









# Exploring Ontology-Based Information Through the Progressive Disclosure of Visual Answers to Related Queries

Dalai S. Ribeiro<sup>(✉)</sup> , Alysson Gomes de Sousa<sup>(✉)</sup> ,  
Rodrigo B. de Almeida<sup>(✉)</sup> , Pedro Henrique Thompson Furtado<sup>(✉)</sup> ,  
Hélio Côrtes Vieira Lopes<sup>(✉)</sup> , and Simone Diniz Junqueira Barbosa<sup>(✉)</sup> 

Pontifícia Universidade Católica do Rio de Janeiro,  
Rio de Janeiro, RJ 22451-900, Brazil

[dalai.ribeiro@gmail.com](mailto:dalai.ribeiro@gmail.com), [{asousa,rbarbosa,lopes,simone}@inf.puc-rio.br](mailto:{asousa,rbarbosa,lopes,simone}@inf.puc-rio.br),  
[pedrothompson@petrobras.com.br](mailto:pedrothompson@petrobras.com.br)

**Abstract.** Web search has become the predominant method for people to fulfill their information needs. Although widespread, the traditional model for search result pages is only satisfactory if the user knows quite precisely how to phrase their query to match their intended information. We propose a new model for search page results, which goes beyond providing a navigable list of visualization search results, by implicitly generating related queries to expand the search space and progressively disclosing the corresponding results.

**Keywords:** Design · Information design · Information retrieval · UX and usability · User interface of search result pages

## 1 Introduction

“In a little more than a decade, the Web has become the default global repository for information” [15]. Search has remarkably contributed to this result and it has become ubiquitously associated with the Web itself, to the point of becoming a default tool in any modern browser and one of the most popular activities online, already in 2008 [5].

As stated by Wilson et al. (2010) [15], “Web search, as provided by Google, Microsoft, Yahoo, etc., allows users to find the information they need via the simplest of interaction paradigms”: the user types in keywords or a natural language query and obtains a related ranked result list. If the results do not fulfill the user’s information needs, he/she may create a new query to obtain new results, making the information seeking process naturally iterative [3].

Our work focuses on searches whose results can be represented as data visualizations. At first glance, this may seem similar to “image search” mechanisms.

---

Supported by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES).

However, as data visualizations usually represent underlying structured data, the known relations between the data points and data sets can be used as input to search expansion mechanisms. In our work, we assume that the data are described by an ontology [7], such as those we can find in linked-open data (LOD)<sup>1</sup>, *e.g.*, DBPedia<sup>2</sup>.

Traditionally, a user submits a search query through a search dialog box and, in response to the query, a search engine delivers one or more search result pages (SRPs) to the user. SRPs often consist of multiple pages of items that are related to the search query submitted by the user. Most of the initial search results are closely related to the query but, as the user navigates to later results, they are increasingly less related to it.

As the users may need to navigate through many SRPs [2], from their point of view, the traditional model is only satisfactory if they know quite precisely how to phrase their query to match their intended search for information.

Let us consider as an example a user named Jack who wants to know the movie *genre* that generated the highest box office in 2018, but who formulates the following query: “Which *movies* had the highest gross revenue in 2018?” In this case, the search results would likely contain a list of the top individual grossing movies, with links to details about each movie. Jack might then think he would need to inspect the movies one by one to try to figure out to which genre most of them belong, a very tedious task. When inspecting a movie, Jack may see that there is genre information associated to each movie and, realizing this is the term he should include in the query, he might reformulate the query to “Which *movie genre* had the highest gross revenue in 2018?”, which then brings the intended information in the search results. This scenario has a successful ending, but in many other situations the user cannot figure out the specific query formulation needed to find the intended information.

In this paper we propose a new model for search user interfaces, focusing on the search results page. Our proposal goes beyond providing a navigable list of visualization search results. It includes an API for implicitly generating related queries to expand the search space, and progressively discloses the corresponding results. Our hypothesis is that such mechanism can improve the user interaction with search results, especially in situations where users cannot figure out how to formulate the precise query to yield the intended results.

## 2 Related Work

Wilson (1999) [16] defines information-seeking behavior as a set of activities that people engage in when identifying their information need, searching for it through an information resource, and using the results to satisfy that need.

Understanding human information-seeking processes is the foundation for the design of effective and usable search systems [16]. In the next sections, we describe existing models of information-seeking behavior.

<sup>1</sup> <https://lod-cloud.net/>.

<sup>2</sup> <https://wiki.dbpedia.org/>.

## 2.1 Iterative Model of Information-Seeking Behavior

Marchionini (1997) [11] laid the foundation for the traditional information-seeking process, defining it as a set of “systematic and opportunistic” subprocesses.

In this work, we will use a simplified version of this process defined by Hearst (1999) [9]. This model also assumes the process is iterative and that the user information need does not change. The model comprises the following sequence of steps [9]:

1. “Recognize the information need.
2. Select the information repository to search.
3. Form a search query.
4. Send the query to the system.
5. Receive the results.
6. Evaluate and interpret the results.
7. Stop, if the information need is fulfilled, or
8. Reformulate the query and return to Step 4.”

Although widespread, the iterative model of information seeking does not capture the richness of genuine information-seeking processes [10], especially because the users’ information demands may change during the search process as a result of their interaction with the search system. The user can at the same time present a behavior that is both systematic and unsystematic, starting their search processes following the hierarchical approach presented by Hearst (2009) [10], and then switching to a more dynamic behaviour that uses the initial result set as a starting point that informs further queries, as pointed out by Marchionini (1997) [11]. Marchionini also advocates that, because individual factors affect information-seeking interaction, there is a need for new models that better account for the dynamic nature of information seeking, *i.e.*, models that can address the challenges of describing how users employ different search tactics and how they can make sense of the results.

## 2.2 Exploratory Search

The traditional information-seeking method is well supported by search engines, especially when the user has well-defined information needs. However, when the user lacks the knowledge or contextual awareness to formulate queries or navigate complex information spaces or the information, the search system should provide more support for a complex information seeking process, where the user is able to browse and explore the results in order to fulfill their needs [15].

Exploratory Search research tackles this issue by studying information-seeking models that blend querying and browsing with a focus on learning and investigating, instead of information lookup [12]. White et al. (2005) [14] distinguish three typical situations in which exploratory search happens:

- The user has partial or no knowledge of the search target

- The search moves from certainty to uncertainty as the user is exposed to new information
- The user is actively seeking useful information and determining its structure

O’Day and Jeffries (1993) [13] describe this incremental search behavior as a process of exploration through a series of related but distinct searches on a specific topic. They identify three distinct search modes:

- “Monitoring a well-known topic over time;
- Following a plan of information gathering;
- Exploring a topic in an undirected fashion.”

This shows that even exploratory information seeking has structure and continuity, which could be supported by the search system.

### 3 Progressive Disclosure of Related Search Results

Many systems allow the user to navigate through search results by refining the search query. Although effective when the user has a clear vision of their interests, those interfaces may not be very suitable when the user is performing an exploratory search or cannot properly formulate their information need.

Some systems, such as Datatone [6], allows the user to navigate through related questions by direct manipulation of the query or through manual interactions with its user interface (in Datatone, through “ambiguity widget”). In other words, Datatone requires users to plan and take action in order to obtain related search results.

We hypothesized that, instead of requiring users to manually adjust the queries to amplify their search results, user interfaces for searching data visualizations might continuously offer answers to related queries by navigating through an underlying ontology. Our proposed search user interface, named *JARVIS - Journey towards Augmenting the Results of Visualization Search*, is based on the progressive disclosure model used by Google Images,<sup>3</sup> where the interface continuously appends content to the search results page. Rather than requiring users to refine their queries, JARVIS automatically amplifies the set of results with answers to related queries.

#### 3.1 Research Context

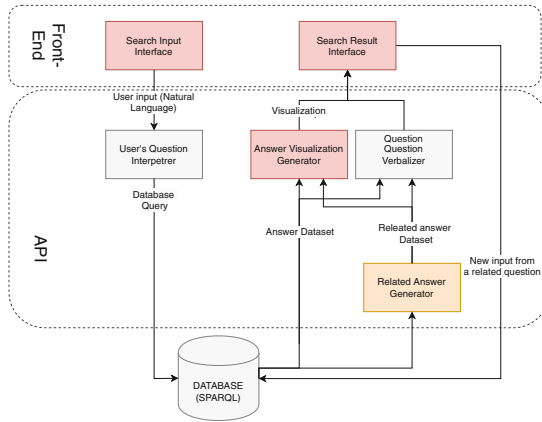
Our work focuses on understanding how to better support the user by designing a result page that is both effective and efficient. More specifically, we propose and evaluate a progressive disclosure mechanism for related questions. For that to happen, many other parts need to be in place.

Figure 1 presents the JARVIS architecture. The topmost part represents the interface of the system, where the user can input their questions and visualize

---

<sup>3</sup> <https://images.google.com/>.

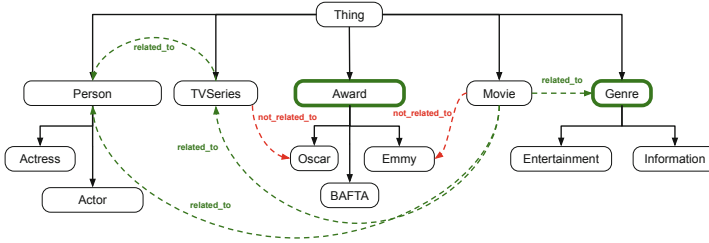
the answers. Take the question “Which actresses won the most Golden Globe awards last year?” as an illustration. The interpreter, fed with a movie ontology, identifies the known entities, such as *actresses* ( $Actress \rightarrow is \rightarrow Person$ ), *Golden Globe Awards* ( $Award \rightarrow has\_name \rightarrow Golden\ Globe\ Awards$ ), relationships, in this case, *won* ( $Person \rightarrow awarded \rightarrow Award$ ) and the temporal attributes like *last year*. With this information, the system transforms the question into an RDF query that accesses the domain database and gathers the answers.



**Fig. 1.** JARVIS architecture

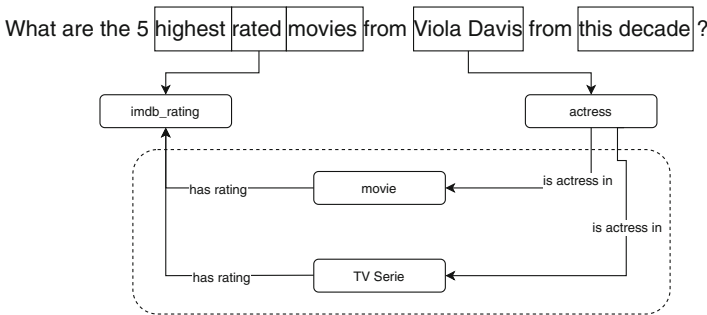
Also crucial is the generation of the relevant related questions whose answers that will be represented as data visualizations to the user. Figure 3 shows an example of how the system generates its related questions. In the question “What were the 5 highest rated movies from Viola Davis this decade?”, after identifying the known entities from the ontology in the question, it looks for the structure of their relationships. In this case:  $actress \rightarrow is\_actress\_in \rightarrow movie \rightarrow has\_rating \rightarrow imdb\_rating$ .

In order to improve the effectiveness of the system, a domain expert can enrich the ontology with relationships that they find interesting to the users of the search engine. Figure 2 shows the simplified ontology proposed by Calvanese et al. (2017) [1], with annotations. This process is especially important when dealing with large and complex ontologies. In our case, the ontology used is fairly small and simple. This aspect enables us to have good results even when we have skipped this stage when building our system. We applied the same methodology for a different domain for a large company of Oil & Gas in Brazil. This latter ontology was significantly larger and presented more complex relationships. Because we did not have access to domain experts in this case, the related question could present variations that would make no sense to the final user of the search system. In a case like this, developers of such system should prioritize direct relationships in the ontology to build the related questions mechanism.



**Fig. 2.** Simplified ontology of Calvanese et al. (2017) [1] with annotations

With that structure defined, it starts to scan for possible traversals in the ontology which are relevant for generating a related question. Those traversals can occur in various ways, for example from an entity to its parent (movie  $\rightarrow$  production) or, like in Fig. 3, to a sibling entity, that is, an entity that belongs to the same structure to the one identified in the user query. In this case, the system can change ‘Movies’ to ‘TV Series’ and select “What were the 5 highest rated *TV series* from Viola Davis this decade?” as the new query.



**Fig. 3.** Relation between a question in natural language and the corresponding elements in an ontology.

### 3.2 The Related Question Mechanism

In order to enhance the answers generated by the interpreter and to reduce the user’s cognitive effort to formulate other related questions which may interest them, our group developed a mechanism that recommends answers to questions related to the initial question the user searched. This mechanism applies operations to the ontology, taking into consideration the entities that were detected in the initial question (see Fig. 3).

Figure 4 depicts the JARVIS interface. Let us take the example described in Sect. 1: the information needed by the user is the movie *genre* that generated the highest box office in 2018, but when formulating their query they typed: “Which

*movies* had the highest gross revenue in 2018?”. JARVIS sends, through an API, the natural language query written by the user. The API looks for the literal answer or answers to the question and ranks the results. It then exhibits the  $n$  highest ranked direct, literal results for the query on the topmost area of the interface, in a slightly shaded area (Fig. 4). Below that area, it progressively displays results from related questions, which are gradually received from the API. Those results are the outcomes of a search mechanism that, given a domain ontology (e.g., related to the IMDB), navigates through the ontology looking for useful relationships between the elements presented in the search query to expand the given question into related ones. JARVIS may offer, for example, results for questions such as “Which *studios* had the highest gross revenue in 2018?” (through a movie–produced by–studio relationship), “Which *movies* had the highest gross revenue in 2018 per country?” (through a movie–produced in–country relationship), and “Which *movie genre* had the highest gross revenue in 2018?” (through a movie–classified as–genre relationship). These related questions may offer the information needed by the user, as well as different perspectives on the data related to the query, without any manual interaction by the user.

In this work, we focus on the delivery mechanism for the results and on how the users interact with it. The challenges of translating a natural language question to a database query, and of navigating in a ontology to find the useful relationships for related questions are relevant research topics, which are currently being developed by other members of our research group, and therefore lie outside the scope of this work.

## 4 Evaluation

Our proposed solution progressively discloses results for related queries. To evaluate the effectiveness and efficiency of our solution, we have devised two other search user interface (SUI) models for the same search task. The first uses the traditional search interaction method described by Wilson (1999) [16] (henceforth called *Traditional SUI* (J1)), and the second is built showing the related questions as links to explore the results (henceforth called *Related-links SUI* (J2)). We then conducted a comparison test of three SUIs. We invited graduate students from different areas to serve as volunteer participants in the study.

The *Traditional SUI* (J1) (Fig. 5) is an almost direct representation of the work described by Wilson (1999) [16]. The user types a search query and receives the highest ranked result for their question. The only way they can obtain further search results is by manually editing or typing a new query for the system, which again will only return the highest ranked result. This model represents a baseline for our work, whereas the interface, although pedestrian, is straightforward and familiar to the participants.

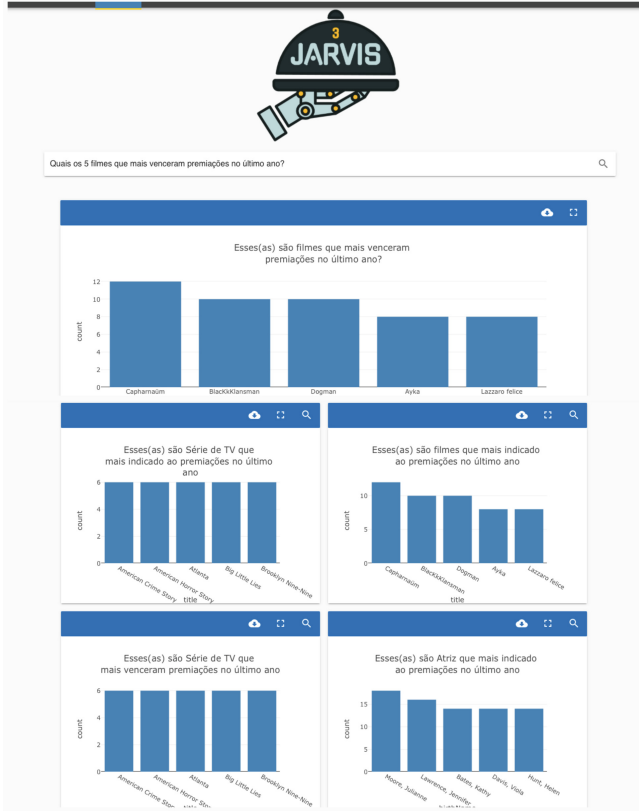


Fig. 4. JARVIS search user interface

The *Related-links* SUI (J2) (Fig.6) introduces a suggested list of related questions. The user is now presented not only with the highest ranked result, but also with a set of related questions on a lateral pane. This allows the user to navigate through related questions more quickly, but still requires manual interaction with the user interface. Model J2 is presented to the participants so we can attempt to understand whether the mere introduction of related questions is enough to reduce the users' cognitive overload and to build a more effective search interface.

To evaluate the interface models, we conducted an empirical comparative test of the three SUIs. We invited graduate students from different areas to serve as volunteer participants in the study. To reduce the learning effects, we varied the order in which each SUI is presented to the users, using the configuration shown in Table 1. With this, we attempted to see if the order on which the user experienced each model affected their evaluation. Since model J1 required more effort to complete the suggested task, we hypothesized that users that had contact with the model J1 prior to the other models would find the introduction



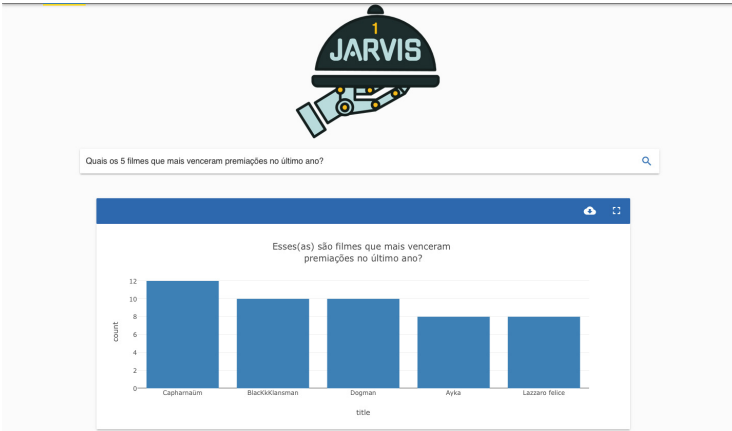


Fig. 5. User interface of the Traditional SUI (J1)

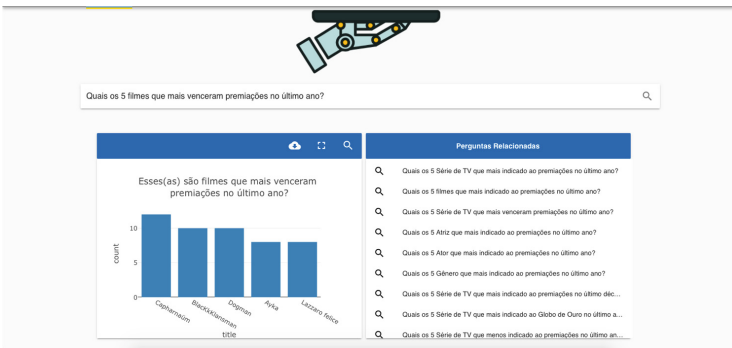


Fig. 6. User interface of the Related-links SUI (J2)

of mechanisms for searching the related queries more useful. That would be especially true with users in group A, where the mechanism scaled in complexity gradually. The user started the experiment with only a straightforward search mechanism in J1, to later test an interface that presents he/she with related queries in J2, to finally evaluate our proposal. This order would help to gradually raise the awareness of the related questions and its answers. By contrast, we also hypothesized that participants in group C might get confused with non-traditional features of model J3 and evaluate it poorly.

Fifteen people participated in the experiment to evaluate the delivery mechanism of the related queries: three females (P01, P04, P14) and 12 males (P02, P03, P05, P06, P07, P08, P09, P10, P11, P12, P13, P14). They were all graduate students at PUC-Rio (11 Master’s students, and 4 PhD students). Apart from P14, who is a psychology student, all the participants were Computer Science students. Eleven participants (P01, P02, P03, P04, P05, P06, P07, P08, P10, P11, P15) fell within the 18–24 age group. Only four participants (P09,

**Table 1.** Experiment groups

Group	Order of SUIs
A	J1 J2 J3
B	J2 J1 J3
C	J3 J2 J1

P12, P13, P14) were 25 to 44 years old. Regarding their previous knowledge of the models, all participants were familiarized with traditional search tools. Four of the participants had already seen a search user interface similar to J3 in another context, but had not used it (P02, P03, P05, P06). We henceforth call these “participants with little previous knowledge”. One participant – henceforth “Developer” – helped develop J3 for an R&D project (P11). The other ten participants had no knowledge of models J2 and J3 – henceforth “participants with no previous knowledge”.

For each SUI, the user received six search tasks, each one representing a search query. We devised the queries in two groups, one which had two related queries and other with four related queries, but we did not inform users of such grouping. Such grouping was designed so that in the Related-links SUI (J2) and our proposal (J3), participants would need to type only two queries, and then they would have quick access to the remaining related queries through the links at the right-hand panel. In the Traditional SUI (J1), however, the user would need to type in each of the six queries manually. For each group, when the participant asked the first question, the results pages also presented either the questions of the following tasks (on J2) or their answers (on J3). The groups were also designed in such a way that the related question ranked high in each related queries mechanism of J2 and J3, except for the last question of the first group (Quais as 5 Séries de TV de menores durações? – What are the 5 TV Series of shortest duration?), which was intentionally more distant and thus required the user to scroll the related queries component (on J2) or the screen (on J3).

The content of the tasks varied: to discover the five movies that had won the most awards last year (through a *Movie-won award-Awards* relationship) and related information such as the 5 TV Series that won most awards last year (through a *TV Series -won award-Awards* relationship) and in the last decade (through a time variation).

After interacting with each SUI, we asked the participant to fill out a questionnaire regarding the perceived ease of use and utility of the SUI – based on the Technology Acceptance Model (TAM) [4] –, and their subjective workload assessment – based on the NASA Task Load Index<sup>4</sup> [8]. At the end of the session, we briefly interviewed the users, asking them to choose their preferred SUI and explain the factors that led them to their choice. We also collected performance data in terms of effectiveness (correctness of the result) and efficiency (time on task). In particular, we used the number of searches as a proxy for efficiency.

<sup>4</sup> <https://humansystems.arc.nasa.gov/groups/TLX/>.

We expected that model J3 followed by J2 would present better results in the TAM questionnaire due to the introduction of mechanisms that offer more ways to explore the search results. However, participants might find that the new interfaces require from them a more significant effort, resulting in poor results on the NASA TLX.

Unfortunately, a few days before the experiments, the API we used as a service to our work became faulty and behaved erratically. During the pilot test, we identified that too often the user interactions with the system would cause the server to restart or not to load data correctly. To build a workaround solution significant changes in the performance of the interface were needed, slowing down the chart load in several seconds in order to ensure that the data presented to all the participants would be consistent. This prevented us from analyzing the time on task.

## 5 Results

### 5.1 NASA Task Load Index Results

Figure 7 shows the results of the entire NASA Task Load Index questionnaire, discussed in detail in the next subsections.

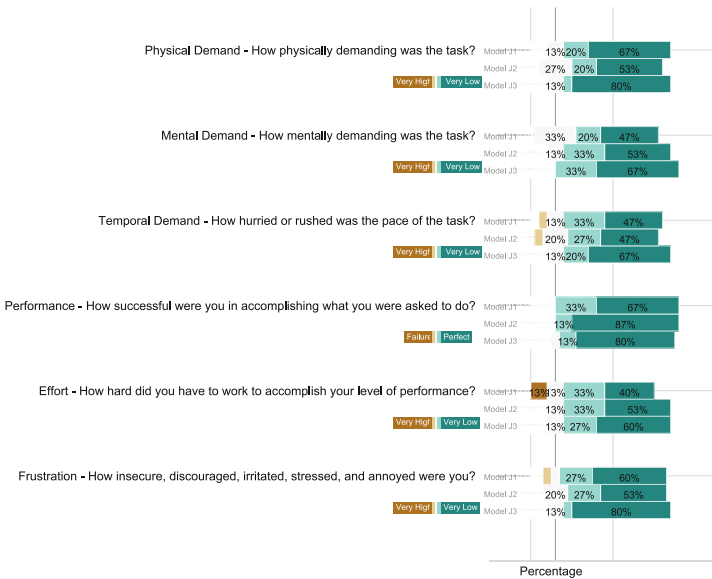


Fig. 7. NASA Task Load Index results

*Mental Demand:* The task in the experiment was relatively simple. However, as the complexity of the models grew (J2 is more complicated than J1, and J3 is more complex than J2), we expected that the questionnaire scores for J1 would be better than for J2 and J3. During the interviews, this hypothesis

seems to gather even more support mainly because of what participants (P01, P02, P03, P04, P010, P13, P14) called “lack of resources” of model J1 and their assessment of J2 and J3. Regarding the other models, we expected results from the questionnaire to follow the comments of the participants that the models J2 (P11, P14) and J3 (P3, P4, P5, P8, P9, P11, P13, P14) would be better suited to multiple search tasks, thus ranking worse than J1. Figure 7 seems to be in accordance with our hypothesis and shows an advantage of J1 for the *Mental demand* measurement over the other models.

*Physical Demand:* The measurement may be a reflection of the number of clicks or the number times the user has to manually inform the query in the main search bar to complete the tasks. In J1, the user had no other choice but to insert the search queries six times, so we might hypothesize that, although the task itself was not physically troublesome, models J2 and J3 would be rated slightly better because they offered options that did not involve typing or copy-and-pasting new queries in to the system. Figure 7 seems to show a slight advantage of J3 for the *Physical demand* measurement over the other models.

*Temporal Demand:* Since model J1 required more interaction and clicks at the user interface than models J2 and J3, we hypothesized that J1 would perform worse on this measurement than the other models. However, during the interviews, we noticed that the model J2 had drawn polarized opinions from participants. This may have been an effect from the limitations of J2. However, we believe that faulty design implementation presented in J2 played a prominent role on those participants’ commentaries. Considering the component was not intuitive, and the text font was quite small, the interaction with the component was deeply affected, and its problems may have overshadowed its virtues. Figure 7 seems to be in accordance with our hypothesis and shows an advantage of J3 for the *Temporal demand* measurement over J1. Unsurprisingly, model J2 seems to have a slight worse temporal demand evaluation than J1 and J3.

*Performance Demand:* The results may have been profoundly affected by a severe problem with the experiment: the server performance. Because the server was very fragile, the system was slower than usual. This problem affected primarily the user interface of the J3 model, which is, by far, the model that needs to receive a larger volume of data to build the visualizations at the user interface. These issues may have influenced the performance scores in the questionnaire, leaving J2 slightly better ranked than J3. Figure 7 seems to be in accordance with our hypothesis, showing an advantage of J2 for the *Performance demand* measurement over the other models.

*Effort:* Similar to the Physical demand measurement, the effort measurement may be a reflection of the number of times the user had to manually inform a query in the main search bar to complete the tasks. Surprisingly, users ranked even J3 poorly, acknowledging that the user interface could further reduce the user effort on searching. Moreover, J1 is the only model that offers no other option to complete the task, *i.e.*, it requires that all queries be informed, one by one, in the search bar. For that matter, we hypothesized that J3 would perform

better than J2 and that J2 would perform better than J1. Figure 7 seems to be in accordance with our hypothesis and shows an advantage of J3 for the *Effort* measurement over the other models.

*Frustration*: Because J3 is the model that it was more complicated than the others, it also had the effects of the problems with the server being more prominent in the J3. When the participants were exploring the interface of J3, the server would often crash and require a reboot in order to become functional again. Because of that, we expected to J3 to be lowest-ranked model in this measurement. Surprisingly, Fig. 7 seems to debunk our hypothesis and shows a slight advantage of J3 for the *Frustration* measurement over the other models, even with the problems on the interface design and the server malfunctions. However, the Kruskal-Wallis hypothesis tests for each measurement of the questionnaire showed no significant difference among the models, at  $\alpha = 0.05$  (either considering all users, only users with little knowledge, and only users with no knowledge).

## 6 Technology Acceptance Model (TAM) Results

Figures 8 and 9 show the results of the Technology Acceptance Model questionnaire, discussed in detail in the next paragraphs.



Fig. 8. TAM results (part 1)

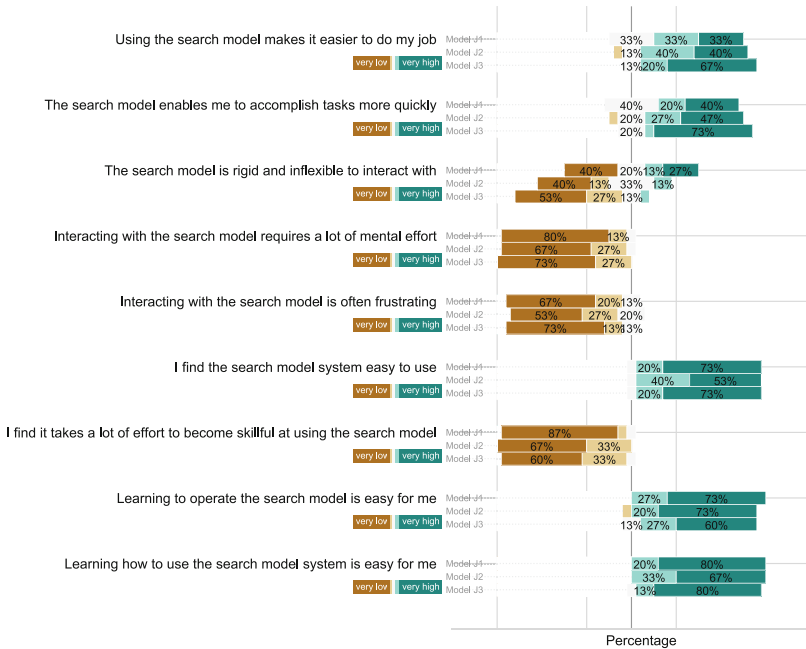


Fig. 9. TAM results (part 2)

Because the questionnaire is fairly long, we decided to not discuss each result in full detail. Instead, since the Technology Acceptance Model evaluates the *perceived usefulness* and the *perceived ease of use*, we will discuss the overall results of these dimensions with a selected example.

Regarding the *perceived ease of use*, because of the nature of the task, all models performed reasonably well. For example, in the question “*I find the search model system X easy to use*”. The figure shows model J2 as worse than models J1 and J3. This seems to contradict our hypothesis that the model J1 is easier to use than models J2 and J3. Instead, it shows identical results to J1 and J3. This means that users do not think JARVIS (J3) is harder to use than the other models, even though it has more complex features than the other models. The unfortunate result of the model J2 may be a consequence of the faulty user interface design choices, that some participants reported as “confusing”.

The *perceived usefulness* was evaluated with questions such as “*The search model X enables me to accomplish tasks more quickly*”. With this item, our goal was to evaluate whether the perceived evaluation from the participants matched their actual time on task. Unfortunately, due to the problems experienced in the server, we were unable to execute this analysis. We expected that model J2 and J3 would perform better than J1 in this item because it offers the answer or question for the other task more efficiently, not requiring the participant to type each search query to finish the experiment tasks. The results shown in Fig. 9

confirms our prediction for the model J3, which had the best evaluation among the participants. Despite presenting to the user an alternative way to search, the model J2 was the worst-ranked among all three models. These results may indicate that the mere recommendation of related questions is not enough to support the user while navigating on search result pages more quickly, or be an outcome of an unrefined user interface design.

## 7 Interviews

Table 2 shows a compilation of the common critiques reported by the participants of the study. We categorized each comment into four categories: Model, Design, Configuration, and Implementation. In *Model*, we selected the comments more closely related to intrinsic aspects of each model and which would likely remain true even if significant changes to the design or implementation of the system were made. In *Design*, we summarized comments related to the user interface design. The comments reported under the *Configuration* category are those that may be related to the parameterization and flexibility of how the related questions are calculated and prioritized. In *Implementation*, we compile the observations strongly associated with the system performance.

### 7.1 Model

Regarding the Model, most of the comments from the participants are related to the occasions in which they seem to be adequate and what the model's intrinsic, positive or negative characteristics are and which had the biggest influence on the participants' experience. Participants mostly commented on the differences between model J1 and models J2 and J3. Most of them (12 participants) found that the model J1 is simple or straightforward, a characteristic that they consider a positive aspect of the model. In contrast, they found that executing the task using the model J1 was very time consuming for multiple searches (5 participants) and better suited for when the user knows what they want (4) or only needs to do a quick search (4 participants). Although in the participants' view the model J1 lacks resources (7 participants), some of the participants perceive this as a positive aspect, because the user interface does not distract the user from the task they are executing (2 participants).

These perspectives of model J1 contrast with what participants said about the models J2 and J3. Furthermore, whereas they found it difficult to use model J1 in long or exploratory tasks, they highlighted the benefits of using the other models for those scenarios. For example, two participants reported that, because both models presented them with related questions, they were able to have insights of new questions that they may not have had if they were only exploring the data following the traditional interface of model J1. They also found that both models support scenarios where the user needs to make multiple searches or search tasks that are broader or exploratory (2 participants for J2 and 4 for





J3). Those comments are in line with the conceptual design of those models, as described in Sect. 4.

Besides that, a crucial point for this research is how the recommendations affected the user interaction. We did not directly ask of participants questions about the perceived effectiveness of the recommendation to avoid inducing certain answers. However, participants spontaneously evaluated aspects of the recommendation. The majority (7 for model J2 and 10 for model J3) agreed that the recommendation makes it easier to achieve the search task. Those comments are an indication that the models J2 and J3 are potentially efficient and useful for the search task, especially when the user is on an exploratory task.

## 7.2 Design

In the Design category, we outline here the most common complaints from the participants that we believe could be solved by redesign the user interfaces. These include the related questions component from model J2. Although it does what is supposed to do, the component could have been better designed to highlight the differences between questions to reduce the amount of text the user needs to read. Eight participants evaluated that the model J2 was harder to interact with because it had too much text to read, and they evaluated it as worse to read than charts. Three of those participants even added that the model J2 was exhausting for them to use.

However, regarding the differences from J1 to models J2 and J3, there is a trade-off between using the chart or the text component. Although participants noted that the use of text could be exhausting, they also elucidated on problems with model J3, such as the need to scroll the interface (3 participants) and some (3 participants) even stated that they found the interface distracted them from the task they needed to perform.

## 7.3 Configuration

The Configuration category represents issues that may be related to either the parameterization or the flexibility in how the related questions are calculated and prioritized.

In models J2 and J3, users complained about the selection of the related questions/answers that were exhibited. Although this mechanism is out of the scope of this work, we must develop a better algorithm for related questions/answers in order to better evaluate the models we have proposed. Participants even commented about the lack of coherence between the related questions and the main query they searched. In other words, in order for models such as J2 and J3 to thrive, as a better alternative to the design of search result pages, it is vital that the ranking engine of those questions/answers be effective.

## 7.4 Implementation

Concerning the implementation category, we were interested in how the system performance affected the participants' experience with each model. Since our

implementation suffered from many issues, as mentioned before, we looked in the interviews for what how those problems influenced their evaluation.

In that regard, two participants reported that the model J3 was too slow. This is a direct effect of the fact that the model J3 was the one most affected by those problems.

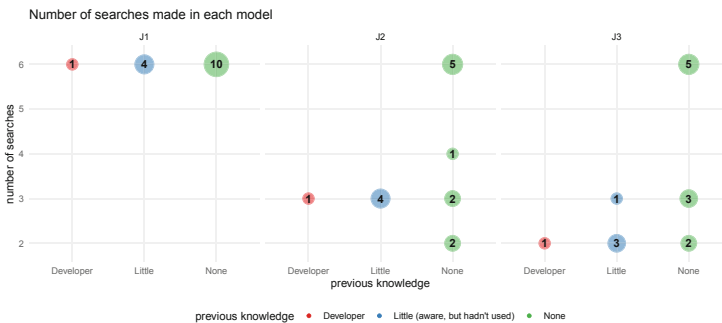
## 7.5 Efficiency

As mentioned before, we were unable to directly measure the time on task. However, we counted the number of explicit searches the participants did during the experiment. This can be considered as an indirect indication of efficiency of each model. It is flawed, however, as it does not take into account the actual time it took for participants to locate the related questions when using models J2 and J3.

Figure 10 shows the number of searches made in each model, by each group, according to the participants' previous knowledge. It shows the median number of searches and corresponding interquartile range using each model, in the group of all users. We note that model J1 always requires six searches to complete the task.

We conducted statistical analyses of the differences in the number of searches across models in three different groups: all users, users with little previous knowledge of the models, and users with no previous knowledge of the models, as described in the next subsections. We ran a Kruskal-Wallis rank sum test, which showed a **significant difference** at  $\alpha = 0.05$  level ( $\chi^2 = 14.600$ ,  $df = 3$ ,  $p\text{-value} = 0.0022$ ). We therefore ran a Conover-Iman post-hoc test, with Bonferroni correction. There was a significant difference in the number of searches between J1–J2 and J1–J3, in the group of all users.

We also investigated whether there was a significant difference between the number of searches across models considering only users with little knowledge of the models. We ran a Kruskal-Wallis rank sum test, which showed a **significant difference** at  $\alpha = 0.05$  level ( $\chi^2 = 9.900$ ,  $df = 2$ ,  $p\text{-value} = 0.0071$ ). We therefore ran a Conover-Iman post-hoc test, with Bonferroni correction and found that



**Fig. 10.** Bubble plot of number of searches using each model

was a significant difference in the number of searches between J1–J2, J1–J3, and J2–J3, in the group of users with little previous knowledge of the models.

Considering users with NO knowledge of the models, we ran a Kruskal-Wallis rank sum test, which showed no significant difference at  $\alpha = 0.05$  level ( $\chi^2 = 5.872$ ,  $df = 3$ ,  $p\text{-value} = 0.1180$ ).

## 8 Discussion

Although many of the results presented in this work are not statistically significant, it is essential to note that the evaluation of JARVIS (model J3) did not perform worse than the other models and, in most cases, it received a better evaluation from the user than models J1 and J2. These results indicate that, even though JARVIS is more complex and less familiar than the other models, from the user perspective it is a potential solution for the design of search result pages that enhance the user experience when doing an exploratory search.

The results in Sect. 5 imply that, between the traditional search behaviour of model J1 and the exploratory based of models J2 and J3, there is a significant difference. These differences imply that since the results from the TAM and NASA TLX were mostly in pair with the other models, the alternative models are possible useful solutions and should be explored more in-depth in future works.

Based on our results, we believe a possible alternative solution for the design of search results pages such as the one we proposed in this study may be a hybrid of from the models evaluated in this research. This hybrid model can be displayed in at least two forms of interface design:

1. A new model that shows the related question as a preview and only unfolds the visualization when the user explicitly ask.
2. An adaptative interface model that increases the personalization of search results pages by showing or hiding the related answers to the user.

We leave these hybrid models for future work.

## 9 Conclusion

The main contribution of this paper is a model to amplify cognition for search tasks. The model involves generating and presenting related queries to expand the search space and progressively disclosing the corresponding results.

We conducted an evaluation of the proposed model (J3) in comparison with two distinct search user interface models for data visualization: a *Traditional* SUI Model (J1), inspired by the work of Wilson (1999) [16], and a *Suggested-links* SUI (J2), which combines J1 with a suggested list of related questions.

The outcomes of the analysis suggest that the model proposed with JARVIS may be a promising path for the design of new SUIs. The results from the Task Load Index and Technology Acceptance Model Questionnaires showed that,

although J3 presents a more complex user interface and more features, it did not perform worse than the other models evaluated by the questionnaires. Moreover, even in the simple experiment we conducted, the number of searches made with J3 was significantly lower than the number of searches made with the Traditional SUI (J1), as shown in Sect. 5.

## References

1. Calvanese, D., et al.: Ontop: Answering SPARQL queries over relational databases. *Semant. Web* **8**(3), 471–487 (2017)
2. Chen, H., Dumais, S.: Bringing order to the web: automatically categorizing search results. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 145–152. ACM (2000)
3. Chowdhury, G., Chowdhury, S.: *Introduction to Digital Libraries*. Facet Publishing, UK (2002). <https://strathprints.strath.ac.uk/2603/>
4. Davis, F.D.: Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Q.* **13**(3), 319–340 (1989). <https://doi.org/10.2307/249008>
5. Fallows, D.: Search Engine Use—Pew Research Center, August 2008. <https://www.pewinternet.org/2008/08/06/search-engine-use/>
6. Gao, T., Dontcheva, M., Adar, E., Liu, Z., Karahalios, K.G.: DataTone: managing ambiguity in natural language interfaces for data visualization. In: *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. UIST 2015, pp. 489–500. ACM Press, New York (2015). <https://doi.org/10.1145/2807442.2807478>
7. Gruber, T.R.: A translation approach to portable ontology specifications. *Knowl. Acquis.* **5**(2), 199–220 (1993). <https://doi.org/10.1006/knac.1993.1008>
8. Hart, S.G.: NASA-task load index (NASA-TLX); 20 years later. *Proc. Hum. Factors Ergon. Soc. Annu. Meet.* **50**(9), 904–908 (2006). <https://doi.org/10.1177/154193120605000909>
9. Hearst, M.A.: *User Interfaces and Visualization*, vol. 2. Addison-Wesley, Boston (1999). <https://ci.nii.ac.jp/naid/10022005353/en/>
10. Hearst, M.A.: *Search User Interfaces*, 1st edn. Cambridge University Press, New York (2009)
11. Marchionini, G.: *Information Seeking in Electronic Environments*. Cambridge Series on Human-Computer Interaction. Cambridge University Press, Cambridge (1997)
12. Marchionini, G.: Exploratory search: from finding to understanding. *Commun. ACM* **49**(4), 41–46 (2006). <https://doi.org/10.1145/1121949.1121979>
13. O’Day, V.L., Jeffries, R.: Orienteering in an information landscape: how information seekers get from here to there. In: *Proceedings of the INTERACT 1993 and CHI 1993 Conference on Human Factors in Computing Systems*. CHI 1993, pp. 438–445. ACM, New York (1993). <https://doi.org/10.1145/169059.169365>
14. White, R.W., Kules, B., Bederson, B.: Exploratory search interfaces: categorization, clustering and beyond: report on the XSI 2005 workshop at the human-computer interaction laboratory, University of Maryland. *SIGIR Forum* **39**(2), 52–56 (2005). <https://doi.org/10.1145/1113343.1113356>

15. Wilson, M.L., Kules, B., Shneiderman, B., et al.: From keyword search to exploration: designing future search interfaces for the web. *Found. Trends® Web Sci.* **2**(1), 1–97 (2010)
16. Wilson, T.: Models in information behaviour research. *J. Doc.* **55**(3), 249–270 (1999). <https://doi.org/10.1108/EUM0000000007145>