable number	4.5
Deliverable title	Rank Analysis Techniques for Interactive Environments
Delivery date	24 September 2013
Lead contractor for this deliverable	UNIPD
Author(s)	Marco Angelini, Nicola Ferro, Giuseppe Santucci, Gianmaria Silvello
Participant(s)	UNIPD, SICS, ROMA1, ZHAW
Workpackage	WP4
Workpackage title	Evaluation Metrics and Methodologies
Workpackage leader	UvA
Dissemination Level	PU – Public
Version	1.00
Keywords	Rank Analysis, Interactive Visualizations, Experimental Evaluation Tool

History of Versions

Version	Date	Status	Author	Description
0.10	2013-07-07	Draft	UNIPD	Initial scheleton
0.15	2013-07-08	Draft	UNIPD	Initial draft of the sections
0.20	2013-07-09	Draft	UNIPD, ROMA1	Draft of the deliverable
0.30	2013-08-27	Draft	UNIPD, ZHAW, SICS, UvA	Draft circulated to the partners
0.30	2013-08-28	Draft	UNIPD, ZHAW	Comments received from the partners
0.90	2013-08-28	Draft	UNIPD	Comments implemented
0.95	2013-09-16	Draft	UNIPD, UvA	Comments received from the partners
1.00	2013-09-20	Draft	UNIPD	Comments implemented

Abstract

This deliverable describes the rank analysis activities conducted in the context of task 4.5 of WP4. It presents an innovative visual analytics environment, called *Visual Analytics Tool for Experimental Evaluation (VATE^2)*, which eases and makes the experimental evaluation process more effective. In particular, VATE^2 supports and improves two typical phases of the experimental evaluation process,

namely *performance analysis* and *failure analysis*, and introduces a completely new phase, the *what-if analysis*. All these activities are crucial for conducting rank analysis and they have been carried out by producing an interactive environment.

Contents

Executive Summary

Information Retrieval (IR) systems, ranging from World Wide Web search engines [Buettcher et al., 2010; Croft et al., 2009] to enterprise search [Burnett et al., 2006], intellectual property and patent search [Lupu and Hanbury, 2013], expertise retrieval systems [Balog et al., 2012] and passing through information access components in wider systems such as digital libraries [Candela et al., 2007; Fox et al., 2012; Witten et al., 2009], are key technologies to get access to relevant information items in a context where information overload is a day-to-day experience of every user.

To get rid of such huge amount of information, ever increasing, IR systems are getting more and more complex: they rely on very sophisticated ranking models where many different parameters affect the obtained results and are comprised of several components, which interact together in very complex ways to produce a list of relevant documents in response to a user query. Ranking is a central and ubiquitous issue in this context since it is necessary to return the results retrieved in response to a user query according to the estimation of their relevance to that query and the user information need [Mizzaro, 1997].

Designing, developing, and testing an IR system is a challenging task, especially when it comes to understanding and analysing the behaviour of the system under different conditions of use in order to tune or to improve it as to achieve the level of effectiveness needed to meet the user expectations. One of the very goals of *Participative Research labOratory for Multimedia and Multilingual Information Systems Evaluation (PROMISE)* is to ease this process and to provide interactive and visual tool to simplify the work of analysts.

Experimental evaluation and large-scale evaluation campaigns provide a means for assessing the performance of IR systems and represent the starting point for investigating and understanding their behaviour. However, the complex interactions among the components of an IR system are often hard to trace down, to explain in the light of the obtained results, and to interpret in the perspective of possible modifications to be made to improve the ranking of the results, thus making this activity extremely difficult. Conducting such analyses is especially resource demanding in terms of time and human effort, since it requires to manually inspect, for several queries, system logs, intermediate outputs of system components, and, mostly, long lists of retrieved documents which need to be read one by one in order to try to figure out why they have been ranked in that way with respect to the query at hand. This activity is usually called, in the IR field, *failure analysis* [Buckley, 2004; Harman, 2008; Savoy, 2007] and it is deemed a fundamental activity in experimental evaluation and system development even if it is too often overlooked due to its difficulty.

This deliverable aims at reducing the effort needed to carry out both the performance and failure analyses, which are fundamental steps in experimental evaluation, by introducing the possibility of effectively interacting with the experimental results.

Moreover, this deliverable introduces a completely new phase in the experimental evaluation process, that we called *what-if analysis* and is aimed at getting an estimate of what could be the effects of a modification to the IR system under examination before actually implementing it and starting a new evaluation and analysis cycle for understanding how it has produced the expected outcomes. This represent a major step forward, since this kind of analysis allows us to save huge amounts of time and effort in IR system development and, to the best of our knowledge, it has never

been attempted before.

The main results achieved are synthesized by the feature of *Visual Analytics Tool for Experimental Evaluation (VATE^2)* which is the tool we developed for carrying out interactive rank analysis and that has been integrated in the *Distributed Information Retrieval Evaluation Campaign Tool (DIRECT)* infrastructure [Agosti et al., 2011a, 2012b, 2011b]. VATE^2:

(i) eases the *performance analysis*, which is one of the most consolidated activities in IR evaluation, albeit it is often the only one performed. This is achieved by interactive visualization and exploration of the experimental results, according to different metrics and parameters, and by providing simple visual means to immediately grasp whether the system would already have the potential to achieve the best performances or whether a complete new ranking strategy would be preferred;

(ii) explicitly assists *failure analysis*, making it part of a single and coherent workflow, while it is usually overlooked due its effortful nature. In particular, it introduces two new indicators, called *Relative Position (RP)* and *Delta Gain (ΔG)*, which allow us to visually (and also numerically) figure out the weak and strong parts of a ranking in order to quickly detect failing documents or topics and make hypotheses about how to improve them. This greatly reduces the effort needed to carry out this fundamental but extremely demanding activity and promises to make it a much more widespread practice.

(iii) introduces a completely new phase, the *what-if analysis*, aimed at estimating what effects fixing a failure might have, before needing to implement the fix and perform another round of evaluation in order to assess it. This helps in choosing the most promising fixes to be implemented and avoiding the potential harmful ones, thus saving effort and resources. This is achieved by learning a model of the behavior of the system, which allows us to guess which documents would be affected by a potential fix improving their ranking in the desired way and to further estimate the overall effect on the performances for all the topics of a given experiment.

Moreover, we model all the above phases in a single formal analytical framework, where all the different concepts and operations find a methodologically sound formulation and fit all together to contribute to the overall objective of making a step forward in experimental evaluation. This formal analytical framework paves the road also for future research since, due to its modularity, it allows for substituting its components and exploring alternative or improved solutions, keeping the overall coherence and enabling meaningful comparison between alternative approaches. For example, in Section 3.6, we propose a straightforward approach to learn a model of the examined system but alternative and more sophisticated solutions can be envisioned: the proposed formal analytical framework will allow for formulating them in a coherent way, to compare them with the alternative, and to seamlessly integrate them with the other parts of the framework. Furthermore, the formal analytical framework provides the bases for the design and development of the *Visual Analytics (VA)* environment which demonstrates the feasibility of the proposed approaches.

The overall idea of exploiting visual and interactive techniques for exploring the experimental results is quite new to the IR field, since representation and analysis of the experimental results

typically happens in static ways or batches. This is also new to the VA field since VA techniques are usually applied to the presentation and interaction with the outputs, i.e. the ranked result list and documents [Agosti et al., 2012a; Zhang, 2008], produced by an IR system but almost never to the analysis, exploration, and interpretation of the performances and behavior of the IR system itself.

A final contribution of the deliverable is to have performed an initial validation of the VATE^2 environment with domain experts in order to get feedback about its innovation potential, its suitability for the purpose, and the appropriateness of the proposed solutions.

1 Introduction

This deliverable describes the *Visual Analytics Tool for Experimental Evaluation (VATE^2)* which is an interactive visual tool developed for carrying out ranking analysis in the context of PROMISE:

1. it describes the main phases of ranking analysis in the PROMISE project;
2. it reports the formal framework which paved the road for future developments and improvements of VATE^2;
3. it presents the actual prototype developed and tested with experimental data managed by the DIRECT infrastructure [Agosti et al., 2011a, 2012b, 2011b];
4. it reports the results of the system validation conducted with IR experts;
5. and, it details the functioning of VATE^2 explaining how it can be employed for actually carrying out the experimental evaluation process.

The aim of this deliverable is to present VATE^2 which is a fundamental component of the DIRECT infrastructure, but that can also live independently in other environments. We will show how it improves the state-of-the-art in ranking analysis.

Basically, VATE^2 aims at reducing the effort needed to carry out both the performance and failure analyses, which are fundamental steps in experimental evaluation, by introducing the possibility of effectively interacting with the experimental results.

Moreover, it introduces a completely new phase in the experimental evaluation process, that we called *what-if analysis* and is aimed at getting an estimate of what could be the effects of a modification to the IR system under examination before actually implementing it and starting a new evaluation and analysis cycle for understanding how it has produced the expected outcomes. This represent a major step forward, since this kind of analysis allows us to save huge amounts of time and effort in IR system development and, to the best of our knowledge, it has never been attempted before.

The most tight and innovative integration between the visual and analytical parts happens in the *what-if* analysis where clustering and machine learning algorithms are used to drive the visualization animations and, in turn, user inputs on the visualization are used to reactivate classification and learning algorithms whose results produce new visualizations and animations and lead to the estimation of the possible impact on the performances of fixing a given failure.

The overall idea of exploiting visual and interactive techniques for exploring the experimental results is quite new to the IR field, since representation and analysis of the experimental results typically happens in static ways or batches. This is also new to the VA field since VA techniques are usually applied to the presentation and interaction with the outputs, i.e. the ranked result list and documents [Agosti et al., 2012a; Zhang, 2008], produced by an IR system but almost never to the analysis, exploration, and interpretation of the performances and behavior of the IR system itself.

The deliverable is organized as follows: Section 2 introduces the conceptual framework which supports and enhances the experimental evaluation methodology and practice by exploiting visual

analytics techniques; Section 3 explains in detail the proposed formal analytical framework; Section 4 presents the actual prototype which implement the proposed methodologies; Section 5 discusses the validation of the adopted methodologies and prototype with domain experts; finally, Section 6 draws some conclusions.

2 Conceptual Framework

According to the Cranfield paradigm and the best practices followed by the large-scale international evaluation campaigns, the inputs to the analysis phase in the overall experimental evaluation process are:

- the set of *documents* D and the set of *topics* T;
- the *ground-truth* or *relevance judgements* or *pool*[1] GT which determine the "correct answers" for each topic $t_i \in T$ from which performance measures are computed and systems are compared;
- one (or more) *experiment* or *run*[2] of the IR system under examination which, for each topic $t_i \in T$ is constituted by a ranked list of documents $d_j \in D$ retrieved by the IR system in response to t_i;

It can be noted that in this paradigm IR systems are dealth with as a kind of "black boxes", whose internals and intermediate results cannot be examined separately, as also pointed out by Robertson [Robertson, 1981]: "if we want to decide between alternative indexing strategies for example, we must use these strategies *as part of a complete information retrieval system*, and *examine its overall performance* (with each of the alternatives) directly". As we will discuss in the following, these features of the experimental evaluation process have been explicitly taken into account in modeling, formalizing, designing, and developing VATE^2.

Figure 1 shows the overall framework adopted by VATE^2 to support experimental evaluation. As discussed in Section 1, *performance analysis* and *failure analysis* are the traditional phases carried out during experimental evaluation, where VATE^2 contributes to make them more effective and to reduce the needed effort via both tailored visualizations and measures and high interaction with the experimental data; *what-if analysis* is a new phase aimed at estimating the possible effects of a modification to the IR system under examination before needing to actually implement it and starting a new evaluation cycle to assess its impact on performances. *Topic Level* concerns the analysis of the documents retrieved in response to a given topic of a run while *Experiment Level* deals with overall statistics and effects concerning the whole set of topics of a run, i.e. all the different ranked lists of retrieved documents.

Therefore, VATE^2:

- supports performance analysis on a topic-by-topic basis and with aggregate statistics over the whole set of topics;
- facilitates failure analysis to let researchers and developers to more easily spotting and understanding failing documents and topics;

[1] To be precise, for each topic, the documents to be judged are sampled, according to some strategy, into a *pool* and then, for each of them, a *relevance judgment* (binary or graded) is associated. The final set of pooled documents together with their relevance judgements constitutes the *ground-truth*. However, in practice, these three terms are often used as synonyms.

[2] In the following we will use *run* and *experiment* interchangeably.

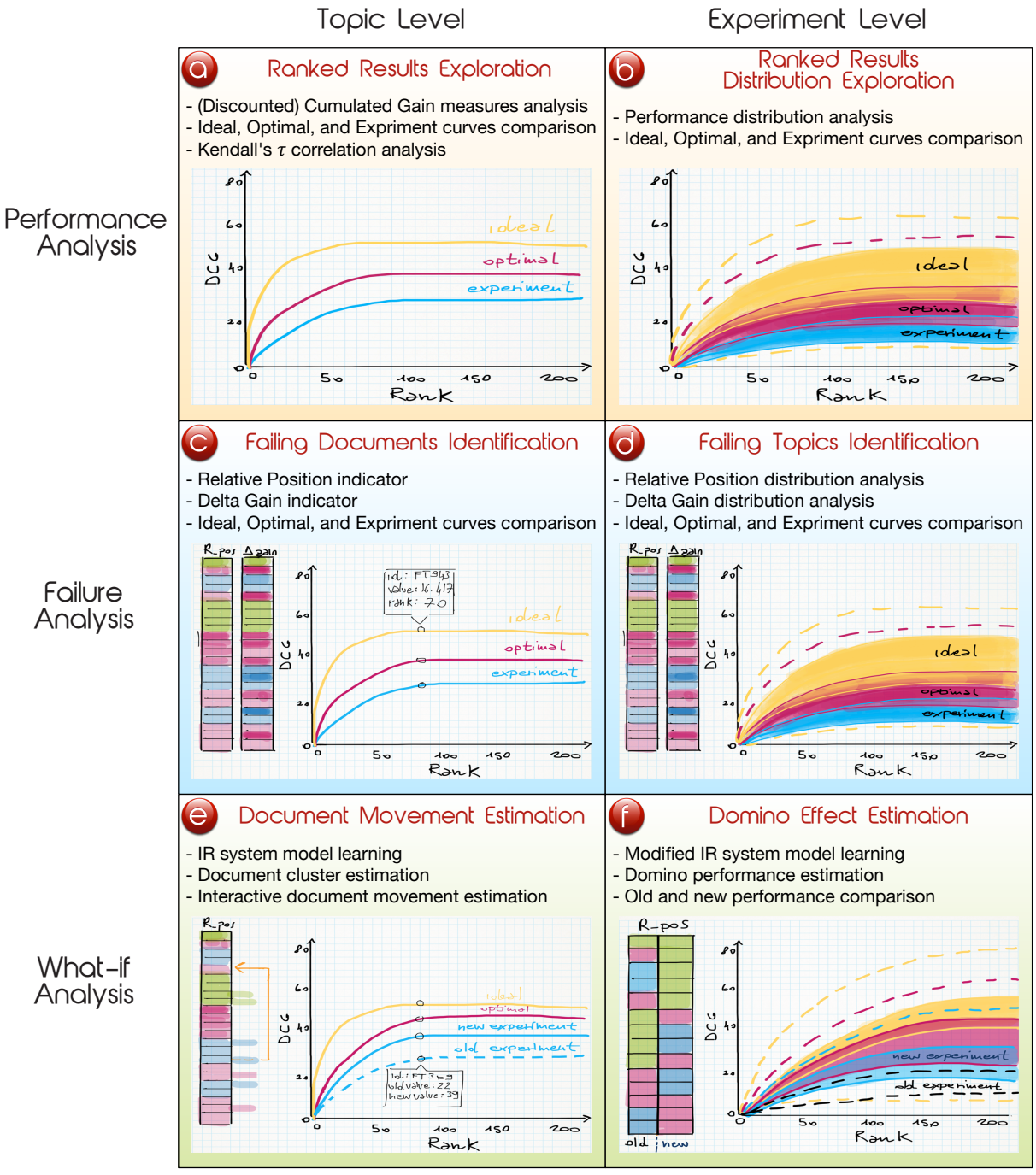


Figure 1: VATE^2 overall framework.

- introduces what-if analysis to allow for an estimation of the possible effects of a modification both at the single topic level and on the whole experiment.

Main target users of VATE^2 are domain experts, i.e. researchers and developers in the IR and related fields who need to understand and improve their systems. Moreover, VATE^2 can be useful also for educational purposes, e.g. in undergraduate or PhD courses where information retrieval is taught and where explaining how to interpret the performances of an IR system is an important part of the teaching. Finally, it may find application also in production contexts as a tool for monitoring and interpreting the performances of a running system as to ensure that the desired service levels are met.

In the following sections, we will describe each of these steps of Figure 1 in more detail from top left to bottom right.

2.1 Ranked Results Exploration

In order to quantify the performances of an IR system, we adopt the *Discounted Cumulated Gain (DCG)* family of measures [Järvelin and Kekäläinen, 2002a; Keskustalo et al., 2008] which have proved to be especially well-suited for analyzing ranked results list because they allow for graded relevance judgements and embed a model of the user behavior while he scrolls down the result list which gives also an account of its overall satisfaction.

The overall idea of the DCG family of measures is to assign a gain to each relevance grade and, for each position in the ranked list, a discount is computed. Then, for each rank, DCG is computed by using the cumulative sum of the discounted gains up to that rank position. This gives raise to a whole family of measures, depending on the choice of the gain assigned to each relevance grade and the used discounting function. Typical instantiations of DCG measures make use of positive gains – e.g., 0 for not relevant documents, 1 for partially relevant ones, and 3 for highly relevant ones – and logarithmic functions to smooth the discount for higher ranks – e.g. a $\log_2$ function is used to model impatient users while a $\log_{10}$ function is used to model very patient users in scanning the result list. DCG curves have a typical monotonic non-decreasing behavior and the higher the value of DCG at a given rank position the better the performances as all well as the steeper the slope the better is the ranking.

We compare the result list produced by an experiment with respect to an *ideal* ranking created starting from the relevant documents in the ground-truth, which represents the best possible results that an experiment can return – this ideal ranking is what is usually used to normalize the DCG measures. In addition to what is typically done, we compare the result list with respect to an *optimal* one created with the same documents retrieved by the IR system but with a optimal ranking, i.e. a permutation of the results retrieved by the experiment aimed at maximizing its performances by sorting the retrieved documents in decreasing order of relevance. Therefore, the *ideal ranking* compares the experiment at hand with respect to the best results possible, i.e. considering also relevant documents not retrieved by the system, while the *optimal ranking* compares an experiment with respect to what it could have been done better with the same retrieved documents.

The proposed visualization, shown in Figure 1.(a), allows for interaction with these curves, e.g. by dynamically choosing different measures in the DCG family, adjusting the discounting function,

and comparing curves and their values rank by rank.

Overall, this method makes it easy to grasp the distance of an IR system from both its own optimal performances and the best performances possible and to get an indication about whether the system is going in the right direction or whether a completely different approach would be preferred. Indeed, we support researchers and developers in try to answer an ambitious question: is it better to invest on improving the ranking of the documents already retrieved by the system or is it better to develop a complete new strategy for searching documents? Or, in other terms, the proposed techniques allow us to understand whether the system under examination is satisfactory from the recall point of view but unsatisfactory from the precision one, thus possibly benefiting from re-ranking, or if the system has also a too low recall, and thus it would benefit more from a new strategy. The former case is when the experiment curve is somewhat far away from the optimal curve but the optimal curve is close to the ideal one; the latter case is when the optimal curve is far away from the ideal one, regardless how close is the experiment curve to the optimal one.

In order to support the visual intuition, we provide also a Kendall's τ correlation analysis [Kendall, 1948; Voorhees, 2001] between the three above mentioned curves: each experiment is described by a pair $(\tau_{ideal-opt}, \tau_{opt-exp})$, where $\tau_{ideal-opt}$ denotes the Kendall τ correlation among the ideal and the optimal rankings, while $\tau_{opt-exp}$ denotes the Kendall τ among the optimal and experiment rankings. When the pair is $(1,1)$ the best performance possible is achieved. A pair where $\tau_{ideal-opt}$ is high and $\tau_{opt-exp}$ is low suggests that "re-ranking" could probably improve effectiveness, since there is a strong correlation between ideal and optimal ranking, thus suggesting that the IR approach was quite effective in retrieving relevant documents, but not in the document ranking. A pair where $\tau_{ideal-opt}$ is low or negative suggests "re-query" on the entire collection as possible strategy to improve retrieval effectiveness, since also an optimal re-ranking of the retrieved document is far from the ideal ranking.

2.2 Ranked Results Distribution Exploration

The interactive visualization and performance analysis methodology described in the previous section concerns a single topic of an experiment. What is usually needed is to be able to analyze a run as a whole or to analyze a subset of its topics together because, for example, they are considered the hard ones where more problems occurred.

The ranked results distribution exploration, shown in Figure 1.(b), provides an aggregate representation based on the box-plot statistical tool [McGill et al., 1978; Tukey, 1970, 1977] showing the variability of the three DCG curves calculated either on all the topics considered by an experiment or on those selected by the user. In order to keep the visualization as clear as possible, instead of representing single box-plots concerning the distribution of the performances across different topics for each rank position, a line joining the corresponding points of the various box-plots at different rank positions is used.

Therefore, in the visualization, there are five different curves: upper limit, upper quartile, median, lower quartile, and lower limit. All these curves are determined for the ideal, the optimal and the experiment cases. For each case, the area between lower and upper quartile is color filled in order to highlight the central area of the analysis – what is typically represented with a box in a box-plot. Following this rationale the median lines are thicker in order to be different to the upper/lower quartile ones represented with normal thickness and to upper/lower limit ones represented with dashed lines.

Moreover, the visualization allows user to interactively choose the topics to whose performances have to be aggregated in order to support the exploration of alternative retrieval scenarios.

For example, this kind of visualization support users in understanding whether the optimal and experiment areas overlap for a good extent and the median curve of the experiments tends to the one of the optimal, indicating that the overall performances of a run are close to the best that can be done with that set of retrieved documents. Understanding whether this is a result good enough or not, it is then a matter of understanding how these areas overlap with the one of the ideal curves.

This visualization has been first proposed in [Angelini et al., 2012c] as a means to offer users an overall view of the systems performances. In this deliverable, we improve it by framing it in the context of a whole analysis workflow and adding to it further interaction by allowing users to dynamically select different subsets of topics to be explored.

2.3 Failing Documents Identification

As it as been discussed in Section 1, failure analysis is a fundamental but demanding activity. Moreover, looking at a performance curve, as the DCG curve is, it is not always easy to spot what are the critical regions in a ranking. For example, as explained in Section 2.1, DCG is a not-decreasing monotonic function which increases only when you find a relevant document in the ranking. However, when DCG does not increase, this could be due to two different reasons: either you are in an area of the ranking where you are expected to put relevant documents but you are putting a not relevant one and thus you do not gain anything; or, you are in an area of the ranking where you are not expected to put relevant documents and, correctly, you are putting a not relevant one, still gaining nothing. So, basically, when DCG stays constant, it is not immediate to understand whether this is due to a failure of the system which is not retrieving relevant documents while it would still be expected to do so, or whether the system is performing properly since there would be nothing to gain at that rank position.

In order to overcome this and similar issues, we introduce two indicators, *Relative Position (RP)* and *Delta Gain (ΔG)*, which allow to quantify and explain what happens at each rank position and are paired with a visual counterpart which eases the exploration of the performances across the ranking, immediately grasping the most critical areas.

RP quantifies the effect of misplacing relevant documents with respect to the ideal case, i.e. it accounts for how far a document is from its ideal position. Indeed, the ideal case represents an ordering of the documents in the ground-truth in decreasing degree of relevance putting, for example, all the highly relevant documents first, followed by the partially relevant ones, and then the not relevant ones, thus creating contiguous intervals of documents with the same degree of relevance. In RP, zero values denote documents which are within their ideal interval; positive values denote documents which are ranked below their ideal interval, i.e. documents of higher relevance degree that are in a position of the ranking where less relevant ones are expected; and, negative values denote documents which are above their ideal interval, i.e. less relevant documents that are in a position of the ranking where documents of higher relevance degree are expected. Overall, the greater is the absolute value of RP, the bigger is the distance of the document from its ideal interval.

RP eases the interpretation of the DCG curve since, for example, a constant value of DCG implies a negative value of RP, if this is due to a failure of the system which is not retrieving relevant

documents while it would still be expected to do so, or a zero value of RP, if the system is performing properly since there would be nothing to gain at that rank position.

ΔG quantifies the effect of misplacing relevant documents with respect to the ideal case in terms of the impact of the misplacement on the gain at each rank position. In ΔG zero values indicate documents which are within their ideal interval and are gaining what is expected from them; negative values denote documents that are ranked above their ideal interval and are causing a local loss in the gain with respect to what could have been achieved; positive values indicate document that are ranked below their ideal interval and are causing a local profit in the gain. ΔG supports the interpretation of DCG curves in a similar way to RP but providing the additional information about how much gain/loss happened at each rank position with respect to the ideal case.

These two indicators are paired with a visual counterpart that makes it even easier to quickly spot and inspect critical areas of the ranking. Two bars are added on the left of the visualization, as shown in Figure 1.(c): one for the RP indicator and the other for the ΔG indicator. These two bars represent the ranked list of results with a box for each rank position and, by using appropriate color coding to distinguish between zero, positive and negative values and shading to represent the intensity, i.e. the absolute value of each indicator, each box represents the values of either RP or ΔG.

For example, in this way, looking at the bars and their colors, the user can immediately identify not relevant documents which have been ranked in the positions of relevant ones. Then, the visualization allows them to inspect those documents and compare them with the topic at hand in order to make hypothesis about causes of a failure. This greatly reduces the effort needed to carry out failure analysis because: (i) users are not requested to interpret the not always intuitive DCG curve to identify potential problems; (ii) users can grasp the critical areas of the ranking by means of color coding and shading and focus on them, instead of scrolling through almost each rank position to individuate potential problems; (iii) once a critical area has been identified, the visualization allows to interactively inspect the failing documents and to readily make guesses about the causes of the failure.

The RP and ΔG indicators have been first proposed in [Ferro et al., 2011] together with the idea of exploiting them for creating a visual tool for exploring the performances of an IR system. Here they are fully formalized in the context of the proposed analytical framework, they are made part of a complete workflow and not used in isolation, and the visualization backing them is improved in terms of interaction with the user and possibility of exploring the retrieved documents.

This visualization based on the RP and ΔG has also been exploited in [Di Buccio et al., 2011] to develop an iPad based version of it with the purpose of exploring the following scenarios where having interaction and visualization via a tablet can be an added value: (i) a researcher or a developer is attending the workshop of one of the large-scale evaluation campaigns and s/he wants to explore and understand the experimental results as s/he is listening at the presentation discussing them; (ii) a team of researchers or developers is working on tuning and improving an IR system and they need tools and applications that allow them to investigate and discuss the performances of the system under examination in a handy and effective way. This work is not reported here since it is out of the scope of the present deliverable.

Finally, the RP indicator opened the way for a designing and developing a brand new metric,

called *Cumulated Relative Position (CRP)*, for evaluating the performances of an IR system [Angelini et al., 2012b]. The CRP metric is the cumulative sum of the RP indicator and it shares a similar approach to the DCG measures, i.e. cumulating what happened up to a given rank position. However, CRP and DCG have two different user models: the former assumes a "lazy" user who would like to receive all the relevant information with (almost) no effort and thus measures how much "space" the user is forced to cover in order to get the desired information; the latter assumes a more or less "committed" user who is willing to perform a certain amount of effort to gain relevant information. While as robust and as sensitive as other IR metrics, CRP is lowly correlated to them, thus offering an alternative and complementary view point on the system behavior. Moreover, with respect to DCG measures, CRP has the further property of being summarized by three indicators that provide a single number which condense the behavior of the whole CRP curve and system performances. A deeper discussion on CRP is out of the scope of the present deliverable but this brief introduction to it should suggest the reader how powerful is the formal analytical framework proposed in this deliverable and how it is possible to stem new research direction from it, also beyond its original purposes.

2.4 Failing Topics Identification

The techniques described in the previous section support and ease failure analysis at the topic level and allow users to identify and guess possible causes for wrongly ranked documents. However, it is often needed to grasp an overall picture for a whole run in order to understand if the critical areas of the ranking identified in the previous step are an isolated case concerning just a given topic or they are common to more topics or even a whole run and thus they have a greater impact.

The visualization of Figure 1.(d) merges the approaches of the visualizations presented in Figure 1.(b) and Figure 1.(c): it allows users to assess the distribution of the performances of the ideal, optimal, and experiment curves over a set of selected topics or the whole run and it adds the bars reporting the RP and ΔG indicators to ease the interpretation of the performance distribution.

In particular, this visualization offers user different strategies according to which RP and ΔG values of the experiment are aggregated for a given rank position over the selected set of topics: for example, the user can choose to compute the average, the median, a quartile, and so on of the RP and ΔG values. In this way, users can not only interactively explore different features of the performance distribution but they also can align the way in which RP and ΔG values are aggregated to the specific area of the performance distribution they are focusing on. Suppose, for example, that the user is exploring the lower quartile of the performance distribution because his goal is to ensure a minimum level of performances across the topics instead of having some performing very high and some very low, in this case it is preferred to aggregate RP and ΔG values by their lower quartile in order to have a kind of "magnification" of the behavior of the corresponding areas highlighted in the DCG curves.

2.5 Document Movement Estimation

Suppose that thanks to the steps from Figure 1.(a) to Figure 1.(d), the user has formulated an hypothesis about the possible cause of a failure. For example, the stop list of the SMART system [Salton,

1971] removes 571 words among which the term “wonder”; this would hamper the retrieval and ranking of documents about the singer Stevie Wonder, even if the term “wonder” appeared both in the topics and in the documents since it is removed via the stop list.

What usually happens at this point is that the user implements a modification of the stop list and related components in his IR system, performs another round of experiments, computes the performances of the system, and inspects whether that modification has produced the expected effect or not, i.e. ranking higher the documents talking about Stevie Wonder, possibly not hampering the performances on the other topics.

The objective of the visualization of Figure 1.(e) is to provide a rough estimation of what could be the impact of fixing a possible failure on the performances in order to assess if it might be worth or not implementing it. This means neither that we suggest the possible cause of a failure, e.g. a stop list removing too many terms, nor that we suggest how this can be fixed/implemented in the actual system, since these activities can be performed only by the user. We provide a tool that, driven by the intuition of the user and guided by the interaction with him, supports the user by showing him the effects of a modification on the rank of a document indicated by the user and how such modification may interact with the ranks of other documents for the same topic.

In particular, according to the example above, we foresee the following scenario. By performing failure analysis, the user hypothesizes that the problem is the stop list which removes the term “wonder”. At the same time, the user hypothesizes that, if he fixes that failure, a given relevant document would be ranked higher than it is in the current system. What visualization of Figure 1.(e) offers to the user is: (i) the possibility of dragging and dropping the target document in the desired position of the rank; (ii) the estimation of what other documents would be affected by the movement of the target document and how the overall ranking would be modified; (iii) the computation of the system performances according to the new ranking. Indeed, if the stop list is fixed, not only the target document identified by the user would be affected by this modification but also other documents which, for example, contain the term “wonder” and which were not examined by or known to the user. Therefore, moving a single target document would actually cause the movement and repositioning of a whole set of documents that share features impacted by the same modification which will affect the target document selected by the user. These complex interactions between documents may generate modifications on the ranking that go well beyond what imagined by the user when moving the single target document and which are definitely hard to be guessed by him. Thus, the contribution of the visualization of Figure 1.(e) is to automatically point out to the user all these complex interactions and how they affect the overall ranking.

In order to carry out the scenario just envisioned, VATE^2 needs: (i) to understand which documents would be affected by the movement of a target document indicated by the user; (ii) to adopt a strategy for simulating which the movement of the documents in the ranked list could be. Both of these items require a quite complex analytical model and computations.

We will now introduce two possible and straightforward instantiations for the two items above which demonstrate the overall feasibility of the proposed approach, while it is clear that more sophisticated ones are possible: they are beyond the scope of this deliverable and represent a valuable future work which can rely on the solid bases provided by the proposed formal analytical framework for keeping the overall coherence and comparison.

Document clustering has been used in IR systems for many years and a wide range of alternative techniques has been developed to perform it [Carpineto et al., 2009; Manning et al., 2008; Salton and McGill, 1983; van Rijsbergen, 1979; Willett, 1988]. The original goal of document clustering was to improve efficiency of search by reducing the number of documents that needed to be compared to the query [Salton, 1971]. However, it was soon realized that document clustering could have been used to produce an improvement on the effectiveness of an IR system [Hearst and Pedersen, 1996; Jardine and van Rijsbergen, 1971; Voorhees, 1985]. This line of research stemmed from the *cluster hypothesis* [van Rijsbergen, 1979]: "closely associated documents tend to be relevant to the same requests", which was used to justify the fact that documents in a given cluster could all be relevant to a given query and that the query could be compared against the centroid of the cluster instead of every single document.

Document clustering is exploited in VATE^2 in order to understand which documents would be affected by the movement of a target document indicated by the user, using a variation of the cluster hypothesis we could call the *failure hypothesis*: "closely associated documents tend to be affected by the same failures", stating the common intuition that a given failure will affect documents with common features, in our example all the documents where the term "wonder" appears, and, consequently, that a fix for that failure will have an effect on the documents sharing those common features.

Therefore, VATE^2 uses document clustering to select all the potential documents which can be affected by a fix, to show to the user which other documents will be involved by the fix he is guessing, and to provide an estimation of the impact of this fix on the performances for a given topic. However, before performing document clustering, we need to consider that, as introduced in the beginning of the section, we do not have any information about the internals of the system but we can only access its inputs, i.e. the topics, and its outputs, i.e. the ranked result lists produced by the system for each topic. At this point we have two alternatives, discussed below.

The first alternative is to use a standard IR system, such as Apache Lucene[3] [McCandless et al., 2010] or Terrier[4] [Ounis et al., 2006], but this would mean a consistent mismatch with respect to the system under examination, because a standard IR system would almost certainly use different components (tokenizers, stop lists, stemmers, ...) and different weighting schemes, or components configured and tuned in a different way with respect to the system under examination. As a consequence, the document clusters produced using a standard IR system would almost certainly be different from the ones which the IR system under examination would produce and they would be almost not representative of the system behavior (and its failures).

The second alternative is to try to learn a model representing the behavior and functioning of the system and then to use the learned model to create the document clusters in a way that is as adherent as possible to the actual document clusters which would be created by the IR system under examination. In particular, we use regression trees [Breiman et al., 1984; Ruggeri et al., 2013] to learn a model of the IR system under examination because of their non-parametric nature and capability of handling an high number of classes. Then, we use the learned model in order to compute document-to-document similarity and produce clusters according to the techniques described

[3] http://lucene.apache.org/
[4] http://terrier.org/

in [Willett, 1988].

Note that the choice of exploiting machine learning techniques is not driven only by necessity, i.e. the fact we cannot assume to know the system internals but we have to consider it as a "black box" only whose inputs and outputs are known, but it is also a guarantee of "fairness" and consistent behavior when analyzing a wide range of systems and technical solutions. Indeed, if we had built the analytical framework around specific IR models or assuming some specific system internals, we would have favored a specific class of systems and techniques. On the contrary, learning a model of the system from its inputs and outputs ensures a consistent treatment of different systems, not biased by the specific features of a given class of systems, and, even if this choice represents an approximation of the actual system behavior, the introduced error is systematic and comparable across different system categories, thus not penalizing only a specific one. Moreover, as already stated, the proposed formal analytical framework opens the way at exploring alternatives, in this case different machine learning techniques which may be more effective or appropriate than the used one for learning an IR system model.

The learned document clusters are then exploited in the interactive visualization of Figure 1.(e): after failure analysis, the user can select a document whose rank can be improved by an hypothesized fix, and drag and drop it in the rank position that the fix should allow it to achieve. At this point, the visualization highlights the cluster of documents which would be probably affected by the same fix, according to the failure hypothesis stated above, and animates the movement of all these documents to their new rank positions. Finally, the DCG curves corresponding to the new ranking are computed allowing the user to get an estimation of what the impact of the supposed fix can be on the system performances.

The movement of the document and the related document cluster happens according to a straightforward algorithm that tries to move the documents in the cluster of the same amount of positions as the document dragged and dropped by the user. However, this is not always possible since, for example, a document in the cluster might be ranked higher than the document selected by the user and may not exist enough space on the top of the ranking to place it; in this and similar cases, the movement algorithm "compresses" the movement of the documents in the cluster, approximating at its best the user intent. As in the case of the learning algorithm, also the movement algorithm can be replaced by more sophisticated versions which are left for future work.

The idea of interactively identifing and moving the whole cluster of documents which may be affected by a fix guessed by the user has been first proposed in [Angelini et al., 2012a,c] and here its modeling is improved and formalized in the context of the full analytical framework. Moreover, the learning algorithm has been changed from [Angelini et al., 2012a,c] where learning to rank techniques [Liu, 2009; Tiu et al., 2010] were used, because these techniques not only learn a model of an IR system but they also try to contextually improve and optimize it with respect to a selected performance metric, while our purpose here is just to learn a model of the IR system under examination as adherent as possible to it.

2.6 Domino Effect Estimation

The what-if analysis of Figure 1.(e) can be iterated several times for a given topic. However, the effects of the modifications performed with the visualization of Figure 1.(e) are limited to the topic

under inspection. On the other hand, it would be useful to get an overall estimation of how these modification would affect the whole set of topics and this is exactly the purpose of the the visualization of Figure 1.(f).

In this step, the IR system model is re-learned using the results list modified by the user according to the what-if analysis of the previous step with the overall goal of understanding the impact that these modifications may have also on the other topics, which were not considered in the analysis. This impact can be either beneficial or detrimental to the other topics and we call it “domino effect”.

Therefore, VATE^2 uses the re-learned model to re-compute all the result lists for all the topics in the experiment and then compares the performances of the original run with the one produced by the user modifications. This visualization uses the same performance distribution analysis strategy introduced in visualizations of Figure 1.(b) and 1.(d) in order to quickly assess if the hypothesized modifications have an overall beneficial or detrimental effect on the whole run or a subset of selected topics.

The idea of taking into consideration the domino effect has been first proposed in [Angelini et al., 2012a,c] but there it was limited to analyzing the effect which the modifications made by the user on a topic can have on another topic selected by the user. Here, we extend and improve that estimation by applying it to the whole set of topics comprising a run and using it in the context of the analysis of the performance distribution.

3 Formal Analytical Framework

3.1 Preliminary Concepts

We formalize the basic notions regarding experimental evaluation in IR starting from the concepts of relevance and degree (or grade) of relevance of a document with respect to a topic. Then, leveraging on these two concepts we define the basic concepts of ground truth, recall base, and relevance score.

As discussed above, the notion of relevance and degree (or grade) of relevance of a document with respect to a topic is fundamental to experimental evaluation.

Definition 3.1. Let REL be a finite set of **relevance degrees** and let $\preceq$ be a *total order relation* on REL so that

$$(REL, \preceq)$$

is a totally ordered set.

We call **non-relevant** the relevance degree $nr \in REL$ such that

$$nr = \min(REL)$$

Being a finite totally ordered set, the set of relevance degrees admits the existence of a minimum and a maximum.

Consider the following example: the set $REL = \{\mathtt{nr}, \mathtt{pr}, \mathtt{fr}, \mathtt{hr}\}$ contains four relevance degrees where `nr` stands for "non relevant", `pr` for "partially relevant", `fr` for "fairly relevant" and `hr` stands for "highly relevant". We can define the following total order relation:

$$\begin{aligned} \preceq \ \subset REL \times REL = \ & \{(\mathtt{nr},\mathtt{nr}),(\mathtt{nr},\mathtt{pr}), \quad (\mathtt{nr},\mathtt{fr}),(\mathtt{nr},\mathtt{hr}) \\ & (\mathtt{pr},\mathtt{pr}),(\mathtt{pr},\mathtt{fr}), \quad (\mathtt{pr},\mathtt{hr}),(\mathtt{fr},\mathtt{fr}), \\ & (\mathtt{fr},\mathtt{hr}),(\mathtt{hr},\mathtt{hr})\} \end{aligned}$$

which lead to the ordering $\mathtt{nr} \preceq \mathtt{pr} \preceq \mathtt{fr} \preceq \mathtt{hr}$ one would expect for the relevance degrees introduced above.

The ground truth function associates a relevance degree rel, i.e. a relevance judgment, to each document d for each topic t, where a document is the basic information unit considered in experimental evaluation and a topic is a materialization of a user information need.

Definition 3.2. Let D bet a finite set of *documents* and T a finite set of *topics*. The **ground truth** is a function

$$\begin{aligned} \mathrm{GT}: \quad T \times D \quad &\to \quad REL \\ (t,d) \quad &\mapsto \quad rel \end{aligned}$$

From the definition of ground truth we can derive the related concept of recall base which is the total number of relevant documents for a given topic t; a relevant document is meant by any document with relevance degree above non-relevant.

Definition 3.3. The **recall base** is a function

$$\begin{aligned} \mathrm{RB}: \quad T &\rightarrow \mathbb{N} \\ t &\mapsto RB_t = \left|\left\{d \in D \mid \mathrm{GT}(t,d) \succ \min(REL)\right\}\right| \end{aligned}$$

3.2 Runs

Now, we stated all the definitions necessary to define a run as a set of vectors of documents, where each vector $\mathbf{r}_t$ of length N represents the ranked list of documents retrieved for a topic t with the constraint that no document is repeated in the ranked list.

Definition 3.4. Given a natural number $N \in \mathbb{N}^+$ called the *length of the run*, a **run** is a function

$$\begin{aligned} \mathrm{R}: \quad T &\rightarrow D^N \\ t &\mapsto \mathbf{r}_t = (d_1, d_2, \ldots, d_N) \end{aligned}$$

such that $\forall t \in T,\ \forall j,k \in [1,N] \mid j \neq k \Rightarrow \mathbf{r}_t[j] \neq \mathbf{r}_t[k]$ where $\mathbf{r}_t[j]$ denotes the j-th element of the vector $\mathbf{r}_t$, vectors start with index 1, and vectors end with index N.

The relevance score associates to each element of a run vector the corresponding relevance degree. It is worth noting that, in general, the relevance score is not injective since two different run vectors for two different topics may map to the same vector of relevance degrees; this is also intuitive from the fact that $|D| \gg |REL| \Rightarrow |D|^N \gg |REL|^N$ and so there are much more vectors of documents than vectors of relevance degrees.

Definition 3.5. Given a run $\mathrm{R}(t) = \mathbf{r}_t$, the **relevance score** of the run is a function:

$$\begin{aligned} \widehat{\mathrm{R}}: \quad T \times D^N &\rightarrow REL^N \\ (t, \mathbf{r}_t) &\mapsto \widehat{\mathbf{r}}_t = (rel_1, rel_2, \ldots, rel_N) \end{aligned}$$

where

$$\widehat{\mathbf{r}}_t[j] = \mathrm{GT}\big(t, \mathbf{r}_t[j]\big)$$

From the relevance score it is straightforward to introduce the new definition of relevance weight of a run.

Definition 3.6. Let $W \subset \mathbb{Z}$ be a totally ordered finite set of integers, REL be a finite set of relevance degrees and let $\mathrm{RW}: REL \rightarrow W$ be a monotonic function which maps each relevance degree ($rel \in REL$) into a relevance weight ($w \in W$).

Then, given a run $\mathrm{R}(t) = \mathbf{r}_t$ its **relevance weight** is a function:

$$\begin{aligned} \widetilde{\mathrm{R}}: \quad T \times D^N &\rightarrow W^N \\ (t, \mathbf{r}_t) &\mapsto \tilde{\mathbf{r}}_t = (w_1, w_2, \ldots, w_N) \end{aligned}$$

where

$$\tilde{\mathbf{r}}_t[j] = \mathrm{RW}\left(\widehat{\mathbf{r}}_t[j]\right)$$

The relevance weight function $\widetilde{\mathrm{R}}$ can also be defined as the composition between the relevance score function $\widehat{\mathrm{R}}$ and the the function RW – i.e. $\widetilde{\mathrm{R}} = \widehat{\mathrm{R}} \circ \mathrm{RW}$.

The relevance score as well as the relevance weight allow us to discern between two main different types of run: the ideal and the optimal run. We define the ideal run for a given topic $t \in T$ as the run where all the relevant documents for t are arranged in the vectors in descending order according to their relevance score. Therefore, the ideal run contains the best ranking of all the relevant documents for each considered topic. In the following definition, condition (1) ensures that all the relevant documents are retrieved in the ideal run while condition (2) guarantees that they are in descending order of relevance thereby forming intervals of descending quality. Note that the ideal run actually defines a whole set of permutations of the documents with the same relevance degree.

Definition 3.7. The **ideal run** $\mathrm{I}(t) = \mathbf{i}_t$ is a run which satisfies the following constraints

$$\begin{aligned} &\textit{(1) recall base:} && \forall t \in T, \left|\left\{ j \in [1,N] \mid \mathrm{GT}(t, \mathbf{i}_t[j]) \succ \min(REL) \right\}\right| = RB_t \\ &\textit{(2) ordering:} && \forall t \in T, \forall j,k \in [1,N] \mid j < k \Rightarrow \widehat{\mathbf{i}}_t[j] \succeq \widehat{\mathbf{i}}_t[k] \end{aligned}$$

From definition 3.7, it follows that, for each topic $t \in T$, the relevance score of the ideal run $\widehat{\mathbf{i}}_t$ is a monotonic non-increasing function by construction. Therefore, the maximum of the function is at $j = 1$ and it is equal to $\widehat{\mathbf{i}}_t[1] = \max(REL)$ and the minimum is at $j = N$ and it is equal to $\widehat{\mathbf{i}}_t[N] = \min(REL)$.

Following the same line of reasoning, we define the optimal run as a variant of the ideal one. Indeed, the ideal run ranks in descending order all the relevant documents for a given topic t_i and it is the same for every possible run $R(t_i) = \mathbf{r_{t_i}}$; whereas, the optimal run directly depends by a given run $R(t_i) = \mathbf{r_{t_i}}$. Indeed, the optimal run orders all documents retrieved by $\mathbf{r_{t_i}}$ in descending order according to their relevance score. This means that the ideal run is the best possible run for a given topic, whereas the optimal run is the best ordering of the documents retrieved by a run. In the following definition, given a run $\mathbf{r_t}$ and its optimal run $\mathbf{o_{r_t}}$ with the same length, condition (1) guarantees that they contain the same documents and condition (2) guarantees that the documents in $\mathbf{o_t}$ are in descending order of relevance.

Definition 3.8. Given a run $R(t) = \mathbf{r_t}$ with length $N \in \mathbb{N}^+$, its **optimal run** $\mathbf{o}_{r_t}$ is a run with length N, which satisfies the following constraints:

$$\begin{aligned} &\textit{(1) retrieved documents:} && \forall t \in T, \forall j,k \in [1,N], \exists! \mathbf{o_{r_t}}[k] \mid \mathbf{r}_t[j] = \mathbf{o_{r_t}}[k] \\ &\textit{(2) ordering:} && \forall t \in T, \forall j,k \in [1,N] \mid j < k \Rightarrow \widehat{\mathbf{o_{r_t}}}[j] \succeq \widehat{\mathbf{o_{r_t}}}[k] \end{aligned}$$

From this definition we can see that the ideal run depends only by the given topic, whereas the optimal run depends by the topic and by a given run. The ideal run tells us the best possible ranking an hypothetic system can return for a given topic, whereas the optimal run tells us the best ordering of the results returned by a real system. When we compare the ideal run with an experimental run, we understand how far the system which produced the experimental run is from the perfect retrieval and how many relevant documents it missed; when we compare the optimal run with an experimental run produced by a tested system, we determine how far the tested system is from a perfect ordering of the retrieved documents.

3.3 (Discounted) Cumulated Gain Metrics

The evaluation metrics considered in this deliverable exploit the idea that documents are divided in multiple ordered categories [Järvelin and Kekäläinen, 2002b] and, specifically, they are a family of metrics composed by the *Cumulated Gain (CG)* and its discounted version which is the DCG; both CG and DCG have normalized versions called *Normalized Cumulated Gain ((n)CG)* and *Normalized Discounted Cumulated Gain ((n)DCG)* respectively. In VATE^2 we provide the possibility of analyzing the experimental runs on the basis of all the Cumulated Gain metrics; to this end, we can exploit the preliminary definitions given above to formally present them.

Definition 3.9. Let $\mathrm{R}(t)$ be a generic run with length $N \in \mathbb{N}^+$, where $t \in T$ is a given topic, RB_t its recall base, and $j \leq N$, then CG[j] is defined as:

$$\text{CG[j]} = cg_{\mathbf{r}_t}[j] = \sum_{k=1}^{j} \tilde{\mathbf{r}}_t[k]$$

The normalized version of the cumulated gain at position j – i.e. nCG[j] – is defined as the ratio between the CG of $\mathrm{R}(t)$ and the CG of the ideal run $\mathrm{I}(t)$:

$$\text{nCG[j]} = \frac{cg_{\mathbf{r}_t}[j]}{cg_{\mathbf{i}_t}[j]}$$

The visualization of (n)CG curves are useful for the analyses conducted via VATE^2 because they are not monotonically non-decreasing curves like the CG ones are. (n)CG curves allow for an easier analysis of the performances of a run at earlier ranks than CG curves, but the normalized ones lack of the straightforward interpretation of of the gain at each rank given by the CG curves. For this reason, it is important to be able to pass from a curve to the other dynamically in order to catch the differences between different runs.

To this purpose, the discounted cumulative versions of these metrics are important to give another view of the run, thus providing additional analytic possibilities to the analyst. Indeed, the discounted versions realistically weight down the gain received through documents found later in the ranked results, thus giving more importance to the early positions in ranking. DCG measures assign a gain to each relevance grade and for each position in the rank a discount is computed. Then, for each rank, DCG is computed by using the cumulative sum of the discounted gains up to that rank. This gives rise to a whole family of measures, depending on the choice of the gain assigned to each relevance grade and the used discounting function.

Definition 3.10. Given a run $\mathrm{R}(t)$ with length $N \in \mathbb{N}^+$ and a log base $b \in \mathbb{N}^+$, for all $k \in [1,N]$ the **discounted gain** is defined as:

$$dg^b_{\mathbf{r}_t}[k] = \begin{cases} \tilde{\mathbf{r}}_t[k] & \text{if } k < b \\ \dfrac{\tilde{\mathbf{r}}_t[k]}{\log_b k} & \text{otherwise.} \end{cases}$$

So, the discounted cumulative gain at j (DCG^b[j]) is defined as:

Definition 3.11. Let $\mathrm{R}(t)$ be a generic run, then DCG[j] is defined as:

$$\mathrm{DCG}[j] = \sum_{k=1}^{j} dg^b_{\mathbf{r}_t}[k]$$

Typical instantiations of DCG measures make use of positive gains (i.e. relevance scores) and logarithmic functions to smooth the discount for higher ranks – e.g. a $\log_2$ function is used to model impatient users while a $\log_{10}$ function is used to model very patient users in scanning the result list. DCG is the most used metric of the cumulated-gain family and VATE^2 mainly leverages on it for the study of system performances while supporting all the other metrics in the family.

Lastly, let us see the normalized version of the discounted cumulative gain (nDCG$^b[j]$) that can be defined as:

$$\mathrm{nDCG}^b[j] = \sum_{k=1}^{j} \frac{dg^b_{\mathbf{r}_t}[k]}{dg^b_{\mathbf{i}_t}[k]}$$

3.4 Correlation Analysis

Given a run, for each one of the presented metrics it is possible to draw three curves: the curve of the run, the optimal run curve and the ideal run curve. VATE^2 enables a thorough study of these curves and their inter-relations; to this end, a significant means is Kendall's τ which estimates the distance between two run rankings [Kekäläinen, 2005; Voorhees, 2001]. Given a run, its optimal ranking and the ideal run, Kendall's τ is useful to determine analytically if it is necessary to re-rank the documents in the run or if it is required to re-query to obtain a new set of results as discussed in Section 2.1.

Basically, given two runs, say $A(t)$ and $B(t)$, Kendall's τ correlation between them is determined, rank-by-rank, calculating how many document pairs are concordant or discordant, where: if at rank $i \in [1,N]$ the document in run $A(t)$ is the same as the document in run $B(t)$, then the pair is said to be concordant, otherwise it is discordant. Kendall's τ is given by the total number of discordant pairs subtracted from the number of concordant ones, then divided by the total number of pair combinations.

Definition 3.12. Let

$$A(t) = \mathbf{a}_t = (d^a_1, d^a_2, \ldots, d^a_N)$$

and

$$B(t) = \mathbf{b}_t = (d_1^b, d_2^b, \ldots, d_N^b)$$

be two runs, where $\forall j \in [1,N]$,

$$(d_j^a, d_j^b) = 1 \Leftrightarrow d_j^a = d_j^b$$

and

$$(d_j^a, d_j^b) = -1 \Leftrightarrow d_j^a \neq d_j^b$$

Then,

$$\tau = \frac{\sum_{j=1}^{N}(d_j^a, d_j^b)}{\frac{1}{2}N(N-1)}$$

Kendall's τ varies in the $[-1,1]$ range, where $\tau = 1$ means that the two compared rankings are equal, $\tau = -1$ means that one ranking is the reverse of the other (i.e. a perfect disagreement), and $\tau = 0$ means that the two compared ranking are independent one from the other.

3.5 Relative Position and Delta Gain

Relative Position (RP) and Delta Gain (ΔG) are the two metrics on which VATE^2 bases the "failing documents identification" (Section 2.3) and the "failing topics identification" (Section 2.4). They are complementary one to the other, RP quantifies the misplacement of a document in a run ranking with respect to the ideal ranking, and ΔG estimated the effect of this misplacement in the overall calculation of the DCG.

In order to introduce RP we need to define the concepts of minimum rank and maximum rank of a given relevance degree building on the definition of ideal run. Indeed, the minimum rank is the first position at which we find a document with relevance degree equal to rel while the maximum rank is the last position at which we find a document with relevance degree equal to rel in the ideal run.

Definition 3.13. Given the ideal run $\mathrm{I}(t)$ and a relevance degree $rel \in REL$ such that $\exists j \in [1,N] \mid \widehat{\mathbf{i}}_t[j] = rel$, the **minimum rank** and the **maximum rank** are, respectively, a function

$\min_{\mathbf{i}_t}(rel)$:

$$\begin{aligned} T \times D^N \times REL &\rightarrow \mathbb{N}^+ \\ (t, \mathbf{i}_t, rel) &\mapsto \min\Big(\big\{j \in [1,N] \mid \widehat{\mathbf{i}}_t[j] = rel\big\}\Big) \end{aligned}$$

$\max_{\mathbf{i}_t}(rel)$:

$$\begin{aligned} T \times D^N \times REL &\rightarrow \mathbb{N}^+ \\ (t, \mathbf{i}_t, rel) &\mapsto \max\Big(\big\{j \in [1,N] \mid \widehat{\mathbf{i}}_t[j] = rel\big\}\Big) \end{aligned}$$

Note that, by construction, we have: $\min_{\mathbf{i}_t}\left(\max(REL)\right)=1$; $\min_{\mathbf{i}_t}\left(\min(REL)\right)=RB_t+1$; $\max_{\mathbf{i}_t}\left(\min(REL)\right)=N$; and, given a relevance degree $\overline{rel}\in REL$ strictly above $\min(REL)$ and below any other relevance degree, i.e. $\overline{rel}\in REL \mid \overline{rel}\succ\min(REL)\wedge\forall rel_i\in REL, rel_i\neq\min(REL)\Rightarrow\overline{rel}\preceq rel_i$, $\max_{\mathbf{i}_t}\left(\overline{rel}\right)=RB_t$.

We can now introduce the RP metric which points out the instantaneous and local effect of misplaced documents and how much they are misplaced with respect to the ideal case $\mathbf{i}_t$. In the following definition, zero values denote documents which are within the ideal interval; positive values denote documents which are ranked below their ideal interval, i.e. documents of higher relevance degree that are in a position of the ranking where less relevant ones are expected; and, negative values denote documents which are above their ideal interval, i.e. less relevant documents that are in a position of the ranking where documents of higher relevance degree are expected. Note that the greater is the absolute value of RP, the bigger is the distance of the document from its ideal interval.

Definition 3.14. Given a run $\mathrm{R}(t)$, the ***Relative Position (RP)*** is a function

$$\begin{array}{rccl}\mathrm{RP}: & T\times D^N & \rightarrow & \mathbb{Z}^N \\ & (t,\mathbf{r}_t) & \mapsto & \mathbf{rp}_{\mathbf{r}_t}=(rp_1,rp_2,\ldots,rp_N)\end{array}$$

where

$$\mathbf{rp}_{\mathbf{r}_t}[j]=\begin{cases}0 & \text{if } \min_{\mathbf{i}_t}\left(\widehat{\mathbf{r}}_t[j]\right)\leq j\leq\max_{\mathbf{i}_t}\left(\widehat{\mathbf{r}}_t[j]\right)\\ j-\min_{\mathbf{i}_t}\left(\widehat{\mathbf{r}}_t[j]\right) & \text{if } j<\min_{\mathbf{i}_t}\left(\widehat{\mathbf{r}}_t[j]\right)\\ j-\max_{\mathbf{i}_t}\left(\widehat{\mathbf{r}}_t[j]\right) & \text{if } j>\max_{\mathbf{i}_t}\left(\widehat{\mathbf{r}}_t[j]\right)\end{cases}$$

ΔG is a metric which quantifies the effect of misplacing relevant documents with respect to the ideal run. ΔG allows for a deeper comprehension of the behavior of DCG curves indicating, rank-by-rank, how a document contributes to the overall computation of DCG. ΔG has value zero if a document is ranked in the correct position w.r.t. the ideal case, a positive value if it is ranked above its ideal position and a negative value otherwise. The higher is the absolute ΔG value of a document, the bigger its misplacement w.r.t. the ideal ranking.

ΔG explicitly takes into account the effect of the discounted function and it is calculated by exploiting the discounted gain presented in Definition 3.10.

Definition 3.15. Given the ideal run $\mathrm{I}(t)$, a run $\mathrm{R}(t)$, and the discounted gains $dg^b_{\mathbf{i}_t}$ and $dg^b_{\mathbf{r}_t}$ for $I(t)$ and $R(t)$ respectively. Then, **delta gain** (ΔG) is a function:

$$\begin{array}{rccl}\Delta\mathrm{G}: & T\times D^N & \rightarrow & \mathbb{Z}^N \\ & (t,\mathbf{r}_t) & \mapsto & \Delta\mathbf{g}_{\mathbf{r}_t}=(\Delta g_1,\Delta g_2,\ldots,\Delta g_N)\end{array}$$

where

$$\Delta\mathbf{g}[j]=dg^b_{\mathbf{r}_t}[j]-dg^b_{\mathbf{i}_t}[j]$$

3.6 Learning Model

As we have discussed in Section 2.5, IR systems are seen as black boxes in experimental evaluation, because, in most cases, we can analyze the ranking lists produced by a system, but we cannot analyze the system which produced them. This means that we cannot modify a system, run new and diversified tests to understand how the system behaves and how it can be improved. To this end we have to rely only on the outputted ranking lists and from these we need to infer how the system behaves under specific conditions.

Learning to rank is a branch of IR which exploits machine learning algorithms to learn the ranking model of an IR system starting from its ranking lists outputted during the tests on a specific experimental collection. The purpose of learning to rank techniques is to improve the original ranking model in order to obtain better performances or to grip on machine learning to build new and more effective ranking models. In VATE^2 we leverage on these techniques with a slightly different purpose; indeed, we use the produced ranking lists, the experimental collection and a machine learning algorithm to learn a ranking model of a given IR system in order to throughly study it without actually having it available. A ranking model is one of the most complex component of an IR system and it ranges from conventional ranking models comprising boolean models [Salton and McGill, 1983], vector space models [Deerwester et al., 1990; Salton and McGill, 1983], and probabilistic models [Ponte and Croft, 1998; Robertson, 1997] to importance ranking models such as the Hyperlink-Induced Topic Search (HITS) [Agosti and Pretto, 2005; Kleinberg, 1999], the PageRank [Langville and Meyer, 2003], and TrustRank [Ganesan et al., 2004]. Learning such models via machine learning techniques is thus complicated by the wide spectrum of models available, the fact that they are composed by many parameters manually tuned, and they are often combined together [Liu, 2009]. In the following we introduce the state-of-the-art learning to rank framework adopted in VATE^2, for further details the reader can consult [Liu, 2009, 2011].

In IR, most of the state-of-the-art learning to rank algorithms are “feature-based”, which means that they learn the optimal way of combining features extracted from topic-document pairs[5] through a process called “discriminative training” [Liu, 2009]. Feature-based means that the topic-document pairs under investigation are represented as vectors of features, representing the relevance of documents w.r.t. a given topic. For a given topic $t_i \in T$, its associated document $d_j \in D$ can be represented as a vector of features $\mathbf{x}_j^{(i)} = \Phi(t_i, d_j)$, where Φ is a feature extractor. We can divide the typical features used in learning to rank into three main categories: document-based, topic-based, and model-based. Document-based features are extracted from the given document such as term-frequencies (TF) of specific terms (i.e. number of occurrences of a term in a document), inverse term frequency (IDF) which is the number of occurrences of a term in the collection of documents, TF in the body of a document, TF in the title, the combination of TF and IDF, and the length of the document. Topic-based features are the same as the document-based but calculated on the text of the topic; they contain also the relationship between a topic and a document such as the number of occurrences of a topic term in a document. Furthermore, these features can be combined together for generating other features, for instance can be given by the logarithm of the TF of a term in a document summed with the TF of the same term in the related topic. Model-based features are

[5] In literature they are also referred as query-document pairs.

the output of ranking models such as the BM25 model, or the PageRank model or the relationship between a document and other documents in the collection.

In VATE^2 we adopt document-based and topic-based features and we do not consider the model-based ones. This choice derives from the fact that our goal is to learn the ranking model of a system in the most reliable way and not to improve their performances. Model-based features are oriented at improving a ranking model by adding peculiarities of different ranking models and thus, they are out-of-scope, or even misleading, for the purposes of VATE^2. The most used and reliable list of features used in learning to rank framework is provided by the LEarning TO Rank (LETOR)[6] initiative run by Microsoft Research and proposed by Liu et al. in [Liu et al., 2007]. Currently, VATE^2 has been tested on a well-known test collection provided by *Text REtrieval Conference (TREC)*, which is the TREC7 Ad-Hoc Track [Voorhees and Harman, 1999]. The original TREC7 collection has binary relevance judgments, but to calculate cumulated-gain metrics we need graded relevance ones, for this reason we employ 21 topics from the original TREC7 where the documents reassessed, using graded relevance degrees, by [Sormunen, 2002] are used. From the LETOR list we selected twenty document-based and topic-based features which apply to TREC7 collection; indeed, this collection is composed by newspaper articles which do not contain any URL, thus, for instance, all the URL-based LETOR features (e.g. the number of slashes in the URLs) cannot be extracted and used. Most of the features are TF, IDF and their combinations for a total of 636 features for each topic-document pair.

The TREC7 test collection is composed by 21 topics and $566,077$ documents, for a total of about 12 millions of topic-document pairs from which we extracted about 8 billions features. As feature extractor Φ we used the Terrier v3.5 search engine[7] which allows us to extract all the features in the LETOR list. Each one of the 54 IR systems which participated to TREC7 submitted 21 runs $\{r_{t_1},\ldots,r_{t_{50}}\}$ of length 1000, one for each topic $t_i \in \{t_1,\ldots,t_{21}\}$ in the collection. For each topic $t_i \in T$, where $i \in [1,21]$ we define 1000 feature vectors: $\mathbf{x}^{(i)} = \{x_j^{(i)}\}_{j=1}^{m^{(i)}}$ where $m^{(i)}$ is the number of document associated with the topic t_i, so in the case of TREC7 $\mathbf{x}^{(i)} = (x_1^{(i)}, x_2^{(i)}, \ldots, x_{1000}^{(i)})$. In VATE^2 each $\{x_j^{(i)}\}_{j=1}^{m^{(i)}}$ contains 636 features.

The featured extracted are employed in the discriminative training process which is composed of four pillars:

1. The *input space* containing the object under investigation, i.e. the feature vectors $\mathbf{x}^{(i)} = \{x_j^{(i)}\}_{j=1}^{m^{(i)}}$.

2. The *output space*, which contains a learning target (i.e. y_i for each $\mathbf{x}^{(i)}$) w.r.t. the learning object. The output space we consider in this context is "task-based" which is highly dependent by the application. For example, employing a regression machine learning algorithm the output space is the space of real numbers $\mathbb{R}$; in classification it is a set of discrete categories. In VATE^2 it is composed by a set of 1000 categories each one indicating the position at which a document is ranked by the tested system for a given topic.

[6] http://research.microsoft.com/en-us/um/beijing/projects/letor/
[7] http://terrier.org/

3. The *hypothesis space*, which defines the class of function mapping the input space in the output space.

4. The *loss function*, which measures to which degree the prediction generated by the hypothesis space is in accordance with the ground truth label.

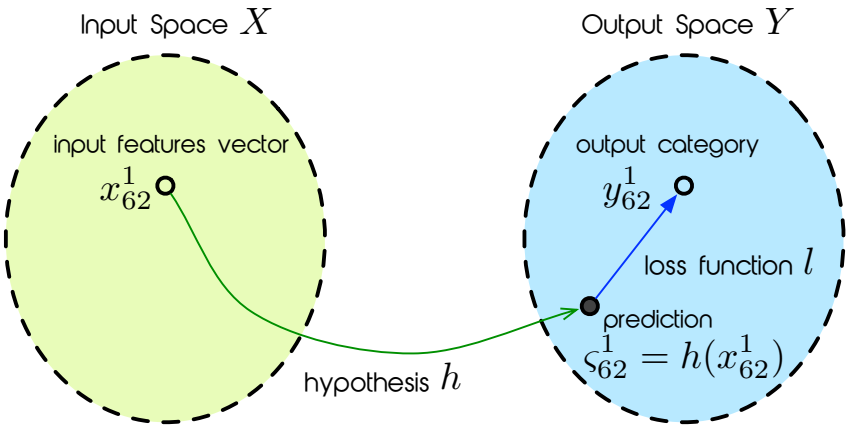


Figure 2: An example based on document 62 related to topic 1 of the general machine learning framework employed by VATE^2.

In a normal learning to rank setting, the output space may be composed by a set of discrete categories each of them representing a relevance degree taken from the ground truth; i.e. for all $i \in [1,21], y_i^j = \mathrm{GT}(t_i, \mathbf{r}_{t_i}[j]) = \widehat{\mathbf{r}}_t[j]$, where, for TREC7, $j \in [1,1000]$. In VATE^2 we do not want to train our learning model to rank a document w.r.t. the ground truth relevance degree, whereas we want that given a topic the learning model ranks the document as the tested system would do. To this purpose, for all $i \in [1,21], y_i^j \in [1,1000]$, thus the hypothesis space has to learn how to map each vector $\{x_j^{(i)}\}_{j=1}^{m^{(i)}}$ into a category y_i^j representing the original position where the tested system ranked document d_j for topic t_i. In Figure 2 we can see a graphical representation of the machine learning framework we employ in VATE^2 where the features vector of document 62 for topic 1 (i.e. x^1_{62}) is mapped by the hypothesis space h into the predicted value $\varsigma^1_{62} = h(x^1_{62})$ which distance from the real value y^1_{62} is determined by the loss function l; the goal of these machine learning algorithm is to optimize l in order to have a predicted value close to the real value.

In Figure 3, we can see the main component of the learning to rank adopted in VATE^2. The training set corresponds to the input space and it is composed by the features vectors $\{x_j^{(i)}\}_{j=1}^{m^{(i)}}$ (for TREC7, $i \in [1,21]$, and $m = 1000$) plus the target value which is a value in the set of output categories. The training set is a feature-based representation of the ranking lists of the tested system and we employ it to learn the ranking model of the system; this is done by exploiting a learning system which produces a learned model (i.e. the hypothesis space). The learned model is then used to process test data producing as output a predicted ranking list. It is possible to employ whichever learning system and VATE^2 is not bound to a specific one; the current version of VATE^2 employs a learning system based on regression trees [Bishop, 2006].

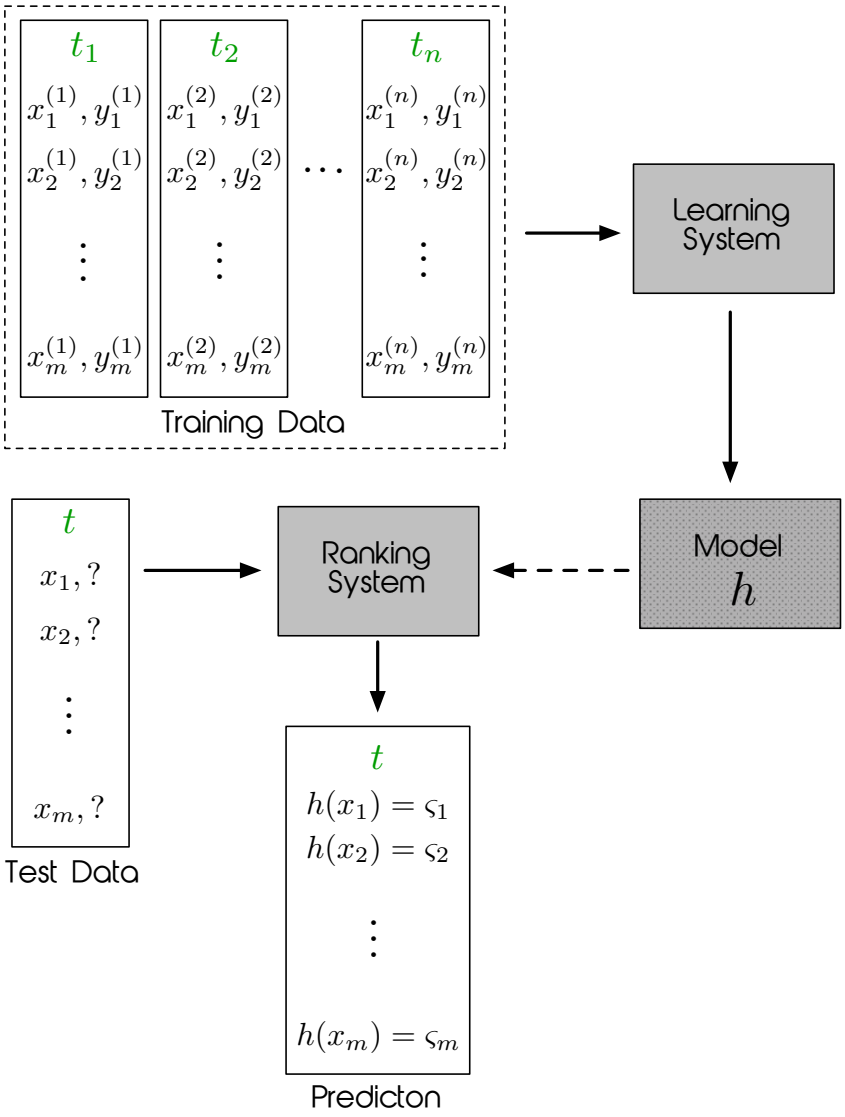


Figure 3: Learning to rank framework adopted in VATE^2.

A regression tree is a specific case of decision tree. It is a learning algorithm that builds a tree structure where the leaves represent class labels and branches represent conjunctions of features that lead to those class labels. In a regression tree the class labels are considered real numbers and thus this algorithm is particularly well-suited for VATE^2, where the features vectors given as test data have to be mapped into a rank position which can be naturally represented by a real number. Furthermore, regression trees are well-suited for the heterogeneous environment of experimental evaluation where VATE^2 works, because they require little data preparation (i.e. they do not require normalization like blank values removal) and they perform well from the execution time point-of-

view [Bishop, 2006].

This general learning to rank framework is used to obtain an approximation of the ranking model of the test system, that is exploited to create cluster of documents used for the "document movement estimation" in VATE^2 leveraging on the *clustering hypothesis* discussed in Section 2.5. In VATE^2 we define the *failure hypothesis* stating that if in a ranking list an analyst spots a document which is misplaced because of a bug in the system, then we suppose that similar documents are affected by the same bug. To this end, we create clusters of similar documents, where the similarity is estimated via the learned ranking model; this means that, given a topic, for every document in a ranking list there exists a cluster of documents which are treated in a similar way by the ranking model.

The construction of the cluster is based on the very framework illustrated in Figure 3. We extract the document-based and topic-based features from the collection of document D by considering every document $d_k \in D$ as a topic. So, for each d_k we have a vector $\mathbf{x}_j^{(k)}$ that is used as test data thus obtaining a ranking list employing the learned ranking model. The higher is a document d_k in the ranking predicted for document d_j used as a topic, the more similar d_j and d_k are from the learned ranking model point-of-view. In general, all the documents in the predicted ranking list belong to the cluster C_j obtained for d_j, but in practice we need to fix a threshold size for the cluster. The current version of VATE^2 uses clusters of size 10 for each document $d_j \in D$; this means that the document between rank 1 and rank 10 in the predicted ranking list for d_j compose the cluster C_j. The document d_j belongs to the cluster C_j; we have experimentally verified that d_j is always ranked in the first position by the learned model.

3.7 Document Movement Estimation

The clusters of documents defined above play a central role in the document movement estimation of VATE^2. Indeed, once a user spots a misplaced document, say d_4, and s/he decides to move it upward or downward, also the ten documents in the C_4 cluster are moved accordingly. The current implementation of VATE^2 employs the simple linear movement strategy described in Section 2.5.

Let us consider a general environment where a run $R(t) = \mathbf{r}_t$ is composed by N documents such that $\mathbf{r}_t = (d_1, d_2, \ldots, d_N)$, where the subscript of the documents indicates their position in the ranking list. As a consequence of the *failure hypothesis*, if we move a document $d_j \in \mathbf{r}_t$ from position j to position k – which means that we move d_j of $\lambda = j - k$ positions – we also move the documents in the cluster C_j of m positions accordingly.

In VATE^2 we implement the simplest movement strategy, which is based on three assumptions:

1. Linear movement: if d_j is moved upward or downward from position j to position k where $\lambda = j - k$, all the documents in its cluster C_j are moved of λ in the same direction. If $\lambda > 0$ then the documents C_j are moved upward; if $\lambda < 0$ they are moved downwards.

2. Cluster independence: the movement of the cluster C_j does not imply the movement of other clusters. This means that when we move d_j in the position of d_k, d_k is influenced by the movement, but the cluster C_k is not. As a consequence, when the cluster of documents C_j is moved, other $|C_j|$ documents are influenced by the movement, but not their clusters.

3. Unary shifting: if C_j is moved by λ positions, then the other documents in the ranking have to make room for them and thus they are moved upward or downward, accordingly to the sign of λ, by one position.

Given these three assumptions, let us see how the movement strategy implemented in VATE^2 works by introducing the definition of index cluster.

Definition 3.16. Let $R(t) = \mathbf{r}_t$ be a run where $d_j \in \mathbf{r}_t, \forall j \in [1,N]$, and C_j be the cluster define for d_j, then the index cluster IC_j is defined as:

$$IC_j = \{i \mid d_i \in C_j\}$$

An index cluster IC_j is a set of integers indicating the positions in the ranking of the documents in C_j. We can say that $\max(IC_j) \in \mathbb{N}^+$ is the index of the document at the higher rank within the cluster, and $\min(IC_j) \in \mathbb{N}^+$ the document in the lower one.

It is also worthwhile to recall how the *sign function* (sgn) works; given $\lambda \in \mathbb{Z}$, the sign function is defined as:

$$\operatorname{sgn}(\lambda) = \begin{cases} +1 & \text{if } \lambda > 0 \\ 0 & \text{if } \lambda = 0 \\ -1 & \text{if } \lambda < 0 \end{cases}$$

Given a document d_j moved from j to k, such that $\lambda = j - k$, in VATE^2 $\lambda \neq 0$, so $\operatorname{sgn}(\lambda) = \dfrac{\lambda}{|\lambda|}$.

In the movement algorithm in VATE^2 when d_j is moved to the place of d_k, d_k is shifted by $-\operatorname{sgn}(\lambda)$, where $\lambda = j - k$. This means that if d_j is moved upward, then $m > 0, \operatorname{sgn}(\lambda) = +1$, and d_k is shifted downwards by one position. Symmetrically, if d_j is moved downwards, then $\lambda < 0, \operatorname{sgn}(\lambda) = -1$, and d_k is shifted upwards by one position. Iteratively, this operation is repeated for all the documents in the cluster C_j.

We have seen that in a general setting, if we move d_j upwards or downwards by λ positions, all the documents in C_j move accordingly of m position. There are cases where this is not possible, because the movement is capped on the top or at the bottom by one or more documents in the cluster. As an example, consider a movement upward of d_j, such that $\lambda > 0$, if there is a document $d_w \in C_j$ such that $w < \lambda$, then d_w cannot be moved upwards by λ position, but at most by w. In this case, we say that the movement is top-capped by d_w; a similar situation happens when we move downwards d_j. Since the movements, most of the times, happen in the upper part of the ranking, the top-capped situation is by far the most common.

In general, if d_j has to be moved upward by λ positions, then the movement is top-capped by $\min(IC_j)$, if $\min(IC_j) - \lambda < 0$; in this case, all the documents in the cluster C_j are moved upward by $\min(IC_j)$ positions. If d_j has to be moved downward by λ positions, then the movement is bottom-capped by $\max(IC_j)$, if $\max(IC_j) > N - \lambda$; in this case, all the documents in the cluster C_j are moved downward by $N - \max(IC_j)$ positions.

We can see that this movement strategy can be easily changed by altering the three starting assumptions. For instance, one can decide that linear movement is no longer a valid assumption, e.g. by saying that when d_j in moved by λ positions, the documents in C_j are moved by $\lambda - \sigma$,

where σ is a variable calculated on the basis of the documents rank or score. One can decide that cluster independence is no longer valid, e.g. by saying that the movement of the documents in C_j influences accordingly all the clusters of the documents that have to be shifted for making room to C_j. Finally, one can provide an alternative shifting strategy in place of the unary shifting. All these aspects and their combination make room for further investigation and new research.

3.8 Domino Effect Estimation

The domino effect leverages on the above illustrated steps, in particular it can be performed once the clusters have been defined and at least a movement has been performed. This step provides the capability of simulating how one or more changes in a ranking list for a topic, influence all the other ranking lists produced by the same system as described in Section 2.6.

The domino effect is estimated by employing the learning to rank framework described in Section 3.6. For estimating the domino effect we start from a manually modified ranking list for topic $t_j \in T$ and we build the features vector $\mathbf{x}^i$. We then take the training set used for learning the ranking model h used for build the clusters and we substitute the old features vector with the modified one. In this way we have a new training set that we use to learn a new ranking model $h^{'}$.

Afterwards, we pass to $h^{'}$ the training set, minus the modified features vector $\mathbf{x}^i$, as test data obtaining in this way $|T|-1$ new document lists ordered as the modified ranking model would do.

4 Visual Analytics Environment

In this section will be described the characteristics of the implemented prototype, in terms of both technological and design choices. VATE^2 prototype is implemented in the form of a web application and is accessed by a homepage (that serves as entry point for the application) represented in figure 4. This homepage replicate syntactically the main coordinates of analysis described in section 2, nominally *Performance*, *Failure* and *What-if* analyses, each of them instantiated on a particular granularity level of the domain (*Topic level* or *Experiment level*); this categorization results in 6 possible types of analysis presented to the user, each of them represented by a big tab at the cross point between analysis coordinates. In every tab is present a sketch of the visualization environment that is referred to a sketch of the visualization environment that is referred to, in order to give to the user an hint on the type of visualization that he will interact with; in this way, the schema of presentation helps the user to keep in mind which kind of task is about to start. At each type of analysis is moreover assigned an alphabetical order, from A to F: also if not mandatory, following the sequential order assigned to the different analyses is suggested in order to better understand all the formalism and visualization choices used in this prototype. In the following we will describe each of these tabs in the aforementioned order: it will help us highlight the most important changes and improvements that each tab introduces w.r.t. the previous ones.

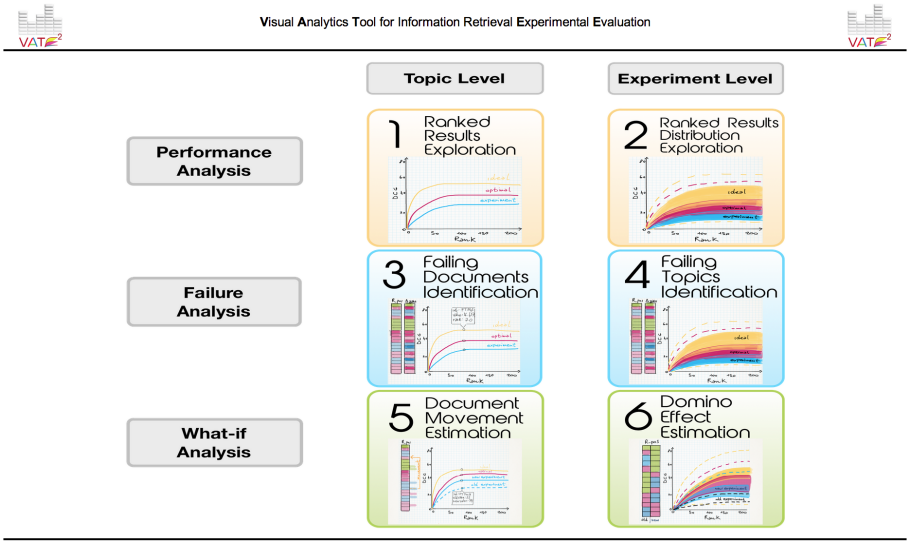


Figure 4: VATE^2 home page

4.1 Ranked Results Exploration

Type of analysis: Performance Analysis
Granularity level: Topic level
The first type of analysis that can be started is the "Performance analysis at single topic level", that envisions the possibility to understand how the system under examination is behaving with respect

to the selected topic.
The general presentation of this tab is reported in figure 5. On top of the page is present an useful breadcrumb icon with the actual tab of analysis highlighted; this allow the user to easily identify at which stadium of the analysis he is staring at. Moreover, it allows to easily return to the homepage and change the type of analysis. The main visualization area is split into 2 sub-areas:

- **commands and options area**: in this area, visible in figure 5 on the left part of the page, are present all the commands and options that the user can specify in order to suit the visualization to the analysis needs: the first drop-down menu, labelled **"Experiment Selection"**, allows the selection of the IR campaign of interest and one of the possible tracks that compose the campaign.
  Immediately under, the self-explanatory **"Legend"** area serves as a trace for understanding the curves represented in the graph area (discussed later in this section). Last part of the area allows the user to change the topic under examination from a grid; these topics are extracted from the track previously selected. The active topic will be drawn in green, while the others will maintain the default grey color. It is possible from this area to change not only the topic under examination, but also the family of metrics used to asses the quality of the experiment: 4 different families of metrics has been implemented, nominally Discounted Cumulated Gain (DCG), Cumulated Gain (CG), and their normalized forms, abbreviated in nDCG and nCG. For what concerns the discounted versions of these metrics (DCG and nDCG), by interacting with a text element it will be possible to specify the base of the logarithmic function used to discount the values of the metrics for each of the documents taken into account: this command will result automatically disabled for the not-discounted versions.

- **graph area**: this area is deputy to the visualization of the graph that represents the assessment of the quality of the experiment with respect to the chosen topic and evaluation metric. It is constituted by a line graph: on the x-axis the first 200 ranking positions of the documents constituing the experiment are reported, while on the y-axis the score w.r.t. the chosen evaluation metric is reported. In this graph area three trends are represented:

  1. *Experiment ranking* trend, displayed with a cyan color, representing the discounted cumulated gain scores obtained by the actual ranking of the documents.
  2. *Optimal ranking* trend, displayed in magenta color, representing the discounted cumulated gain scores obtained by the optimally re-ranked set of documents that constitutes the *Experiment ranking*.
  3. *Ideal ranking* trend, displayed with a yellow color, representing the discounted cumulated gain scores for the best possible ranking achievable.This ranking is obtained by re-querying the system for a new set of documents,the best possible one, only partially overlapping the original queried one.

  This graph area will respond to each of the events triggered by the *commands & options* area, i.e. if a different value for the logarithmic base of the discounting function is selected, the line-graphs will be recomputed, and the scale of the graph area will be readjusted accordingly.

Some additional graphical indicators of the quality of the experiment ranking w.r.t. the ideal and optimal ones are present: two pairs of black circle represent the points of maximum distance between experiment and ideal rankings, and between optimal and ideal rankings respectively. An useful tooltip is then added to each of the points constituting the different 3 trends, detailing various information about the document selected, like the ID, the exact score of the evaluation metric computed up to that document and its ordinal rank.

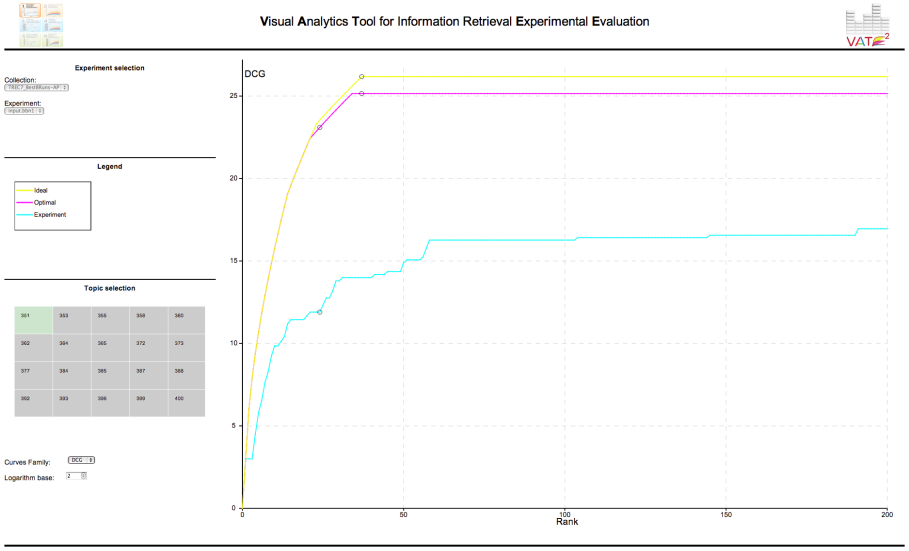


Figure 5: **Tab A: Ranked Results Exploration**

4.2 Ranked Results Distribution Exploration

Type of analysis: Performance Analysis
Granularity level: Experiment level

The second type of analysis that can be started is the "Performance analysis at whole experiment level", that allows the user to understand how the system under examination is behaving with respect to the entire experiment selected.
In regards of the two main areas introduced in the previous paragraph, it is still possible to select collection and experiment of choice: the meaning of the topics grid this time is not to select a single topic each time, but to have as a selection a subsets of topics (that can or cannot coincides with the whole set contained in the experiment); a simple color coding will fill the cell corresponding to a selected topic in green, and the cell of an unselected one in grey; this will help the user to navigates the visualization and adjust it according to analysis needs. An useful "select All" button, positioned under the topics grid, will ease the re-selection of the whole set of topics, to facilitate fast transition among very different types of analysis (i..e., passing from a situation with low number of selected topics to one with an high number of them)

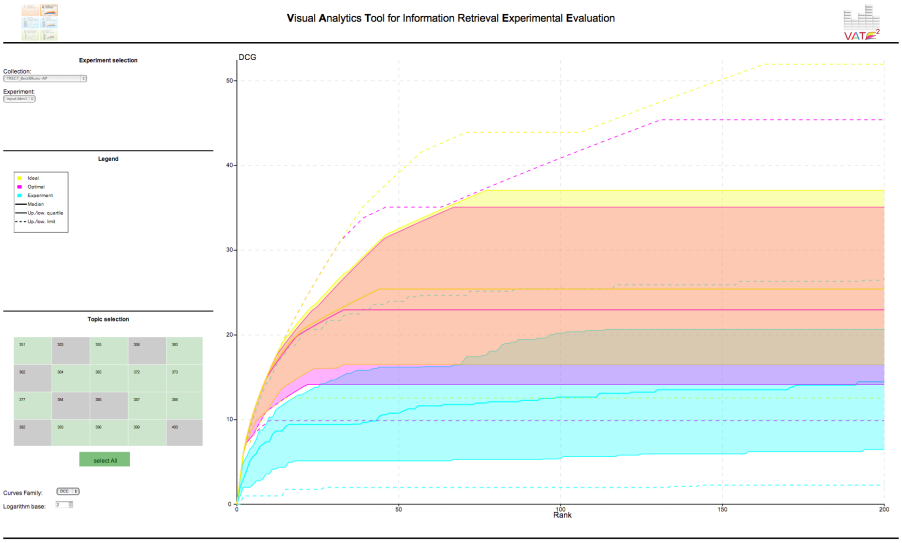


Figure 6: **Tab B: Ranked Results Distribution Exploration**

This behavior is instrumental to the main difference introduced in the line-graph area, illustrated in figure 6. In this analysis, what is represented in the x-axis and y-axis are still ranking position and selected evaluation metric values, for every ranking position a boxplot on the different values of the metric that the experiment earns for every topic is computed; for every boxplot its 5 main constituting points, nominally upper/lower limits, upper/lower quartiles and median are displayed.
This process leads to the plot of 200 values for each of these peculiar points, that given the particular natures of the metrics used in this work, leads to the representation of 5 main trends, each constituted by the unions of the respectively homogeneous type of points: so what we have in the end will be the following 5 trends with the respective visualization patterns:

1. Upper limit trend (dash-stroke line)
2. Lower limit trend (dash-stroke line)
3. Upper quartile (continuous-stroke line)
4. Lower quartile (continuous-stroke line)
5. Median trend (thick continuous-stroke line)

Due to the statistical meaning of a boxplot, it has been chosen to fill the area included between the upper and lower quartile with an alpha-blended color, in order to highlight where concentrates the statistical majority of the experiment results.

What has been described above is repeated for each of the cases studied, nominally Experiment results, Optimal results and Ideal results, and leads to the final state of the visualization shown in figure 6

On the Interaction side, it is possible with a mouseover to highlight the main trends of a single case, as shown in figure 7 for the case of Ideal results: this allows the user to better navigates the visualization in order to further inspect it.

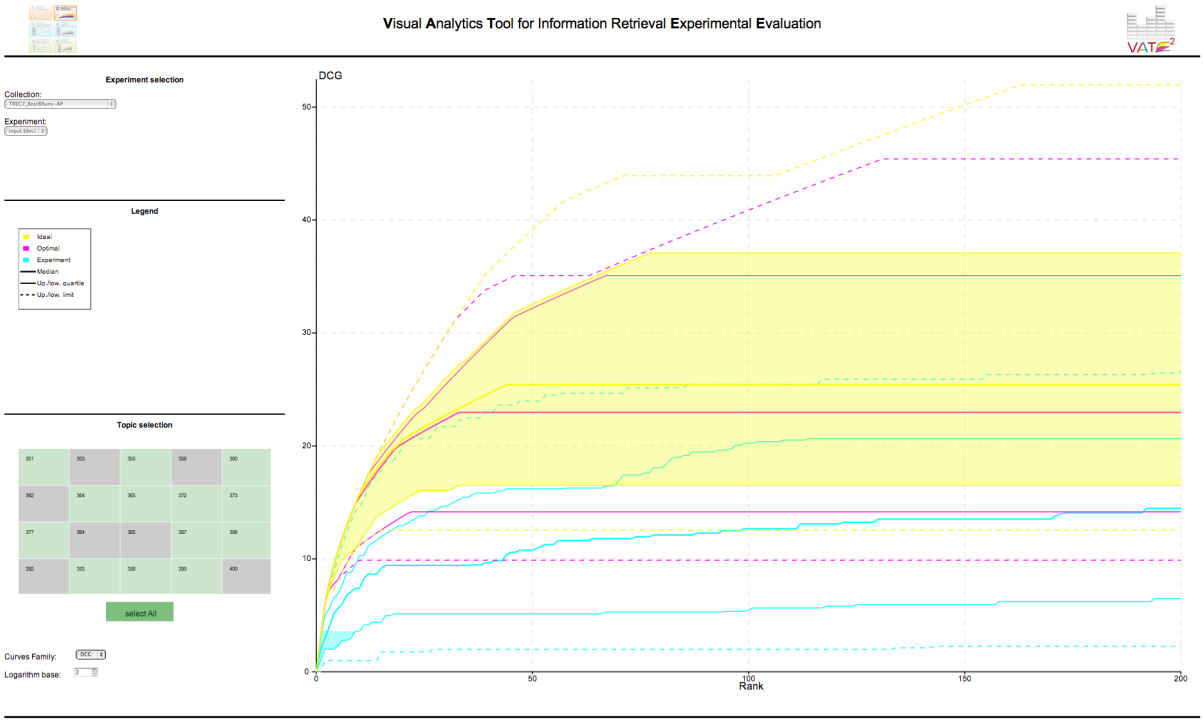


Figure 7: with a mousover operation, the Ideal results trends are isolated from the rest of the visualization

Moreover, by clicking on the squares in the legend area (on the left of the visualization) it is possible to show/hide the real curves of the whole set of results, in contrast with the statistical aggregation curves; the result is shown in figure 8 for the case of Optimal results.

4.3 Failing Documents Identification

Type of analysis: Failure Analysis
Granularity level: Topic level

The third type of analysis that can be executed is the "Failure analysis at single topic level". This is the first tab that allows the user to execute a Failure analysis on the experiment result for the selected topic: with Failure analysis we intend not only the evaluation of the score earned by the experiment result, but a further inspection on the behavior of each document that constitutes the ranking w.r.t. the position it has in the ranking. The resulting visualization is presented in figure 9: the behavior of the commands & options area is the same as the one described in section 4.1, while the main innovation is the presence of a second visualization that serve as a mechanism to execute the Failure analysis.

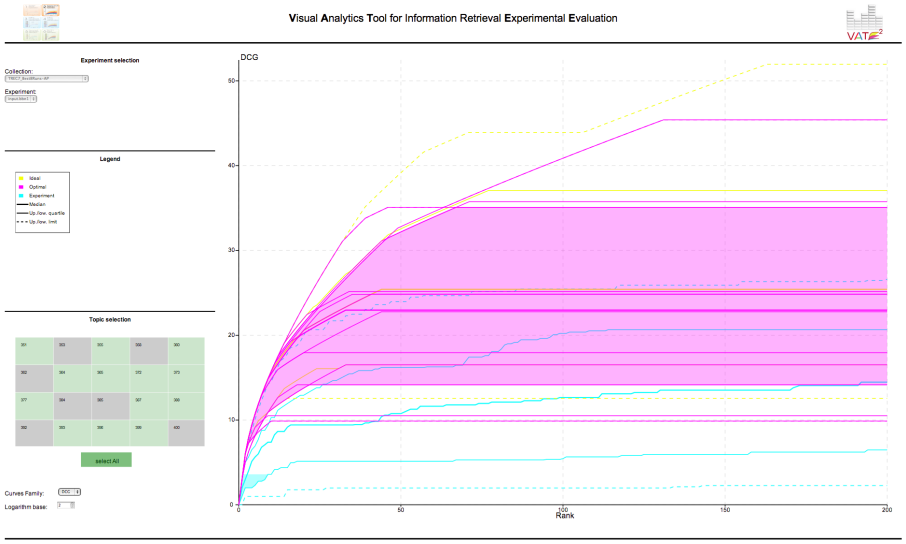


Figure 8: Clicking on the Optimal result legend visualize the whole set of relatives trends

This second visualization is constituted of two bar charts, one that serves at displaying the value of the "Relative position" and the other representing the "$\Delta Gain$" contribution; more in details:

- **Relative position bar**: this visualization represents the relation that exists among the position in the actual ranking of each document and the ideal position it should have in the best possible ranking; details about its computation can be found in section 3.5. The following color coding has been chosen in order to translate visually the contribution of each documents:

  1. document well placed: green color
  2. document placed below its ideal position: blue color
  3. document placed above its ideal position: red color

  Also if both case 1) and 3) represents error situations, in the opinion of the authors the case in which a document less relevant to the topic considered is positioned above its ideal position represents a far more grave situation than the opposite case: the reason lie in the fact that it actually has influence on the discounting function that is computed on the intervals of documents that span from its position to the recall base.

- $\Delta Gain$ **bar**: this visualization represents the relation that exists among the documents constituting the ranking, their position and the contribution that they provide to the evaluation metric score in term of loss or gain: in this way the user can quickly understand by how much the wrong positioned document is affecting the overall score of the experiment and make an idea on the set of documents on which the experiment is not behaving correctly. It is still used a color code for discerning among these 3 cases, where with green is represented the optimal contribution, with blue a gain in contribution w.r.t. the optimal one and with red a loss in contribution. Moreover, just like for the Relative Position bar, the hue of the color will be proportional

to the amount of loss/gain in overall score that particular document provides; clearly this will be not true for the case in which the document is well positioned (only 1 tone of green). The complete visualization is shown in figure 9: the document in second and third position, also if just slightly misplaced (Relative Position bar) affects the computation of the DCG value, resulting a huge loss in score (really strong hue of red in the $\Delta Gain$ bar). This situation is also visible in the line graph, where after a really low number of ranking position the experiment trend starts diverging from the optimal and ideal trends, resulting in a much lower overall score.

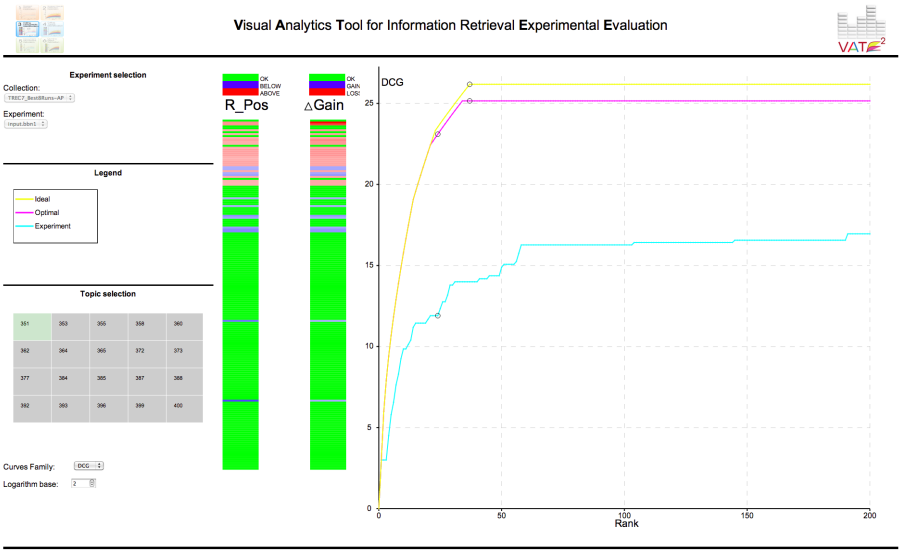


Figure 9: **Tab C: Failing Documents Identification**

4.4 Failing Topics Identification

Type of analysis: Failure Analysis
Granularity level: Experiment level

In the Failing Topics Identification visualization, allocated in the Failure Analysis at whole experiment level, the main focus is to find which are the topics where the experiment under examination is behaving incorrectly. To accomplish this task, the visualization is organized as shown in figure 10: in detail, a commands & options area, a bars visualization area and a line graphs area.

The novelty of this visualization w.r.t. the previously mentioned one is in the way the values of the Relative Position and $\Delta Gain$ bars are computed: this time, instead of presenting for each position in the ranking the actual value that that particular document obtain, is presented the result of an aggregation function on ALL the values of the different documents that, w.r.t. all the different topics against the experiment is tested, obtain that position in the ranking. In this way, a second "global" visualization representing the overall behavior of the experiment is obtained, both in terms of good rankings of the documents and in aggregated contribution to the score of the selected evaluation

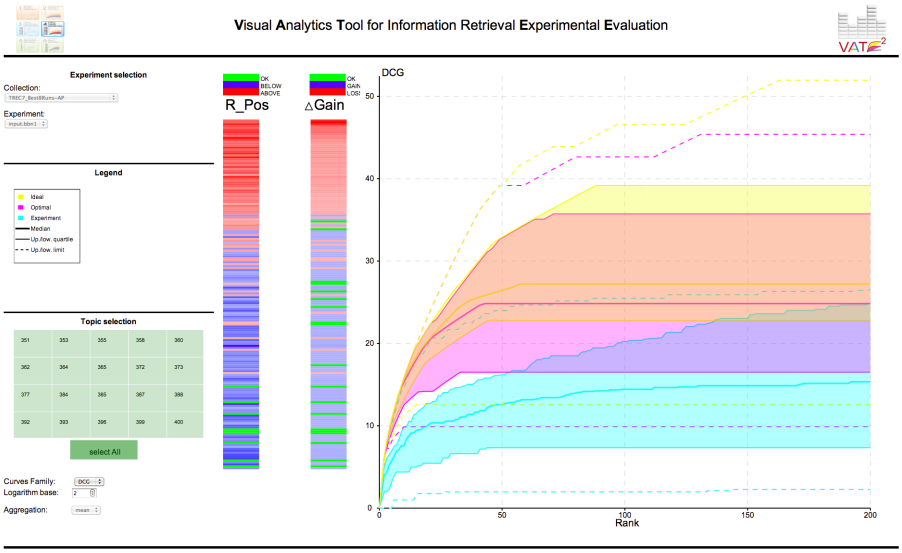


Figure 10: **Tab D: Failing Topics Identification**

metric. From the commands & options area it is possible to select/deselect various of the available topics, in order to understand which are the ones that present the most grave problems. This is clearly visible in figure 11, where consequentially to the selection of a subset of topics, represented in green, the overall status of Relative Position and $\Delta Gain$ presents a higher number of well-placed documents with respect to the initial state (encompassing all the topics) in which that number was much lower. Moreover, it is possible to change the aggregation function from the drop-down menu labelled "Aggregation" in order to conduct different statistical analyses. By default it is set to "mean".

4.5 Document Movement Estimation

Type of analysis: What-if Analysis
Granularity level: Topic level

This visualization introduces for the first time the concept of "What-if" Analysis at single Topic level. The general layout precisely follows what has been described in section 4.3 regarding the Failure Analysis at Topic level; what really is new is the capability to modify the actual ranking, keeping intact the constraints imposed by the system model, in order to obtain a better overall score of the chosen evaluation metric.

This effort is achieved by applying the following visual steps:

1. position the mouse pointer on one of the rectangles representing document positions in the Relative position bar: this action will trigger the highlighting of:
   - the documents affine to the one selected in terms of the clustering exposed in section 3.6. they will be represented as rectangles that slightly come over on the right with respect to the normal alignment.

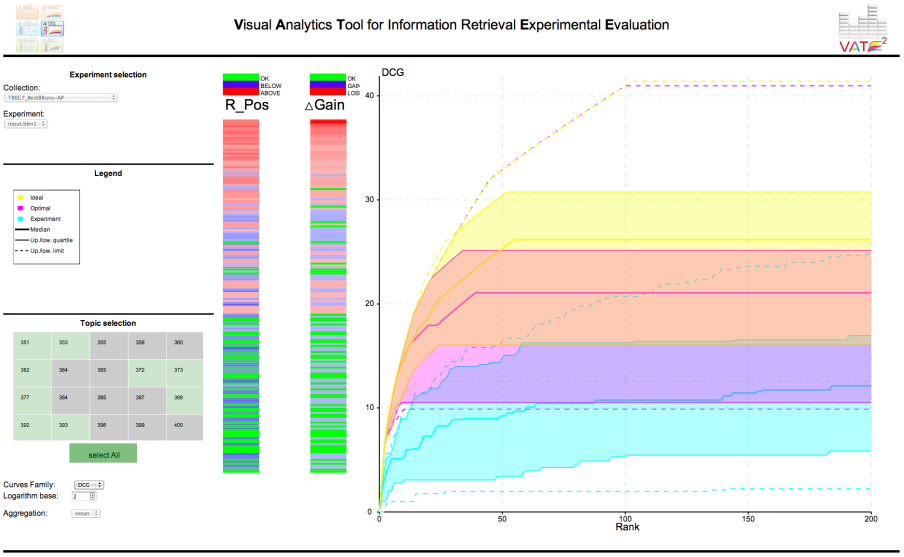


Figure 11: Improvement of aggregate values is obtained retaining a subset of the original topics

- the interval of positions in which the selected document should be positioned, coherently to what is computed in the Ideal Ranking. Visually this property is represented as a yellow vertical segment that encompass initial and final position of the aforementioned interval. Figure 12 shows this behavior.

2. after the selection of the specific document, the operation of dragging it to the new desired position in the ranking will be represented with the movement of the corresponding documents, freeing each time the selected spot where the document can be placed. Generally speaking, it is still possible to place the document in all the positions, but hopefully the yellow visual indicator described above more than suggest the interval in which the document must be placed in order to obtain a gain in the overall score of the chosen evaluation metric.

3. after the release of the selected document, it will be first repositioned in its original position, and then it and its affine cluster will all be moved to the new position selected, according to the particular law of movement chosen and the positioning constraints. The former now is implemented in the form of a constant law, where all the documents belonging to the cluster are displaced by the same amount of positions w.r.t. the displacement of the original document. This behavior is shown in Figure 13.
   The latter instead concerns the possible constraints that arises by particular movements, like for example the choice to move a document with an affine higher than it on the ranking to the top position. Due to the existence of this higher affine document, will be impossible to move the chosen document in the top spot, and the entire repositioning operation will be capped by the maximum possible raise of that affine document: this will happen because the algorithm must preserve the order of documents belonging to the same cluster.

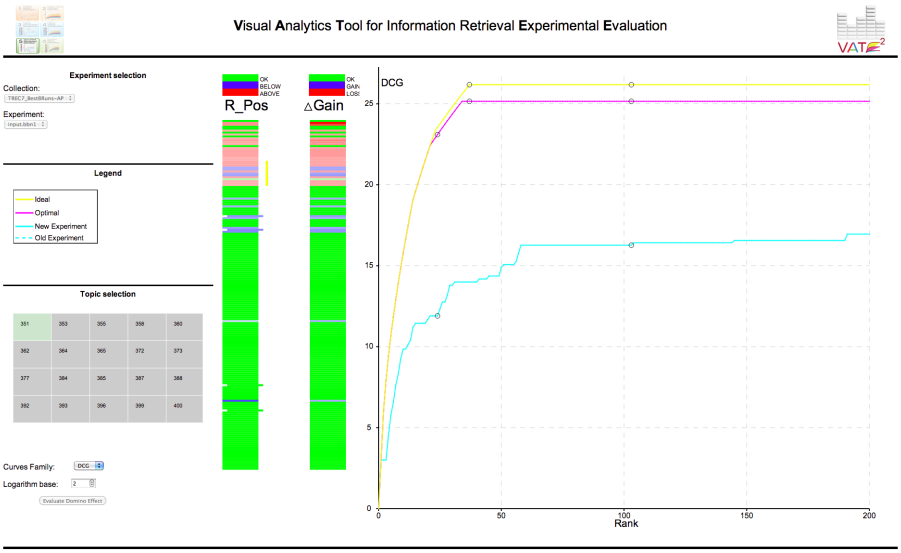


Figure 12: Tab E: Initial selection of the documents to reposition

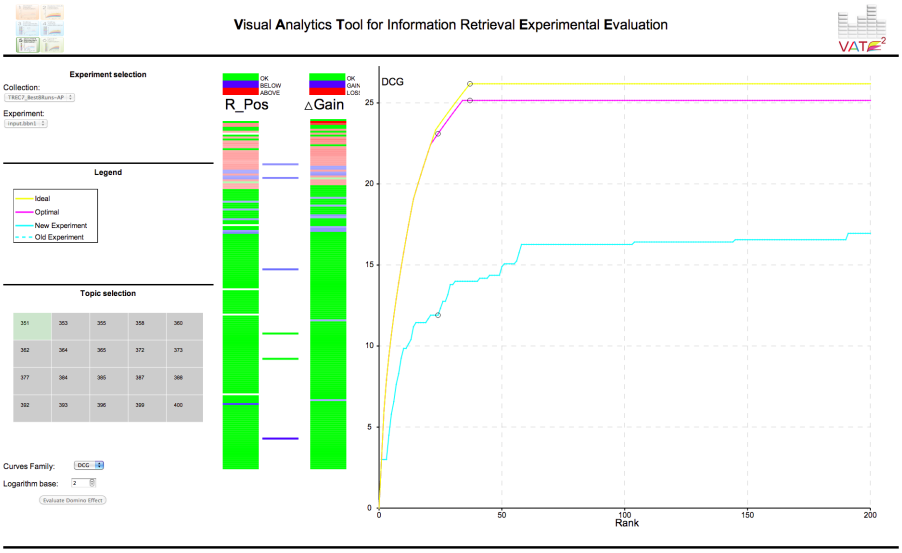


Figure 13: movement of the chosen document and it's affines cluster

4. after the repositioning phase, the new resulting ranking will be presented both in terms of new line curves and bar representations of Relative Position and delta Gain, as shown in Figure 14: the former will presents fill-stroke curves for the new computed rankings and dash-stroke curves for the initial ones. The latter instead will split the areas of the bars into two sub-areas, the left one representing the initial state of the ranking and the right one representing the new one. In both the visualization the visual preservation of the initial state is instrumental in

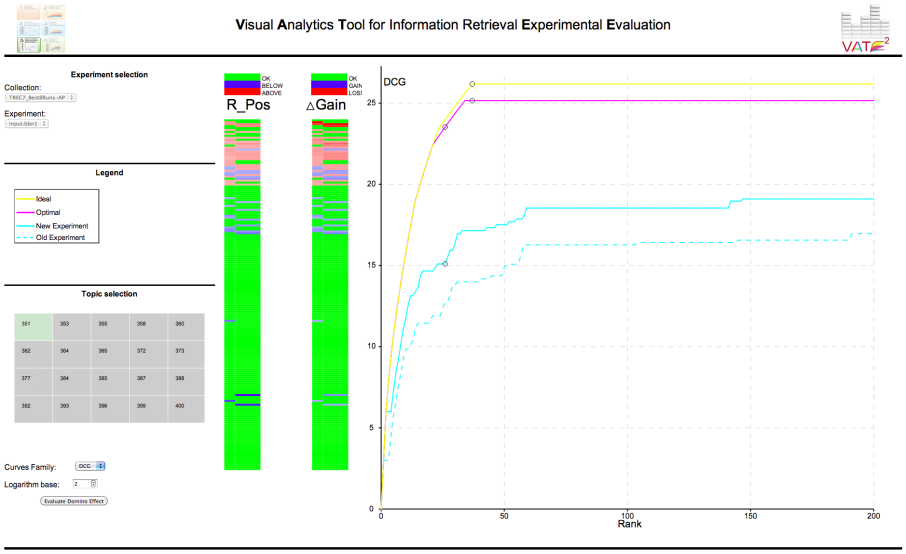


Figure 14: The resulting new ranking obtained after the repositioning phase

terms of comparisons of the new computed rankings with the one initially provided, in order to understands possible improvements and amounts of gaining from these improvements.

The whole process of What-if analysis is completely repeatable, multiple repositioning operations are possible, and stackable, in order to keep on improving the resulting ranking until the point in which the developer/evaluator is satisfied by the result. The user will just need to reposition its mouse cursor over the right part of the new obtained Relative Position bar and the whole process can be started again.

4.6 Domino Effect Estimation

Type of analysis: What-if Analysis
Granularity level: Experiment level

This final visualization introduces the concept of What-if analysis effects at whole Experiment level, nominally Domino Effect. It is not started by the corresponding tab in the index of the prototype, but instead it can be triggered by the previous tab (labelled E), after a series of what-if analysis steps, by the pressure of the corresponding "Evaluate Domino Effect" button. The main focus of this visualization is to present to the user the effects that the obtained ranking at Topic level has on the whole Experiment (constituted by a set of rankings related to the different topics constituting the experiment).

Figure 15 shows as the visualization is presented to the user: the visual aspect is quite similar to the one presented in analysis D, but this time a new set of curves, filled in green color, is presented to the user representing the new trends computed for all the topics of the experiment. In this way it is possible to evaluate the impact of the modification of the ranking at Topic level w.r.t. the whole

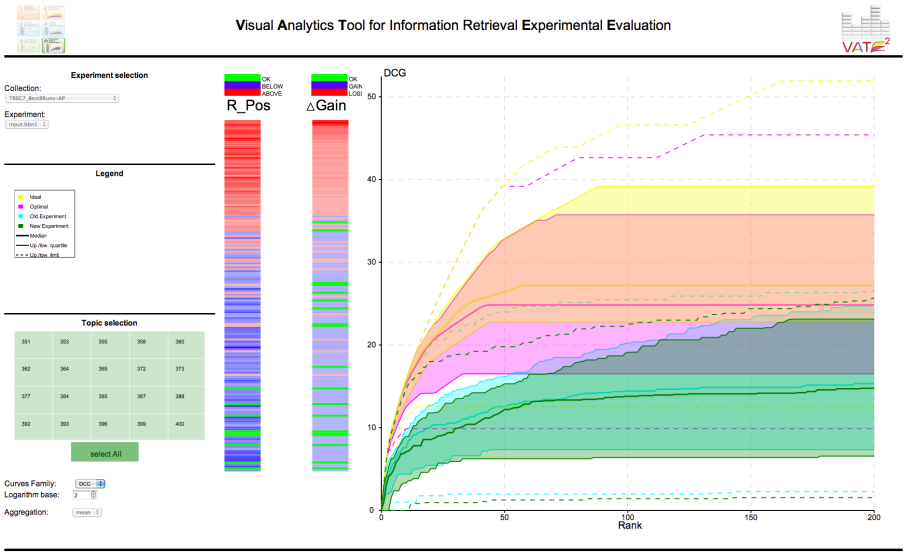


Figure 15: Tab F: Domino effect at Experiment level

experiment, in order to validate it and/or understand the grade of improvement/loss that it generates at experiment level. As in analysis D, it will be possible to further specialize the visualization on just a subset of topics and not necessarily on the whole set: this will help also in isolating the topics on which the effects are more evident.

5 Validation

We conducted a formal user study to evaluate VATE^2, involving information retrieval evaluation experts (i.e., academics, post-docs, and PhD students). It is worth noting that such experts are exactly the users the system is intended for: the tool's goal is to assist developers and researchers in understanding and fixing ranking errors produced by a search engine and this activity is not a typical end user task. In particular, 13 experts (7 female and 6 male) have been involved in the study, coming from 9 European Countries and working on different aspects of IR experiment evaluation. The goal of the study was to assess a) the VATE^2 scientific relevance and innovation and b) the comprehensibility and efficacy of the proposed visualizations.

5.1 Methodology

Before starting the study, people have been instructed through an oral presentation about VATE^2 background and a practical use of the system has been demonstrated, in order to allow participants to know the system and to let them understand how to use it. The performance analysis part as well as the failure analysis one are more straightforward and close to the day by day experience of the experts; whereas, the what if analysis evaluation represents a totally new paradigm which requires some time to be properly understood. Each visualization has been discussed in detail together with the associated automated analysis. Questions about the overall methodology, technical details, and visualizations have been answered.

After that, participants have been given a closed questionnaire, with an interval Likert scale ranging from 1 to 5, in which each numerical score was labeled with a description: {1:not at all, 2:a little, 3:enough, 4:a lot, 5:quite a lot}. The questionnaire was structured in 7 identical sections, one for each visualization described in Section 4 (see Figure 4) plus one for the overall system. An additional open section (optional) has been provided for collecting additional comments. Each of the 7 closed sections was compound of two groups of questions:

Q1 Is the addressed problem relevant for involved stakeholders (researchers and developers)?

Q2 Are the currently available tools and techniques adequate for dealing with the addressed problem?

Q3 Do currently available tools and techniques for dealing with the addressed problem offer interactive visualizations?

Q4 Is the proposed visual tool understandable?

Q5 Is the proposed visual tool suitable and effective for dealing with the addressed problem?

Q6 To what extent the proposed visual tool is innovative with respect to the currently available tools and techniques?

Q7 To what extent the proposed visual tool will enhance the productivity of involved stakeholders (researchers and developers)?

The first three questions, visually split from the other four, were aimed at collecting the experts' opinion about the relevance of the addressed problem (Q1), the adequateness (Q2) and the degree of interactiveness (Q3) of other visual tools designed for the same purpose. The last four questions were aimed at assessing the VATE^2 understandability (Q4), suitability (Q5), visual innovativeness (Q6), and efficiency (Q7).

The study was conducted by allowing the experts to freely use VATE^2 for an hour, following the path "Performance Analysis, Failure Analysis, and What-if Analysis" (see Section 4) and compiling the questionnaire sections that were arranged in the same order. The questionnaire is reported in Section 7.

5.2 Results

The questionnaire results are depicted in Figure 16 that presents the distribution of the answers assessing the system as a whole, and in Figure 17, that provides details, through averages, on each of the 6 VATE^2 components.

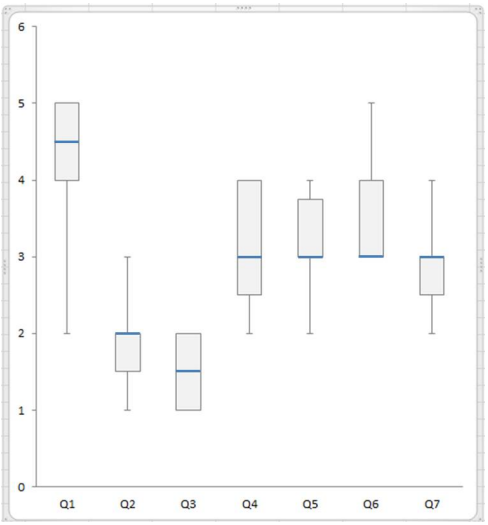


Figure 16: Evaluating VATE^2 as a whole.

Considering Figure 16, we can conclude that the addressed problem has been judged as a relevant one from the involved stakeholders (90% of the answers to Q1 are in the range [4, 5] with mean=4.3 and STD=0.95) and that there not exist any other tool doing the work of VATE^2 (Q2 and Q3, in which more than 85% of the answers are in the range [1, 2] with mean=1.9 and STD=0.69 for Q2, and mean=1.50 and STD=0.55 for Q3). That means that, according to the experts' opinion, VATE^2 is proposing something totally new in the field. We can also conclude that the tool

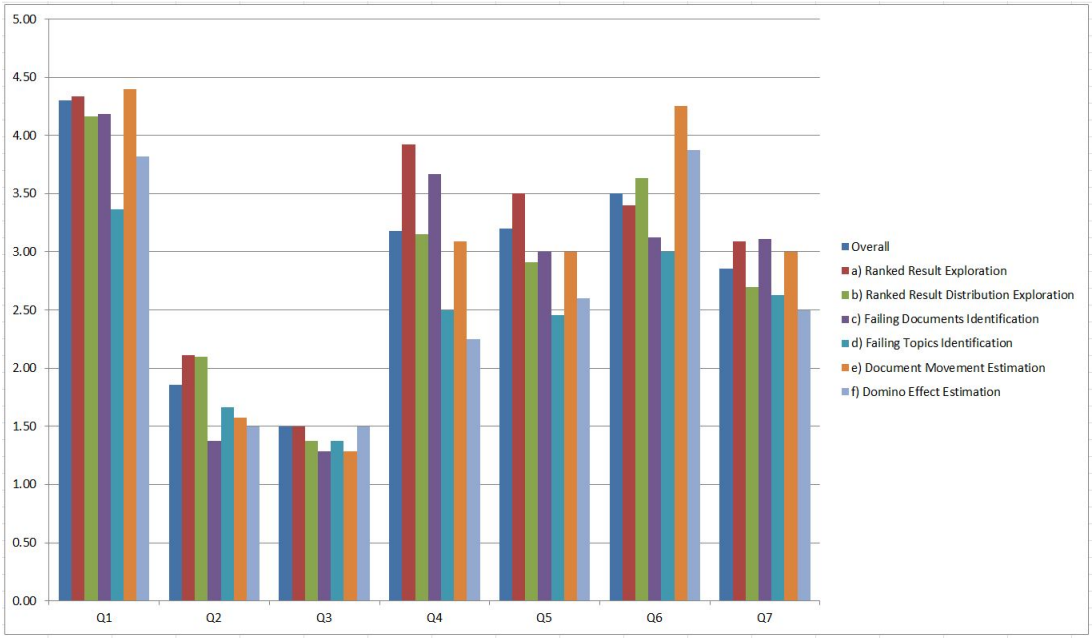


Figure 17: The histogram reporting the averages of the experts' answers to all the sections of the questionnaire.

is understandable (72% of the answers to Q4 are in the range [3, 4], mean=3.18 and STD=0.87), suitable (90% of the answers to Q5 are in the range [3, 4], mean=3.20 and STD=0.63), innovative (all the answers to Q6 are in the range [3, 4], mean=3.50 and STD=0.76). The last question is about productivity; on average the experts think VATE^2 can improve productivity (71% of the answers to Q7 are in the range [3, 4], mean=2.86 and STD=0.89) but the mean is below 3 and we think that this is due to the time needed to learn how to effectively use the system for the what if analysis and by the inherent complexity of the Failure Analysis and What-if Analysis visualizations at experiment level (Figure 1 (d) and (f)).

Such a complexity is confirmed from the detailed results depicted on Figure 17 in which these two visualizations show low values for Q4 (mean=2.50 and STD=0.67) and Q5 (mean=2.25 and STD=0.75). The other values are closer to the overall means, showing a slightly worse behavior for the what-if analysis, which as explained above, is a brand new topic in IR evaluation and likely it requires more time to become useful to the experts.

5.3 Discussion

While the study results give clear indications on the usefulness and the innovation of the VATE^2 system (we have got some enthusiastic comments like P1: "I would love to have this tool, both for research and for teaching purposes" and P8: " If I have had this tool during my PHD thesis writing I would have saved weeks of work"), there are some issues that deserve more attention, requiring a

more clear design and a deeper analysis. That holds for both the methodological approach underlying the what-if analysis and the chosen visualizations at experiment level. These considerations rise from some low scores on questions Q4 and Q5, from the questions the participant asked during the experiment, and from the free comments on the questionnaires. In particular, while the visualization and the analytical models underlying the failing topic identification have been understood and positively judged during the evaluation of the system, the same did not happen for the document movement and the domino effect estimation. In particular, we have got two negative comments on the animation (P5 and P7), e.g., "it is disconcerting when documents fall down again (after moving up) before all documents move" and that failure analysis and experiment level are hard to deal with (P7) "Failure analysis is too hard to use...,experiment level views of performance and failure are difficult to interpret...". Moreover P12 asked for a better algorithm for the what if analysis "what if is really potentially useful but needs to be hooked up to an algorithm". From one side this is partially explained by the novelty of such an approach and by the little time that we planned for the experiment execution; we have acknowledged that we have underestimated the learning curve of the system, even for expert users. From the other side, comments, questions, and validation scores give us the feeling that the visualizations we are proposing for Failure Analysis and What-if Analysis at experiment level, see Figure 1 (d) and (f), are bearing a lot of visual information, i.e., three levels of analysis in (d) (ideal, optimal, and experiment) and six in (f), i.e., the same 3 levels for the situations before and after the changes. Interaction, highlighting, alpha blending, and brushing mitigate the problem but require time to be learned and likely some longitudinal studies can provide more insights on how to improve such visualizations. Moreover we have got several comments on basic usability issues like missing on screen instructions (P1, P3) and on additional required features, like allowing for inspecting details of topics and documents (P1, P5, P7, P9, P11) and having information about the number of relevant documents for a topic (P1); fixing such issues and addressing user suggestions will result in a clear systems improvement. Indeed we have got some useful indications on how to improve the system, pointing out different analysis strategies, e.g., P11: "keep the automatic clustering of docs as an option", and suggesting alternatives for performing more accurate topic analysis (P11: "...it would be nice to give the possibility to cluster topics by good/bad to look at the chosen group of topics only), giving us some insights on how to refine and improve the actual model. Moreover P9, P10, and P13 suggest to use the system to compare two or more experiments.

6 Conclusions

IR is a field deeply rooted in evaluation which is carried out to assess the performances of the proposed algorithms and systems and to better understand their behavior. Nowadays, systems are becoming increasingly complex since the tasks and user needs they need to address are becoming more and more challenging. As a consequence, evaluating and understanding these systems is an increasingly demanding activity in terms of the time and effort needed to carry it out. The goal of this paper is thus to provide the researcher and developer with better and more effective tools to understand the system behavior, its performances, and failures.

To this end, we have designed and developed an innovative tool for conducting performance and failure analysis of IR systems. The proposed tool exploits visual analytics techniques in order to foster interaction with and exploration of the experimental data at both topic and experiment level. It improves the state-of-the-art in the evaluation practice by: (i) easing the interaction and interpretation of DCG curves, a very widely adopted way of measuring ranked result lists; (ii) clearly highlighting critical areas of a ranked result list in order to quickly inspect and detect causes of failure; (iii) providing a convenient way to partner the detailed analysis at the topic level with an overall analysis at the global experiment level which support users in spotting critical topics and/or critical rank areas across several topics.

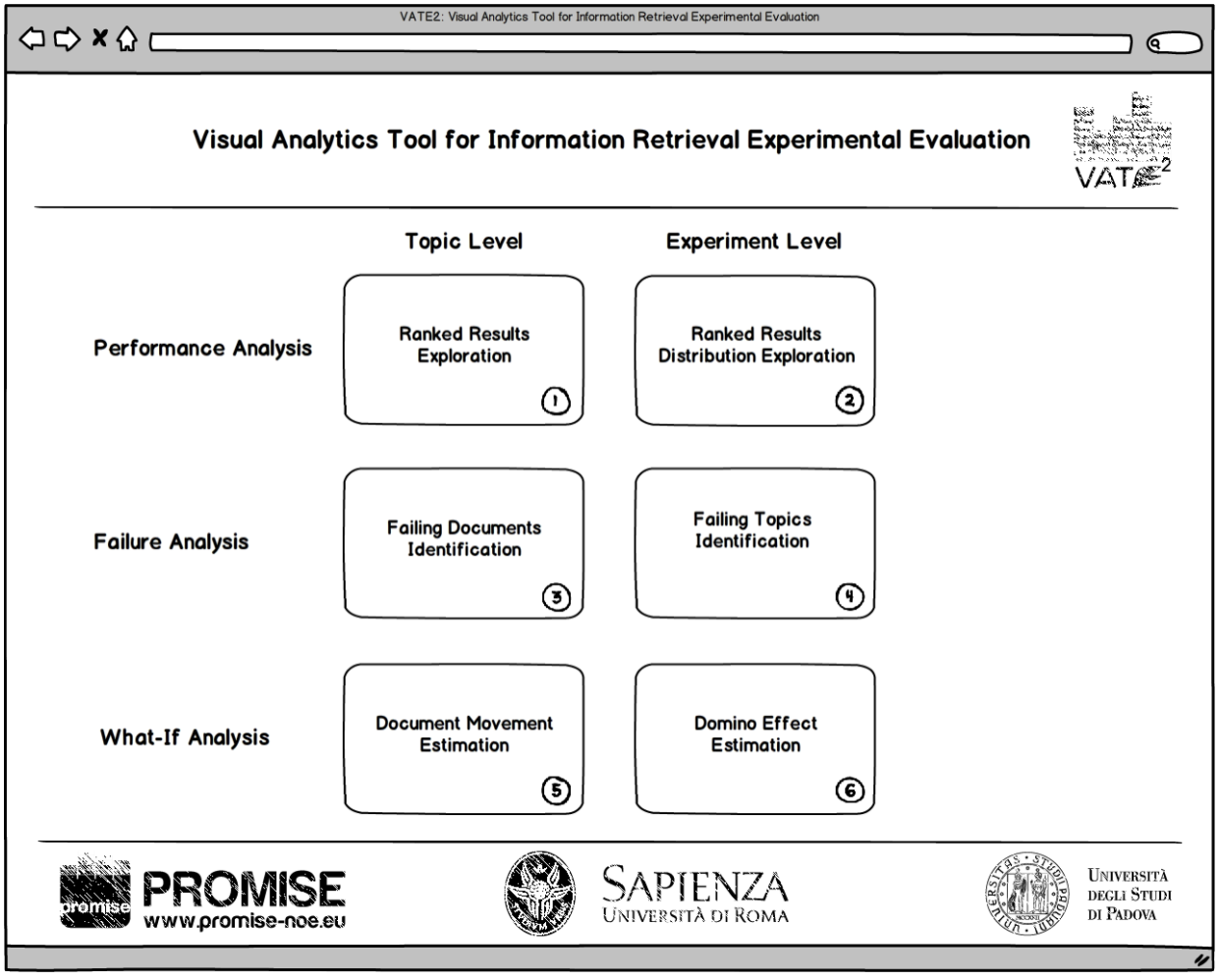
We conducted an evaluation of the proposed tools with IR experts and the outcomes have been encouraging in terms of the usefulness, innovativeness and potential of the proposed approaches.

7 Appendix

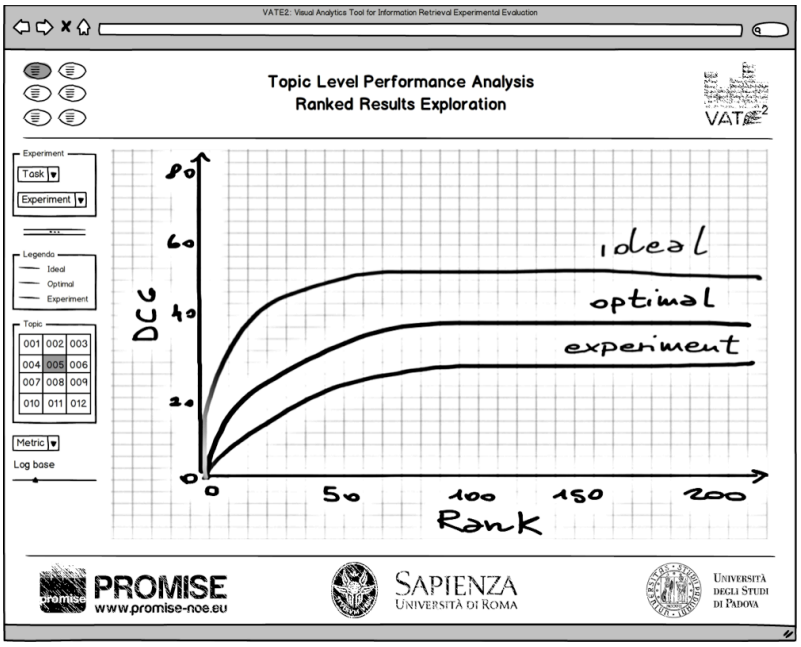
In this section we report the questionnaire adopted for the validation with expert users.

SURVEY ON

Visual Analitycs Tool for Information Retrieval Experimental Evaluation

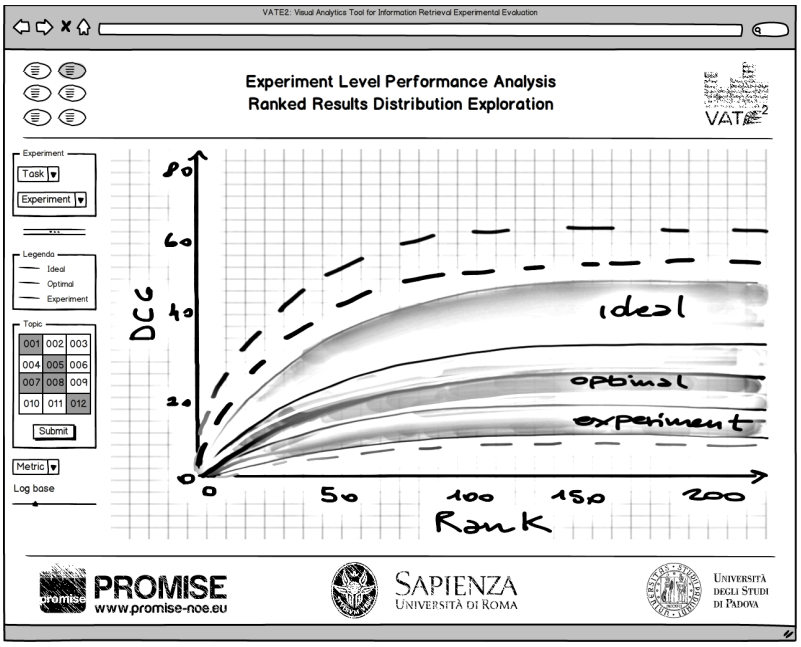


ADDRESSED PROBLEM: TOPIC LEVEL PERFORMANCE ANALYSIS
VISUAL TOOL: RANKED RESULTS EXPLORATION



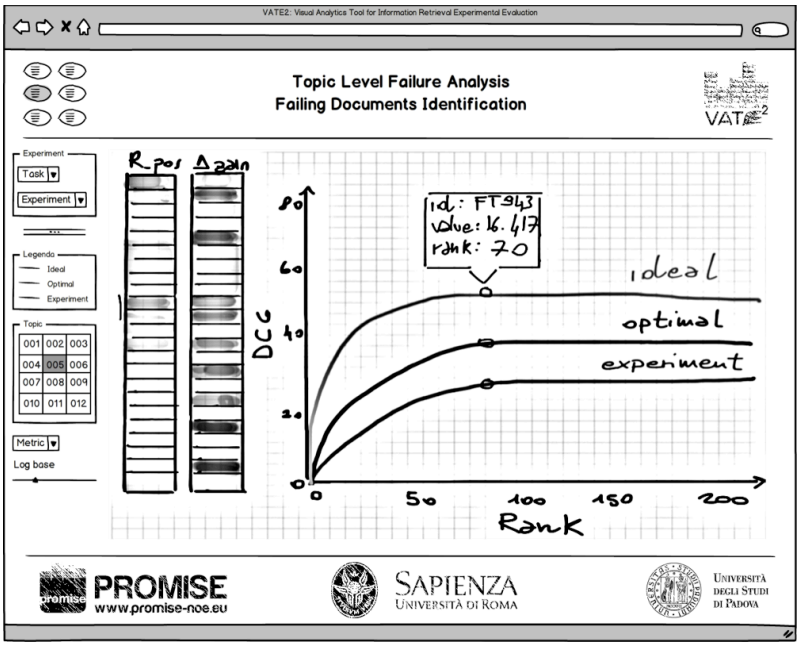
		Not at all	A little	Enough	A lot	Quite a lot
ADDRESSED PROBLEM QUESTIONS	1. Is the addressed problem relevant for involved stakeholders (researchers and developers)?	1	2	3	4	5
	2. Are the currently available tools and techniques adequate for dealing with the addressed problem?	1	2	3	4	5
	3. Do currently available tools and techniques for dealing with the addressed problem offer interactive visualizations?	1	2	3	4	5
VISUAL TOOL QUESTIONS	4. Is the proposed visual tool understandable?	1	2	3	4	5
	5. Is the proposed visual tool suitable and effective for dealing with the addressed problem?	1	2	3	4	5
	6. To what extent the proposed visual tool is innovative with respect to the currently available tools and techniques?	1	2	3	4	5
	7. To what extent the proposed visual tool will enhance the productivity of involved stakeholders (researchers and developers)?	1	2	3	4	5

ADDRESSED PROBLEM: EXPERIMENT LEVEL PERFORMANCE ANALYSIS
VISUAL TOOL: RANKED RESULTS DISTRIBUTION EXPLORATION



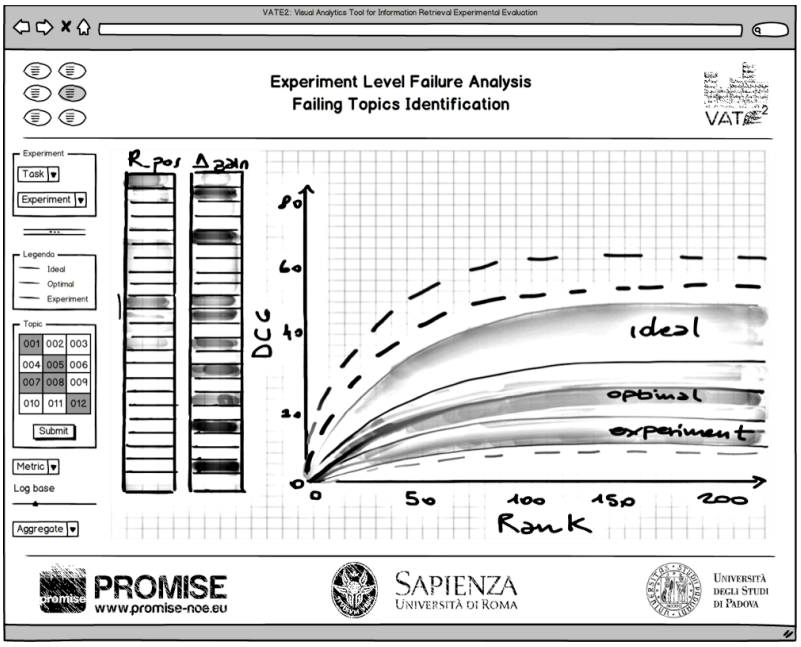
		Not at all	A little	Enough	A lot	Quite a lot
ADDRESSED PROBLEM QUESTIONS	8. Is the addressed problem relevant for involved stakeholders (researchers and developers)?	1	2	3	4	5
	9. Are the currently available tools and techniques adequate for dealing with the addressed problem?	1	2	3	4	5
	10. Do currently available tools and techniques for dealing with the addressed problem offer interactive visualizations?	1	2	3	4	5
VISUAL TOOL QUESTIONS	11. Is the proposed visual tool understandable?	1	2	3	4	5
	12. Is the proposed visual tool suitable and effective for dealing with the addressed problem?	1	2	3	4	5
	13. To what extent the proposed visual tool is innovative with respect to the currently available tools and techniques?	1	2	3	4	5
	14. To what extent the proposed visual tool will enhance the productivity of involved stakeholders (researchers and developers)?	1	2	3	4	5

ADDRESSED PROBLEM: TOPIC LEVEL FAILURE ANALYSIS
VISUAL TOOL: FAILING DOCUMENTS IDENTIFICATION



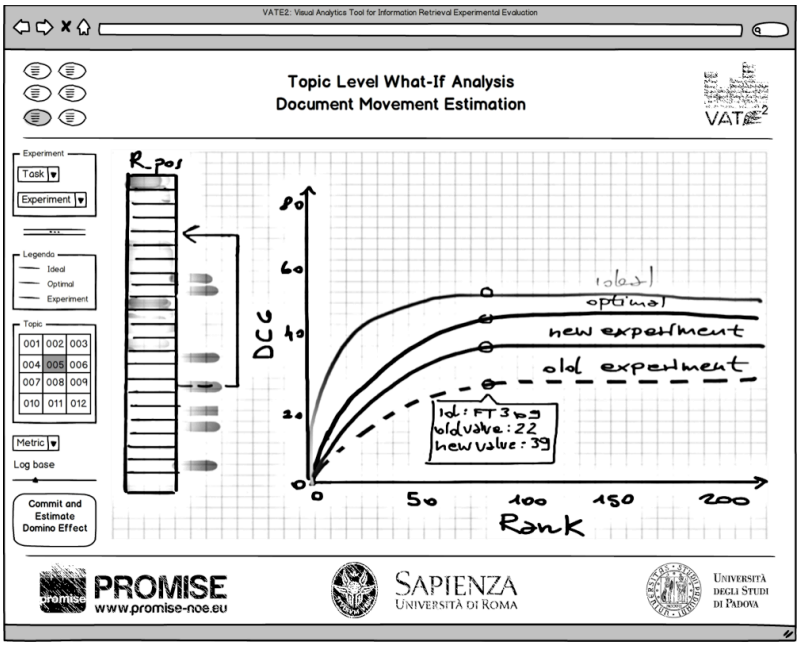
		Not at all	A little	Enough	A lot	Quite a lot
ADDRESSED PROBLEM QUESTIONS	15. Is the addressed problem relevant for involved stakeholders (researchers and developers)?	1	2	3	4	5
	16. Are the currently available tools and techniques adequate for dealing with the addressed problem?	1	2	3	4	5
	17. Do currently available tools and techniques for dealing with the addressed problem offer interactive visualizations?	1	2	3	4	5
VISUAL TOOL QUESTIONS	18. Is the proposed visual tool understandable?	1	2	3	4	5
	19. Is the proposed visual tool suitable and effective for dealing with the addressed problem?	1	2	3	4	5
	20. To what extent the proposed visual tool is innovative with respect to the currently available tools and techniques?	1	2	3	4	5
	21. To what extent the proposed visual tool will enhance the productivity of involved stakeholders (researchers and developers)?	1	2	3	4	5

ADDRESSED PROBLEM: EXPERIMENT LEVEL FAILURE ANALYSIS
VISUAL TOOL: FAILING TOPICS IDENTIFICATION



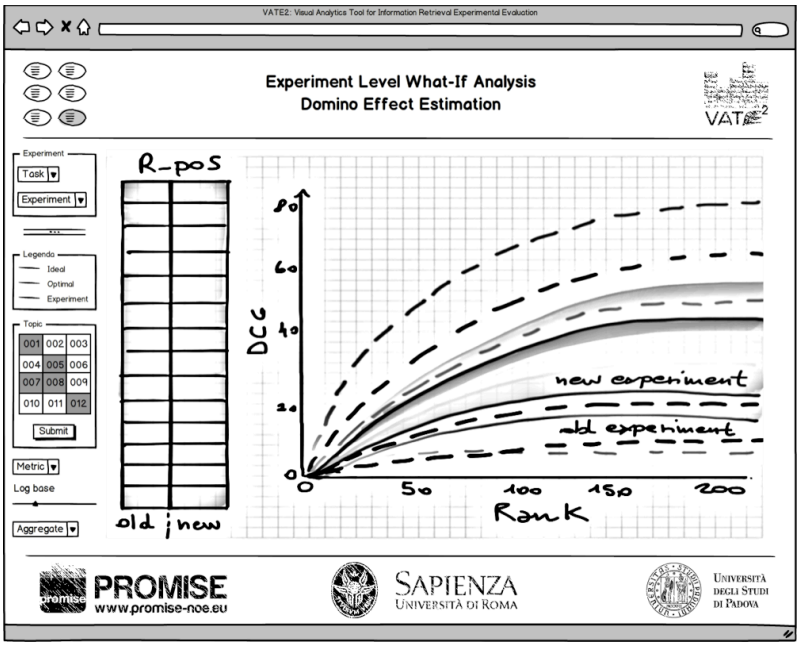
		Not at all	A little	Enough	A lot	Quite a lot
ADDRESSED PROBLEM QUESTIONS	22. Is the addressed problem relevant for involved stakeholders (researchers and developers)?	1	2	3	4	5
	23. Are the currently available tools and techniques adequate for dealing with the addressed problem?	1	2	3	4	5
	24. Do currently available tools and techniques for dealing with the addressed problem offer interactive visualizations?	1	2	3	4	5
VISUAL TOOL QUESTIONS	25. Is the proposed visual tool understandable?	1	2	3	4	5
	26. Is the proposed visual tool suitable and effective for dealing with the addressed problem?	1	2	3	4	5
	27. To what extent the proposed visual tool is innovative with respect to the currently available tools and techniques?	1	2	3	4	5
	28. To what extent the proposed visual tool will enhance the productivity of involved stakeholders (researchers and developers)?	1	2	3	4	5

ADDRESSED PROBLEM: TOPIC LEVEL WHAT-IF ANALYSIS
VISUAL TOOL: DOCUMENT MOVEMENT ESTIMATION



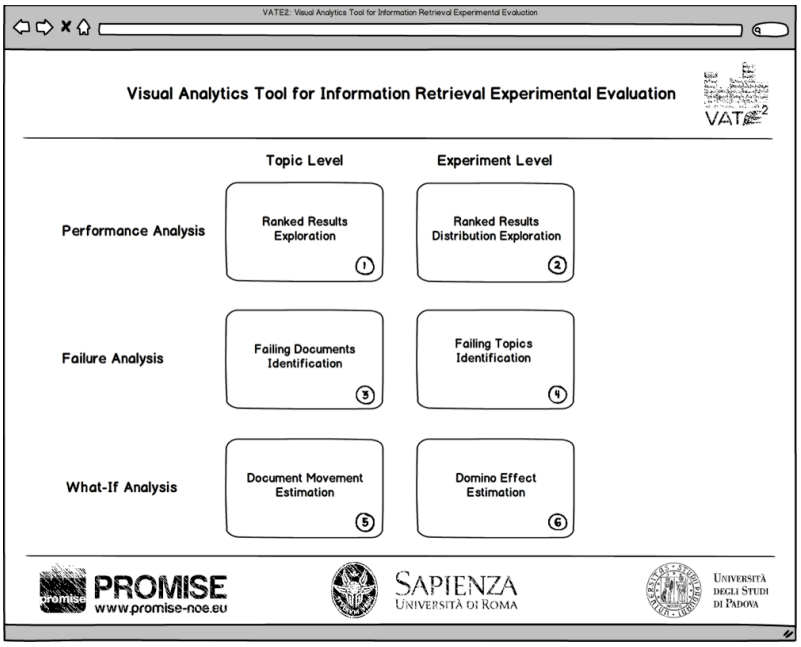
		Not at all	A little	Enough	A lot	Quite a lot
ADDRESSED PROBLEM QUESTIONS	29. Is the addressed problem relevant for involved stakeholders (researchers and developers)?	1	2	3	4	5
	30. Are the currently available tools and techniques adequate for dealing with the addressed problem?	1	2	3	4	5
	31. Do currently available tools and techniques for dealing with the addressed problem offer interactive visualizations?	1	2	3	4	5
VISUAL TOOL QUESTIONS	32. Is the proposed visual tool understandable?	1	2	3	4	5
	33. Is the proposed visual tool suitable and effective for dealing with the addressed problem?	1	2	3	4	5
	34. To what extent the proposed visual tool is innovative with respect to the currently available tools and techniques?	1	2	3	4	5
	35. To what extent the proposed visual tool will enhance the productivity of involved stakeholders (researchers and developers)?	1	2	3	4	5

ADDRESSED PROBLEM: EXPERIMENT LEVEL WHAT-IF ANALYSIS
VISUAL TOOL: DOMINO EFFECT



		Not at all	A little	Enough	A lot	Quite a lot
ADDRESSED PROBLEM QUESTIONS	36. Is the addressed problem relevant for involved stakeholders (researchers and developers)?	1	2	3	4	5
	37. Are the currently available tools and techniques adequate for dealing with the addressed problem?	1	2	3	4	5
	38. Do currently available tools and techniques for dealing with the addressed problem offer interactive visualizations?	1	2	3	4	5
VISUAL TOOL QUESTIONS	39. Is the proposed visual tool understandable?	1	2	3	4	5
	40. Is the proposed visual tool suitable and effective for dealing with the addressed problem?	1	2	3	4	5
	41. To what extent the proposed visual tool is innovative with respect to the currently available tools and techniques?	1	2	3	4	5
	42. To what extent the proposed visual tool will enhance the productivity of involved stakeholders (researchers and developers)?	1	2	3	4	5

ADDRESSED PROBLEM: UNDERSTANDING SYSYEM BEHAVIOUR
VISUAL TOOL: OVERALL VATE^2 APPROACH



		Not at all	A little	Enough	A lot	Quite a lot
ADDRESSED PROBLEM QUESTIONS	43. Is the addressed problem relevant for involved stakeholders (researchers and developers)?	1	2	3	4	5
	44. Are the currently available tools and techniques adequate for dealing with the addressed problem?	1	2	3	4	5
	45. Do currently available tools and techniques for dealing with the addressed problem offer interactive visualizations?	1	2	3	4	5
VISUAL TOOL QUESTIONS	46. Is the proposed visual tool understandable?	1	2	3	4	5
	47. Is the proposed visual tool suitable and effective for dealing with the addressed problem?	1	2	3	4	5
	48. To what extent the proposed visual tool is innovative with respect to the currently available tools and techniques?	1	2	3	4	5
	49. To what extent the proposed visual tool will enhance the productivity of involved stakeholders (researchers and developers)?	1	2	3	4	5

ADDITIONAL COMMENTS AND SUGGESTIONS

GENERAL QUESTIONS

Male ☐ Female ☐ Age: _____

Nationality: _____

Current position:

Master Student ☐

PhD Student ☐

Post-Doc ☐

Academic ☐

Industry ☐

Other: _____

Background/Competencies:

_____

_____

_____

_____

Date:

_____

References

Agosti, M., Berendsen, R., Bogers, T., Braschler, M., Buitelaar, P., Choukri, K., Di Nunzio, G. M., Ferro, N., Forner, P., Hanbury, A., Friberg Heppin, K., Hansen, P., Järvelin, A., Larsen, B., Lupu, M., Masiero, I., Müller, H., Peruzzo, S., Petras, V., Piroi, F., de Rijke, M., Santucci, G., Silvello, G., and Toms, E. (2012a). PROMISE Retreat Report – Prospects and Opportunities for Information Access Evaluation. *SIGIR Forum*, 46(2).

Agosti, M., Braschler, M., Di Buccio, E., Dussin, M., Ferro, N., Granato, G. L., Masiero, I., Pianta, E., Santucci, G., Silvello, G., and Tino, G. (2011a). Deliverable D3.2 – Specification of the evaluation infrastructure based on user requirements. PROMISE Network of Excellence, EU 7FP, Contract N. 258191. http://www.promise-noe.eu/documents/10156/fdf43394-0997-4638-9f99-38b2e9c63802.

Agosti, M., Di Buccio, E., Ferro, N., Masiero, I., Nicchio, M., Peruzzo, S., and Silvello, G. (2012b). Deliverable D3.3 – Prototype of the Evaluation Infrastructure. PROMISE Network of Excellence, EU 7FP, Contract N. 258191. http://www.promise-noe.eu/documents/10156/3783730a-bce3-481b-83df-48e209c6286a.

Agosti, M., Di Nunzio, G. M., and Ferro, N. (2011b). Deliverable D3.1 – Initial prototype of the evaluation infrastructure. PROMISE Network of Excellence, EU 7FP, Contract N. 258191. http://www.promise-noe.eu/documents/10156/e0df8a3c-388f-40e8-bfbd-04434a393004.

Agosti, M. and Pretto, L. (2005). A Theoretical Study of a Generalized Version of Kleinberg's HITS Algorithm. *Information Retrieval*, 8(2):219–243.

Angelini, M., Ferro, N., Granato, G. L., Santucci, G., and Silvello, G. (2012a). Information Retrieval Failure Analysis: Visual analytics as a Support for Interactive "What-If" Investigation. In Santucci, G. and Ward, M., editors, *Proc. IEEE Conference on Visual Analytics Science and Technology (VAST 2012)*, pages 204–206. IEEE Computer Society, Los Alamitos, CA, USA.

Angelini, M., Ferro, N., Järvelin, K., Keskustalo, H., Pirkola, A., Santucci, G., and Silvello, G. (2012b). Cumulated Relative Position: A Metric for Ranking Evaluation. In [Catarci et al., 2012], pages 112–123.

Angelini, M., Ferro, N., Santucci, G., and Silvello, G. (2012c). Visual Interactive Failure Analysis: Supporting Users in Information Retrieval Evaluation. In Kamps, J., Kraaij, W., and Fuhr, N., editors, *Proc. 4th Symposium on Information Interaction in Context (IIiX 2012)*, pages 195–203. ACM Press, New York, USA.

Balog, K., Fang, Y., de Rijke, M., Serdyukov, P., and Si, L. (2012). Expertise Retrieval. *Foundations and Trends in Information Retrieval (FnTIR)*, 6(2-3):127–256.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984). *Classification and Regression Trees*. Chapman & Hall, NY, USA.

Buckley, C. (2004). Why Current IR Engines Fail. In Sanderson, M., Järvelin, K., Allan, J., and Bruza, P., editors, *Proc. 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2004)*, pages 584–585. ACM Press, New York, USA.

Buettcher, S., Clarke, C. L. A., and Cormack, G. V. (2010). *Information Retrieval: Implementing and Evaluating Search Engines*. The MIT Press, Cambridge (MA), USA.

Burnett, S., Clarke, S., Davis, M., Edwards, R., and Kellett, A. (2006). *Enterprise Search and Retrieval. Unlocking the Organisation's Potential*. Butler Direct Limited.

Candela, L., Castelli, D., Ferro, N., Ioannidis, Y., Koutrika, G., Meghini, C., Pagano, P., Ross, S., Soergel, D., Agosti, M., Dobreva, M., Katifori, V., and Schuldt, H. (2007). *The DELOS Digital Library Reference Model. Foundations for Digital Libraries*. ISTI-CNR at Gruppo ALI, Pisa, Italy, http://www.delos.info/files/pdf/ReferenceModel/DELOS_DLReferenceModel_0.98.pdf.

Carpineto, C., Osinski, S., Romano, G., and Weiss, D. (2009). A Survey of Web Clustering Engines. *ACM Computing Surveys (CSUR)*, 41(3):17:1–17:38.

Catarci, T., Forner, P., Hiemstra, D., Peñas, A., and Santucci, G., editors (2012). *Information Access Evaluation. Multilinguality, Multimodality, and Visual Analytics. Proceedings of the Third International Conference of the CLEF Initiative (CLEF 2012)*. Lecture Notes in Computer Science (LNCS) 7488, Springer, Heidelberg, Germany.

Croft, W. B., Metzler, D., and Strohman, T. (2009). *Search Engines: Information Retrieval in Practice*. Addison-Wesley, Reading (MA), USA.

Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., and Harshman, R. A. (1990). Indexing by Latent Semantic Analysis. *JASIS*, 41(6):391–407.

Di Buccio, E., Dussin, M., Ferro, N., Masiero, I., Santucci, G., and Tino, G. (2011). Interactive Analysis and Exploration of Experimental Evaluation Results. In Wilson, M. L., Russell-Rose, T., Larsen, B., and Kalbach, J., editors, *Proc. 1st European Workshop on Human-Computer Interaction and Information Retrieval (EuroHCIR 2011)* http://ceur-ws.org/Vol-763/, pages 11–14.

Ferro, N., Sabetta, A., Santucci, G., and Tino, G. (2011). Visual Comparison of Ranked Result Cumulated Gains. In Miksch, S. and Santucci, G., editors, *Proc. 2nd International Workshop on Visual Analytics (EuroVA 2011)*, pages 21–24. Eurographics Association, Goslar, Germany.

Fox, E. A., Gonçalves, M. A., and Shen, R. (2012). *Theoretical Foundations for Digital Libraries: The 5S (Societies, Scenarios, Spaces, Structures, Streams) Approach*. Morgan & Claypool Publishers, USA.

Ganesan, P., Bawa, M., and Garcia-Molina, H. (2004). Online Balancing of Range-Partitioned Data with Applications to Peer-to-Peer Systems. In Nascimento, M. A., Özsu, M. T., Kossmann, D., Miller, J. R., Blakeley, J. A., and Schiefer, K. B., editors, *VLDB*, pages 444–455. Morgan Kaufmann.

Harman, D. K. (2008). Some thoughts on failure analysis for noisy data. In Lopresti, D., Roy, S., Schulz, K., and Venkata Subramaniam, L., editors, *Proc. 2nd Workshop on Analytics for Noisy unstructured text Data (AND 2008)*, pages 1–1. ACM Press, New York, USA.

Hearst, M. A. and Pedersen, J. O. (1996). Reexamining the Cluster Hypothesis: Scatter/Gather on Retrieval Results. In Frei, H. P., Harman, D., Schaübie, P., and Wilkinson, R., editors, *Proc. 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 1996)*, pages 76–84. ACM Press, New York, USA.

Jardine, N. and van Rijsbergen, C. J. (1971). The Use of Hierarchical Clustering in Information Retrieval. *Information Storage and Retrieval*, 7:217–240.

Järvelin, K. and Kekäläinen, J. (2002a). Cumulated Gain-Based Evaluation of IR Techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446.

Järvelin, K. and Kekäläinen, J. (2002b). Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information System*, 20:422–446.

Kekäläinen, J. (2005). Binary and Graded Relevance in IR Evaluations–Comparison of the Effects on Ranking of IR Systems. *Information Processing & Management*, 41(5):1019–1033.

Kendall, M. (1948). *Rank correlation methods*. Griffin, Oxford, England.

Keskustalo, H., Järvelin, K., Pirkola, A., and Kekäläinen, J. (2008). Intuition-Supporting Visualization of User's Performance Based on Explicit Negative Higher-Order Relevance. In Chua, T.-S., Leong, M.-K., Oard, D. W., and Sebastiani, F., editors, *Proc. 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2008)*, pages 675–681. ACM Press, New York, USA.

Kleinberg, J. M. (1999). Authoritative Sources in a Hyperlinked Environment. *J. ACM*, 46(5):604–632.

Langville, A. N. and Meyer, C. N. (2003). Survey: Deeper Inside PageRank. *Internet Mathematics*, 1(3):335–380.

Liu, T.-Y. (2009). Learning to Rank for Information Retrieval. *Found. Trends Inf. Retr.*, 3(3):225–331.

Liu, T.-Y. (2011). *Learning to Rank for Information Retrieval*. Springer.

Liu, T. Y., Xu, J., Qin, T., Xiong, W., and Li, H. (2007). LETOR: Benchmark Dataset for Research on Learning to Rank for Information Retrieval. In *SIGIR '07: Proceedings of the Learning to Rank workshop in the 30th annual international ACM SIGIR conference on Research and development in information retrieval*.

Lupu, M. and Hanbury, A. (2013). Patent Retrieval. *Foundations and Trends in Information Retrieval (FnTIR)*, 7(1):1–97.

Manning, C. D., Raghavan, P., and Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press, Cambridge, UK.

McCandless, M., Hatcher, E., and Gospodnetić, O. (2010). *Lucene in Action*. Manning Publications Co., NY, USA, 2nd edition.

McGill, R., Tukey, J. W., and Larsen, W. A. (1978). Variations of Box Plots. *The American Statistician*, 32(1):12–16.

Mizzaro, S. (1997). Relevance: The Whole History. *Journal of the American Society for Information Science and Technology (JASIST)*, 48(9):810–832.

Ounis, I., Amati, G., Plachouras, V., He, B., Macdonald, C., and Lioma, C. (2006). Terrier: A High Performance and Scalable Information Retrieval Platform. In Beigbeder, M., Buntine, W., and Yee, W. G., editors, *Proc. of the ACM SIGIR 2006 Workshop on Open Source Information Retrieval (OSIR 2006)*.

Ponte, J. M. and Croft, W. B. (1998). A Language Modeling Approach to Information Retrieval. In Croft, W. B., Moffat, A., van Rijsbergen, C. J., Wilkinson, R., and Zobel, J., editors, *Proc. 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 1998)*, pages 275–281. ACM Press, New York, USA.

Robertson, S. E. (1981). The methodology of information retrieval experiment. In Spärck Jones, K., editor, *Information Retrieval Experiment*, pages 9–31. Butterworths, London, United Kingdom.

Robertson, S. E. (1997). Overview of the Okapi Projects. *Journal of Documentation*, 53(1):3–7.

Ruggeri, F., Kenett, R. S., and Faltin, F. W., editors (2013). *Encyclopedia of Statistics in Quality and Reliability*. John Wiley and Sons Inc., NY, USA.

Salton, G., editor (1971). *The SMART Retrieval System. Experiments in Automatic Document Processing*. Prentice–Hall, Inc., Englewood Cliff, New Jersey, USA.

Salton, G. and McGill, M. J. (1983). *Introduction to Modern Information Retrieval*. McGraw-Hill, New York, USA.

Savoy, J. (2007). Why do Successful Search Systems Fail for Some Topics. In Cho, Y., Wan Koo, Y., Wainwright, R. L., Haddad, H. M., and Shin, S. Y., editors, *Proc. 2007 ACM Symposium on Applied Computing (SAC 2007)*, pages 872–877. ACM Press, New York, USA.

Sormunen, E. (2002). Liberal Relevance Criteria of TREC – Counting on Negligible Documents? In Järvelin, K., Beaulieu, M., Baeza-Yates, R., and Hyon Myaeng, S., editors, *Proc. 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2002)*, pages 324–330. ACM Press, New York, USA.

Tiu, T.-Y., Joachims, T., Li, H., and Zhai, C. (2010). Introduction to special issue on learning to rank for information retrieval. *Information Retrieval*, 13(3):197–200.

Tukey, J. W. (1970). *Exploratory Data Analysis*. Addison-Wesley, USA, preliminary edition edition.

Tukey, J. W. (1977). *Exploratory Data Analysis*. Addison-Wesley, USA.

van Rijsbergen, C. J. (1979). *Information Retrieval*. Butterworths, London, England, 2nd edition.

Voorhees, E. (2001). Evaluation by Highly Relevant Documents. In Kraft, D. H., Croft, W. B., Harper, D. J., and Zobel, J., editors, *Proc. 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2001)*, pages 74–82. ACM Press, New York, USA.

Voorhees, E. and Harman, D. (1999). Overview of the Seventh Text REtrieval Conference (TREC-7). In *NIST Special Publication 500-242: The Seventh Text REtrieval Conference (TREC 7)*. Springer-Verlag, Heidelberg, Germany.

Voorhees, E. M. (1985). The Cluster Hypothesis Revisited. In Tague, J. M., editor, *Proc. 8th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 1985)*, pages 188–196. ACM Press, New York, USA.

Willett, P. (1988). Recent Trends in Hierarchic Document Clustering: A Critical Review. *Information Processing & Management*, 24(5):577–597.

Witten, I. H., Bainbridge, D., and Nichols, D. M. (2009). *How to Build a Digital Library*. Morgan Kaufmann Publishers, San Francisco (CA), USA, 2nd edition.

Zhang, J. (2008). *Visualization for Information Retrieval*. Springer-Verlag, Heidelberg, Germany.