Profiling user belief in BI exploration for measuring subjective interestingness

Alexandre Chanson
University of Tours
Blois, France
alexandre.chanson@etu.univ-tours.fr

Ben Crulis
University of Tours
Blois, France
ben.crulis@etu.univ-tours.fr

Krista Drushku
SAP
Paris, Levallois-Perret
krista.drushku@sap.com

Nicolas Labroche
University of Tours
Blois, France
nicolas.labroche@univ-tours.fr

Patrick Marcel
University of Tours
Blois, France
patrick.marcel@univ-tours.fr

ABSTRACT

This paper addresses the long-term problem of defining a subjective interestingness measure for BI exploration. Such a measure involves prior modeling of the belief of the user. The complexity of this problem lies in the impossibility to ask the user about the degree of belief in each element composing their knowledge prior to the writing of a query. To this aim, we propose to automatically infer this user belief based on the user's past interactions over a data cube, the cube schema and other users' past activities. We express the belief under the form of a probability distribution over all the query parts potentially accessible to the user. This distribution is learned using a random walk approach, and more specifically an adapted topic-specific PageRank. The resulting belief provides the foundations for the definition of subjective interestingness measures that can be use to improve the user's experience in their explorations. In the absence of ground truth for user belief, we simulate in our tests different users and their belief distributions with artificial cube explorations and evaluate our proposal based on qualitative evaluation. We finally propose a preliminary usage of our belief estimation in the context of query recommendation.

CCS CONCEPTS

• **Information systems → Relevance assessment**; *Data access methods*; *Environment-specific retrieval*;

KEYWORDS

BI exploration, user belief, PageRank

1 INTRODUCTION

Business intelligence (BI) exploration can be seen as an iterative process that involves expressing and executing queries over multidimensional data (or cubes) and analyzing their results, to ask more focused queries to reach a state of knowledge that allows to answer a business question at hand. This complex task can become tedious, and for this reason, several approaches have been proposed to facilitate the exploration by pre-fetching data [23], detecting interesting navigation paths [25], recommending appropriate queries based on past interactions [1] or by modeling user intents [12].

Ideally, such systems should be able to measure to which extent a query would be interesting for a given user prior to any recommendation. Indeed, as illustrated in [6] and first elicited in [26] in the context of Explorative Data Mining (EDM), the interestingness of a pattern depends on the problem at hand, and, most importantly, on the user that extracts the pattern. An interestingness measure for such explorative tasks should therefore be tailored for a specific user.

Following the idea of subjective interestingness measures initiated and developed by De Bie [4], our aim is to measure the subjective interestingness of a set of queries, based on the prior knowledge that the user has about the data and the cost for the user to understand the query and its evaluation.

It is therefore crucial, before reaching the definition of such an interestingness measure for BI, to be able to transcribe with an appropriate information-theoretic formalism the prior user knowledge, also called belief, on the data. De Bie proposes to represent this belief as a probability distribution over the set of data. However, it is clearly not possible to explicitly ask a user about the degree of belief in each element composing her knowledge prior to each query, let alone identifying on which element of knowledge expressing this probability distribution. This motivates the investigation of approaches for automatically estimating the user's belief based on their implicit feedback. Let us now consider the following example to illustrate the difficulty of estimating probabilities for the belief.

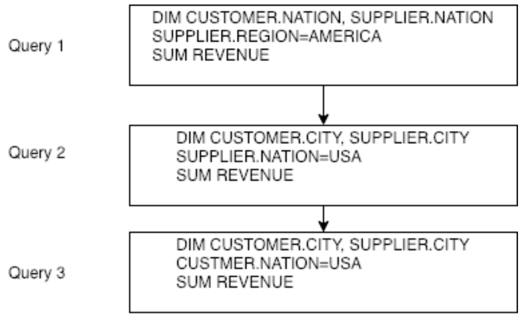

Figure 1: Toy SSB benchmark session

Example. Let us consider the explorative session over the Star Schema Benchmark schema [20] consisting of 3 queries, as illustrated in Figure 1, and loosely inspired by session 3 of the SSB's workload. For the sake of readability, only the relevant query parts (grouping set, filters and measures) are shown. This example showcases the short session initiated by a user that explores the cube looking for information on revenue some company makes in different locations. Assume we are interested in recommending a query to the user for continuing their exploration. This recommendation should be both connected to the most used query parts, so as not to loose focus, but also should bring new, unexpected information, so as not to feed the user with already known or obvious information.

A naive solution would be to use the set of all possible query parts as the set of data and to express the belief based on the frequency of each query part in the past user history. From the session in Figure 1 it is possible to compute the number of occurrences of each query part (for instance, SUM REVENUE appears 3 times, CUSTOMER.CITY 2 times, while SUPPLIER.REGION=AMERICA appears only once, etc.).

However, this simple representation raises major problems: first, the vector of user belief computed from the number of occurrences will mostly contain zero values because the majority of users will concentrate their exploration to a certain region of the data cube. Second, this belief would not give any probability to query parts such as CUSTOMER.NATION=CANADA, while if user knows about *AMERICA* and *USA*, she is likely to have a basic knowledge about sibling countries to *USA* in the dimension *CUSTOMER.NATION*. Finally, it may also be taken advantage of other users' former explorations, as a proxy of what the current user might find interesting.

This example stresses the need for an approach to define the belief based on the users' past activity, as well as an information about how knowledge is structured, which, in the case of the data cube, can be found in the cube schema. We note that while previous works already investigated surprising data in cubes (see e.g., [9, 25]), to the best of our knowledge none of them did so by explicitly modeling a user's belief.

As a first step in this direction, this paper proposes tracking user belief in BI interactions for measuring subjective interestingness of a set of queries executed on a data cube. We propose to build a model of the user's past explorations that will then be used to infer the belief of the user about the query parts being useful for the exploration. However, contrary to the context of pattern mining [4] where in general no metadata information is available, the query parts that we consider in our model cannot be considered agnostically of the cube schema, that the user usually knows. In this context, we propose to take advantage of it to infer what a user may or may not know based on what she already visited and what is accessible from the previous queries. We define the belief of a user as a probability distribution over the set of query parts coming from the log of past activities and the cube schema. We propose to learn this probability distribution with a modified topic-specific PageRank algorithm, where the topology matrix is based on previous usage and the schema of the cube, and where the teleportation matrix corresponds to a specific user model. Finally, in order to evaluate our approach, we will take advantage of the artificial exploration generator CubeLoad [22] that mimics several prototypical user behaviors that should exhibit different types of belief probability distributions. The difficulty of the evaluation in our case lies in the absence of explicit belief ground truth. In this context, two main experiments are reported in this paper: (1) an evaluation based on comparisons of beliefs produced by different user profiles in CubeLoad, and (2) a preliminary study that indicates to which extent it is possible to characterize a recommendation as one of the reference exploratory profile of CubeLoad only based on the belief model of recommended queries. As a side result, this experiment also shows how the belief model is impacted by the history (the log files) on which the recommender system is trained and how some exploration pattern may trap the user in a cognitive bubble.

This paper is organized as follows. Section 2 motivates the use of user belief and subjective interestingness measure in the context of data exploration. Section 3 introduces the concepts used in our approach: a simplified definition of a query part, subjective interestingness and topic-specific PageRank. Section 4 presents our modeling of past usage and database schema as topology and user profile for discovering user beliefs. Finally Sections 5 and 6 present our experiments on CubeLoad generated sessions while Section 7 discusses related work and Section 8 concludes and draws perspectives.

2 USER BELIEF IN DATA EXPLORATION

This section describes how the knowledge of a user belief, and by extension a subjective interestingness measure, could be used to improve the user's experience in the context of interactive data exploration. This example highlights the main scientific challenges of such task, some of them being left as future work as the present paper exclusively focuses on a first expression of user belief in the context of data cube exploration.

In our vision, illustrated in Figure 2, human remain in the loop of data exploration, i.e., the exploration is not done automatically, but we aim at making it less tedious. All users, naive or expert, willing to explore a dataset, express their information need through an exploration assistant. This assistant is left with the task of deriving from the user's need the actual queries to evaluate over the data source. This exploration assistant communicates with a belief processor that is responsible for the maintenance of the user's profile, i.e., a model of that user, in the sense that it includes an estimation of the actual belief unexpressed by the user. This belief is manifold and concerns e.g., hypotheses on the value of the data, the filters to use, how the answer should be presented, etc. The belief processor activates a series of subjective interestingness measures that drives the query generator for deriving and recommending the most interesting queries for this user, in the sense that they produce relevant, unexpected, diverse answers, avoiding undesirable artifacts such as biased or false discoveries, the so-called cognitive bubble trap, etc. These answers and recommendations are packaged (e.g., re-ranked, graphically represented) by the storytelling processor before being displayed to the user and sent to the belief processor for profile updating.

Notably, thanks to the belief processor, once enough diverse users are modeled, the storytelling processor may cope with the cold start problem of generating recommendation for unknown users (the future user of Figure 2).

The work presented in this paper is a preliminary step in the implementation of this vision. We first concentrate on cube exploration, and on expressing the belief in terms of a probability distribution over the query parts the active user may use for the next query of their exploration.

Several improvements, left as future work, are needed before integrating our belief processor into a personal data exploration assistant. A first step is to improve our belief model, either by refining the computation by taking into account more information from the schema and the usage, or to change the scale at which the belief is expressed (for example at the cell's level rather than the query parts). Second, it is crucial to define an efficient

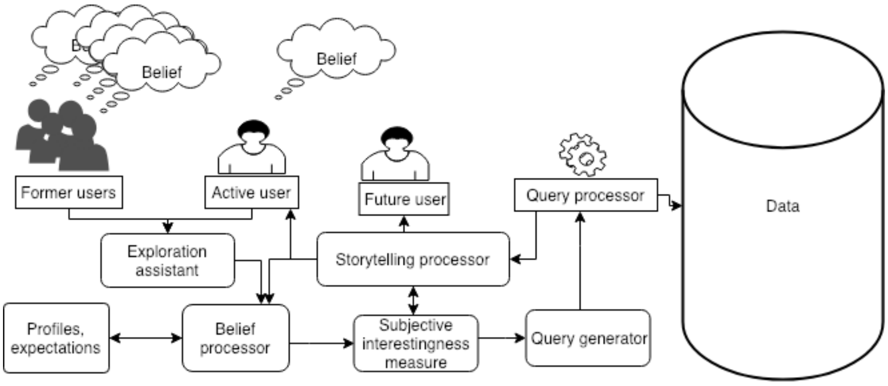

Figure 2: Envisioned use of belief and subjective interestinness measures in data exploration

belief update routine that scales with the size of the distribution probability that we use. Finally, we have to define a set of interestingness measures based on the information content of a sequence of queries and the complexity of these queries. Again, these definitions should take advantage of the peculiarities of the data cube exploration context to be on par with what a human analyst would consider interesting.

3 PRELIMINARIES

3.1 Query parts

Our user belief model relies on query parts. The term query parts can undergo different meanings. Coherent with our objective of taking into account both usage (i.e., previous explorations) and cube schema, our query part definitions encompasses both.

First, concerning the usage, we use the definition of query part provided by [22], where the authors consider it is one constituent of a

> multidimensional query consisting of (i) a group-by (i.e., a set of hierarchy levels on which measure values are grouped); (ii) one or more measures whose values are returned (the aggregation operator used for each measure is defined by the multidimensional schema); and (iii) zero or more selection predicates, each operating on a hierarchy level.

Second, regarding the schema, we consider as query parts all dimensional attributes and possible selection predicates, leaving measures for future work. Indeed, our approach relies on relations between query parts. While it is easy to infer relations based on the schema for dimensional attributes (based on the hierarchy) or selection conditions (based on the value selected), it is not trivial to find a relation based on the measures.

3.2 Interestingness for exploratory data mining

The framework proposed by De Bie [4], in the context of exploratory data mining, is based on the idea that the goal of any exploratory data mining task is to pick patterns that will result in the best updates of the user's knowledge or belief state, while presenting a minimal strain on the user's resources. In De Bie's proposal, the belief is defined for each possible value for the data from the data space and can be approximated by a background distribution.

As a consequence, a general definition for this interestingness measure (IM) is a real-valued function of a background distribution, that represents the belief of a user, and a pattern, that is to say the artifact to be presented to the explorer. Given a set Ω, the data space, and a pattern Ω' a subset of Ω, the belief is the probability $\mathrm{P}(\Omega')$ of the event $x \in \Omega'$, i.e., the degree of belief the user attaches to a pattern characterized by Ω' being present in the data x. In other words, if this probability is small, then the pattern is subjectively surprising for the explorer and thus interesting. In this sense, the IM is subjective in that it depends on the belief of the explorer. De Bie also proposes to weight this surprise by the complexity of the pattern Ω' as follows:

$$IM_{DeBie}(P, \Omega') = \frac{-log(P(\Omega'))}{descComp(\Omega')} \qquad (1)$$

where P represents the user belief, i.e. the background distribution of the pattern Ω' over the set of data x and $descComp(\Omega')$ denotes the descriptional complexity of a pattern Ω'.

The data mining process consists in extracting patterns and presenting first those that are subjectively interesting, and then refining the belief background distribution based the newly observed pattern Ω'.

The key to such modeling as proposed by De Bie lies in the definition of the belief of each user for all possible patterns and how it should evolve based on new patterns explored during time. Section 4 details how we represent the user belief and how we learn it in our context. The update of such belief based on new discovered patterns is left for future work.

3.3 Topic-specific PageRank

Initially, the PageRank algorithm is designed to estimate the relative importance of web pages as a probability to end up on this web page after an infinite surf on the web [8]. A graph whose topology represents the co-occurrence of web pages is defined. The *PR* PageRank vector is the solution to:

$$PR = ((1 - \alpha)M + \alpha T) \times PR \qquad (2)$$

where M is the stochastic transition matrix of the graph of web pages co-occurrences and T represents, in the traditional PR algorithm, a stochastic uniform distribution matrix that ensures the convergence to stable probabilities, and can conceptually be understood as the possibility for the surfer to teleport anywhere on the web graph.

The topic-specific PR [15] proposes to bias the traditional PR algorithm with a stochastic teleportation matrix that restrains the possible teleportations to the vertices assigned to a particular topic. As shown in Equation 2 the parameter α allows to give more or less importance to teleportation. The matrix T can be any stochastic matrix representing a specific topic that will bias the probability distribution. Section 4.2 show how to use this topic-specific PR algorithm to estimate the probabilities of query parts.

4 INFERRING USER BELIEF FROM SCHEMA AND LOG USAGE

This section addresses the following questions: (1) what is user belief in BI exploration? (2) How to estimate it?

4.1 What is user belief in BI?

Ideally, the user belief would be a probability the user attaches to the statement "I believe the value of this cell is exactly this one". Modeling such a belief is one of our long term perspectives. In a first methodological step towards this direction, we consider in this work that the user belief is the importance the user attaches to the statement "I believe this query part is relevant for my exploration". In some sense, we consider query parts as pieces of knowledge about the data that reduce the set of possible values it may take from the original data space, inspired by the De Bie's view of explorative pattern mining [3, 18].

We propose to define the user belief over the set of query parts for two main reasons. First, the set of query parts is measurable and thus respects the formal constraints in the model of De Bie [4] in case we want to extend the belief to an interestingness measure. Second, working at the query level would end-up with a very sparse representation of the data space, as the probability that two queries occur in the same exploration is much lower than the probability that two query parts appear in the same query or exploration. Moreover, when considering query parts, the most interesting ones for the user may appear in several consecutive queries and thus might have more prominent probability values.

As we cannot "brain-dump" the user, the belief is approximated by the importance of the available query parts. The challenge lies in a way to find this probability distribution over a possibly infinite or too large set of query parts even if we restrict to the attributes in a given schema.

Practically, in order to avoid to deal with all these query parts, we restrict to those appearing in a query log and in the schema, where only the active domain of the attributes is considered. This importance attached to query parts appearing in actual users' explorations is consistent with our objective of defining a subjective measure.

4.2 Using PageRank as a belief function

Once restricted the set of query parts that will be considered, we still need to compute their relative importance expressed as a probability distribution for a specific user. This is done by a topic-specific PageRank (PR) that computes the probability for a user u to end up on a query part q when using the cube schema during the exploration, knowing past explorations by other users and knowing the profile of u.

Let M be the topology transition matrix as defined in the topic specific PageRank (TSPR). It is computed from the directed graph of query parts $G = \langle V, A \rangle$ defined over the set V of all query parts found in a log of queries L, and a schema S that represents the topology of the multidimensional space. This schema can be reduced either as a set of sequence of attributes or as a set of hierarchies of attribute values. As motivated in Section 3.1, we ignore the measures in this definition. The graph G is constructed as follows: first we apply schema based construction rules and then log usage construction rules.

Schema based construction rules. (i) for each pair of attributes, there is an arc $a \in A$ if one is the immediate roll-up predecessor or successor of the other in the dimension where they appear and (ii) for each pair of selected values, there is an arc if they are cousins in a dimensional hierarchy (i.e., values taken from the domain of the same attribute). These two rules are repeated when dimensions are combined. Note that, again, more complex relations could be captured in the future in our graph, like sibling relations for selected values in a dimensional hierarchy.

Log usage construction rules. there is an arc a from vertex q_1 to vertex q_2 and an arc a' from q_2 to q_1 if q_1 and q_2 appear together in the same query. There is also an arc a from vertex q_1 to vertex q_2 if q_1 is in a query that precedes another query where q_2 appears.

While the graph G is a general topology of the query space, it is not however subjective in any way. It has to be biased toward a specific user, represented by the subset U of the query parts occurring in their sessions. Let $G_u = \langle V, E_u \rangle$ be the user specific weighted directed graph, defined over the same set V of vertices as G, but only constructed following the specific log usage rules: there is an arc e from vertex q_1 to vertex q_2 (resp. from q_2 to q_1) with weight n if q_1 and q_2 are co-occurring n times in the query log of the user. Similarly, if a query part q_1 is in a query that precedes another query where part q_2 appears, then an arc (q_1, q_2) with a weight 1 is added to the graph or the weight of the existing arc is incremented.

Let P be the transition matrix of G, M be the normalized stochastic version of P, it can be interpreted as a Markov chain.

Let B be the transition matrix of G_u, T is the normalized version of this matrix, also referred to as the teleportation matrix. We can now construct a transition matrix for the underlying Markov model used by the TSPR as follows:

$$TSPR = \alpha T + (1 - \alpha)M \tag{3}$$

The user specific PageRank vector can now be obtained by solving $PR_u = PR_u \times TSPR$. The PR_u vector represents our user belief, as a probability distribution over the set of query parts V, with parameter α ruling the importance attached to the user profile.

5 COHERENCE OF BELIEF WITH REALISTIC PROFILES

Our approach is implemented in Java using `jaxen` to read cube schemas and `Nd4j`[1] for simple and efficient matrix computation. The code is open source and available in a `GitHub` public repository[2].

Our first experiments aim at showing that the belief probability distribution that we learn from our model is coherent with what could be expected in realistic exploration situations. To settle such experiments, we need to have a well formalized environment where possible explorations are already categorized into several prototypical user profiles that could be used to bias our model. To this aim, we use the CubeLoad generator [22] that exhibits 4 main exploratory templates illustrated in Figure 3. Several simulations are conducted to assess that our learned probability distributions

[1] https://deeplearning4j.org/docs/latest/nd4j-overview

[2] https://github.com/AlexChanson/IntrestingnessForOLAP

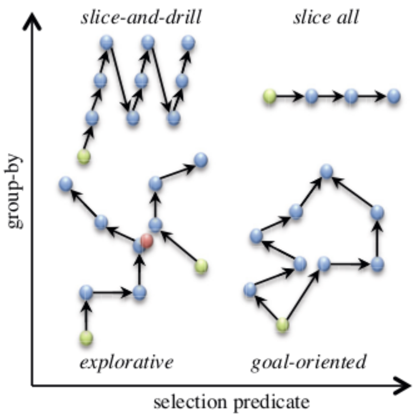

Figure 3: Session templates in CubeLoad (from [22]): , *"seed queries in green, surprising queries in red".*

behave differently and accordingly to what was expected from the CubeLoad templates.

5.1 Experimental protocol

We generated a series of 50 sessions using the Cubeload generator over the schema of a cube constructed using the SSB benchmark [20], that we split in 2 groups: the first 43 sessions are used to construct the topology graph, and the next 7 are taken from a single CubeLoad template, and are used to define the user profile for the teleportation matrix. We run 50 randomized samples to achieve a traditional cross-validation protocol.

We will establish our results around two distinct measures:

- first, we use a distance between two probability distributions to estimate to which extent they are close and behave similarly. A classical choice could have been to use a Kullback-Leibler divergence, but here we prefer to use the discrete Hellinger distance that has the advantage of being symmetric and bounded in the interval $[0, 1]$. The discrete Hellinger distance $H(P, Q)$ compares two discrete probability distributions $P = (p_1, \ldots, p_k)$ and $Q = (q_1, \ldots, q_k)$ as follows:

$$H(P,Q) = \frac{1}{\sqrt{2}}\sqrt{\sum_{i=1}^{k}(\sqrt{p_i} - \sqrt{q_i})^2} \qquad (4)$$

- second, we will plot the average probability distributions for each user model and their standard deviation to appreciate how the distribution evolves and how it is shaped, for a qualitative evaluation.

5.2 Hypothesis

We expect the 4 templates included in CubeLoad to behave differently. The *slice all* template is a local user model that only explores a small fraction of the data space in terms of coverage of different concepts embedded in this space. It is thus expected that when comparing to a distribution probability of the whole topology, it will maximize this distance. In this case, there should also be only a few query parts that concentrate most of the interactions with a higher probability, as the user matrix is very local and is unlikely to teleport to most portions of the data space, which in turns become improbable. Similarly, the probability of knowing something from the data should be very low, which could be traduced by a fast and strong drop in the probability distribution.

On the contrary, the *explorative* template should allow for a broader exploration of the data space. This template should lead to minimizing its distance with a topology based distribution. In this case, it is expected that there are fewer very improbable query parts but that there are more higher probabilities on most query parts, because of the coverage of the data space by the template.

The *goal-oriented* and *slice-and-drill* templates are expected to be intermediate states between the two previous templates. Indeed, both models explore the data space more than *slice-all*, but are a bit more constrained than *explorative*.

5.3 Results

Table 1 represents the distance between:

- the distribution of probabilities computed only from the topology denoted PR, that is to say the distribution probability that a traditional PageRank would have produced in our context,
- and the distribution produced when the PageRank is biased toward a specific user profile, denoted TSPR here, corresponding to a specific template in CubeLoad.

We can first observe in Table 1 that the distance between the resulting distributions is proportional to α as expected. Indeed, if α is very low, the biased distribution is very close to the PR topology distribution. The higher α, the more characteristics from the user profile are introduced in the transition matrix. Second and as expected, we notice that the *slice-all* profile bears the larger distance with the topology as it only explores a small portion of the possible space, while the other profiles seem to have a comparable behavior in terms of distance.

Figures 4 and 5 plot, for two distinct values of parameter α, the average distribution of probabilities for the 4 user profiles and the PR distribution corresponding to the topology. As expected, when $\alpha = 0.2$ all distributions heavily tend to mimic the PR distribution. On the contrary, when $\alpha = 0.8$ the difference brought by the user profile become clearly visible. The *slice-all* profile tends to have a higher number of small probabilities as expected and then decreases very strongly as there is only a few query parts which are likely to be known by the user. *Explorative* user profile favors exploration and thus it is less likely that the probability of knowing already a query part is low, and consequently the distribution of value tends to decrease more smoothly in this case. Finally, and as for our Hellinger distance test, *goal-oriented* and *slice-and-drill* profiles exhibit intermediate behavior for low-probability query parts and then tends to evolve as smoothly as the *explorative* profile.

6 HOW USAGE AND RECOMMENDATION IMPACT BELIEF

The first experiments reported in Section 5 show that our model of belief is able to transcribe different exploration templates based on the shape of their probability distribution functions, although 3 templates, namely *Explorative*, *Goal Oriented* and *Slice And Drill*, remain very close in this respect.

However, if our belief model is to be faithful to the user intent, as it is expressed at the query part level, any change in the accessed area of a cube should be traduced by a change in the assigned probability weights in our belief model. We propose to first simulate this diversity in the cube exploration by learning

User/α	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Explorative	0,021	0,042	0,063	0,084	0,106	0,130	0,155	0,183	0,215
Goal Oriented	0,015	0,031	0,047	0,063	0,081	0,101	0,123	0,150	0,182
Slice All	0,073	0,127	0,170	0,209	0,244	0,279	0,315	0,350	0,392
Slice and Drill	0,022	0,044	0,066	0,089	0,114	0,139	0,167	0,200	0,236

Table 1: Hellinger distance between PR and our biased PR with several user profiles following user templates in CubeLoad generator.

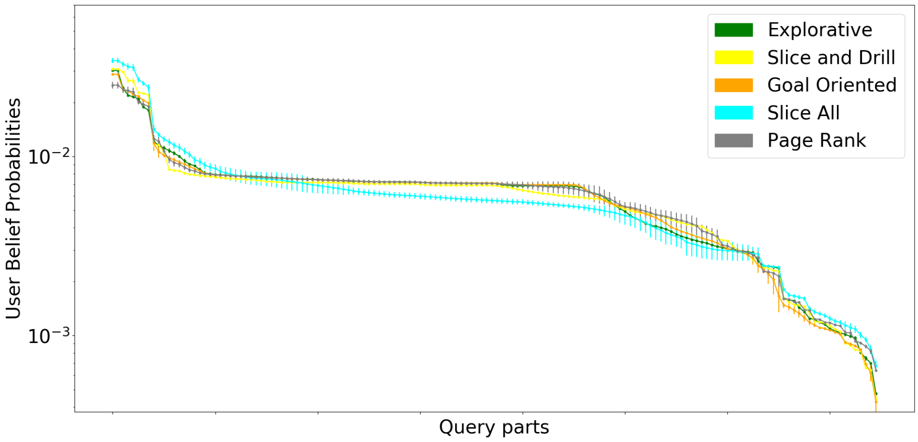

Figure 4: Distribution of probabilities computed by our model for all 4 user profiles when $\alpha = 0.2$ (log scale).

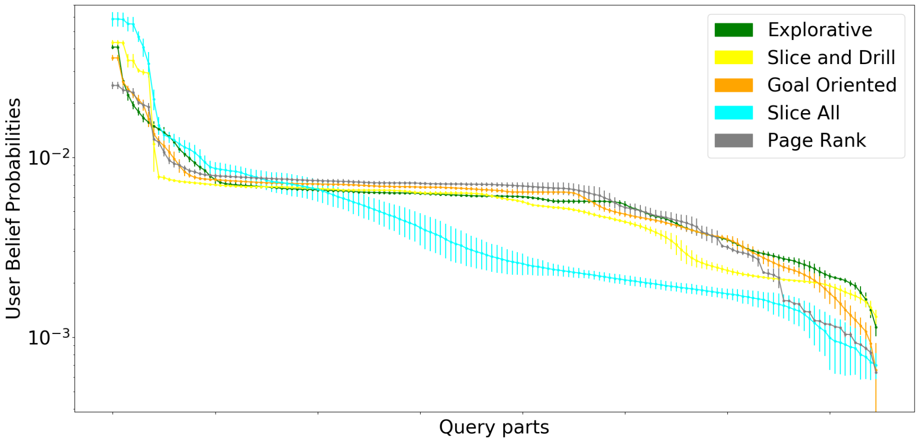

Figure 5: Distribution of probabilities computed by our model for all 4 user profiles when $\alpha = 0.8$ (log scale).

the belief model on different and independent log files generated by CubeLoad.

Second, as the long-term objective is to use belief-based interestingness measure as an input for tailoring personalized recommendations, we want to evaluate to which extent the output of a recommender system could impact the belief of a specific user. To this aim, we use a state-of-the-art OLAP recommender system [1] that is trained on a log file representative of a specific CubeLoad template and whose output recommendation is used to define a new belief distribution. It is important to notice that, at this step, we consider that all recommended queries are executed at the same time and that the belief can be computed based on the same prior knowledge from the user for all recommended queries at once. As explained earlier, dealing with the evolution of the user belief after each query is out of the scope of this paper.

In both cases, we need to define beforehand a *reference belief* for each of the 4 CubeLoad templates in our simulations. Then a divergence measure can traditionally be used to evaluate to which extent changing the template or changing the log file on which we learn the models impacts the belief distribution.

6.1 Experimental protocol

Evaluation scenarios. Figure 6 presents the main evaluation protocol for the impact of usage and recommender system on the belief measure. The left-side of Figure 6 illustrates how to generate the *reference belief*. Similarly to Section 5.1, we use 50 CubeLoad sessions among which 43 are used to compute the topology graph of our PageRank (see *"Log usage construction rules"* in Section 4.2) and 7 sessions issued from a single CubeLoad template are used to simulate a user and bias the topic-specific PageRank.

The right-side of Figure 6 describes how we generate the *test belief* based on the recommender system [1] and a simulated user of the recommender system. To do so, we generate a new *independent* log file from which 10 random sessions from a single CubeLoad template are drawn. They are used to train the recommender system. In addition a separate session is drawn and truncated to be used as a seed for the recommender. In order to evaluate to which extent the diversity of queries affects our belief model, we also consider the case where the recommender system as well as the seed session for the recommender are drawn at random from the same log file as the one used for the reference belief. We call this scenario *identical* while the first one with distinct session files is named *independent* hereafter.

Evaluation of results. We use the Hellinger distance as described in Section 5.1 to compare the reference and test belief distributions. We generate 40 random logs (10 per profile) and we run them against randomly chosen seeds from the 4 profiles and against the 4 possible reference belief models. Results are averaged and reported in Table 2 and 3.

6.2 Hypothesis

We expect several observable results from these experiments. First, it is expected that the belief divergence between the recommended queries from the *test* and the queries used as a *reference* is much larger in case of *independent* log files than in case of *identical* log files. As in both cases the recommender system is involved, this would mean that any observed difference would be due to the difference in log files and that our belief model while preserving the distribution probabilities of each CubeLoad template is able to reflect the differences in usage on query parts.

Second, we expect to observe differences between the 4 CubeLoad templates that could refine the observations of Section 5.3. Indeed, only based on the distribution it was not possible in our setting to distinguish clearly among the *Explorative*, *Goal Oriented* and *Slice and Drill* templates. We expect them to remain close but with slight divergence that would reflect the different abilities of each template to dig into new parts of the data cube. However, similarly to experiments in Section 5.3 we expect *Slice All* to be significantly divergent from the other profiles. Indeed, it is known that this template exhibits tendencies to explore locally the cube by only navigating a dimension following siblings relations.

Finally, if we consider our belief model as representative of the usage, we may be able to draw conclusions about the use of recommender systems with CubeLoad templates and to which extent the latter propose really distinguishable exploration patterns. To this aim, a comparison of divergence values when comparing, for each CubeLoad template, its reference to its recommended test on the same log file would indicate to which extent the template explores various portion of the cube. In other words, the more explorative a template is, the larger the divergence between a recommendation based on this template and its reference should be.

6.3 Results

Tables 2 and 3 present experimental values of Hellinger distance for the *identical* and the *independent* scenarios and each type of recommendation based on the 4 CubeLoad templates (lines) for each reference belief model (columns).

It can be seen that, as expected, the divergence is significantly lower when considering recommendations based on the same log file that was used for the reference belief computation. This shows that our belief method is sensitive to the actual queries that are involved in the building of the model despite the general trends observed for the probability distribution in Section 5.3.

Tests\References	*Explorative*	*Goal Oriented*	*Slice All*	*Slice and Drill*
Explorative	**0.64**	0.60	0.47	0.60
Goal Oriented	0.64	**0.61**	0.46	0.60
Slice All	0.67	0.63	**0.47**	0.62
Slice and Drill	0.63	0.60	0.43	**0.58**

Table 2: Average Hellinger distance values on 10 runs when log files are identical. Lines represent test and columns represents reference belief.

Tests\References	*Explorative*	*Goal Oriented*	*Slice All*	*Slice and Drill*
Explorative	0.85	0.86	0.81	0.86
Goal Oriented	0.85	0.85	0.81	0.86
Slice All	0.83	0.84	0.79	0.84
Slice and Drill	0.86	0.86	0.82	0.87

Table 3: Average Hellinger distance values on 10 runs when log files are independent. Lines represent test and columns represents reference belief.

Then, results from Tables 2 and 3 corroborates that there exists differences between the CubeLoad templates but that are more or less difficult to observe depending on the CubeLoad template, as shown by the distribution of Section 5.3.

In the identical scenario, *Explorative* recommendation is the most distant to its corresponding *Explorative* reference (distance = 0.64) and the most distant to all other recommendations as shown in column *Explorative* from Table 2. This can be understood by the nature of the CubeLoad template which makes it difficult to model for a recommender system, which in turns implies a higher probability to access new portions of the cube and thus new query parts.

Goal oriented column of Tables 2 and 3 presents not significantly different distance values whatever the recommendation profile is. This is expected as this template can generate several different paths in the cube to access the same region in the end. As a consequence, there is a variety in the accessed query parts which in turns tends to smooth the distance values.

Slice and Drill exhibits significant differences and with a small distance to itself of 0.58, which means that the recommendation that is produced reflects the CubeLoad template and stays in a localized region of the data cube with more common query parts between recommendations and the reference template.

Finally, an interesting observation is that *Slice All* has a very low inner distance of 0.47 in Table 2 which means that the exploration is very focused and that even a recommender system based on this template will reproduce this local exploration behaviour with no surprising query parts. This reflects a sort of cognitive bubble associated to this template, which can be detected with our approach of user belief estimation.

The latter is confirmed by the analysis of the line *Slice All* in Table 2 where the distance between the recommended *Slice All* and the other CubeLoad reference templates that are higher than

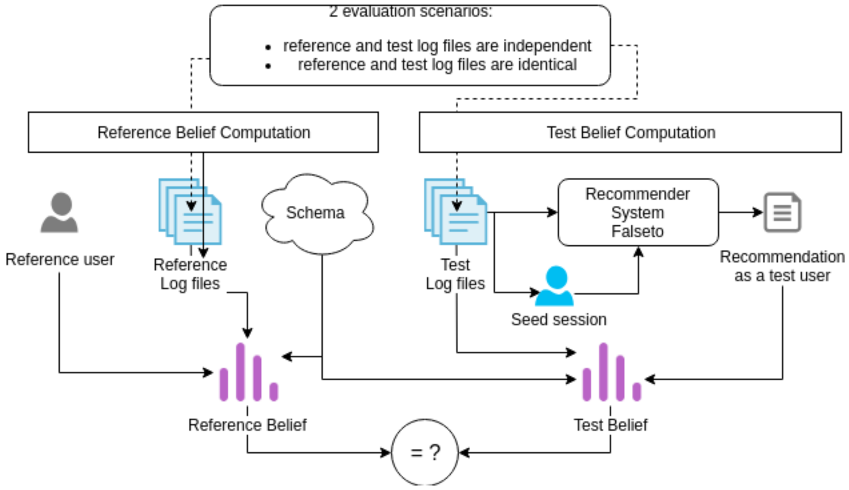
2 evaluation scenarios:
reference and test log files are independent
reference and test log files are identical
Reference Belief Computation
Test Belief Computation
Reference user
Reference Log files
Schema
Test Log files
Recommender System Falseto
Recommendation as a test user
Seed session
Reference Belief
Test Belief
= ?

Figure 6: Evaluation protocol for the impact of usage and recommender system on the belief measure.

the other distances for each of these templates. For example, in Table 2, the distance of *Slice All* to *Explorative* is 0.67 which is the largest distance for the column *Explorative*.

These first results bring new insights on our belief model, and to which extent it is coherent with the CubeLoad profiles. A more thoroughful analysis of these results is needed to conclude about differences on CubeLoad templates and is left as future work.

7 RELATED WORK

Our work deals with subjective interestingness and how to define such a measure by learning a belief distribution from users' past activities in the context of BI. This section presents some interestingness measures, and how they have been used in the context of recommendation.

Interestingness has attracted researchers since a long time in the context of data mining. Indeed, there exists numerous tasks, for example in pattern mining, for which it is critical to be able to filter out uninteresting patterns such as item sets or redundant rules, to control the complexity of the mining approaches and increase their usability.

In [7, 14], the authors identify two main types of interestingness measures. Objective measures are based only on data and corresponds to quality metrics such as generality, reliability, peculiarity, diversity and conciseness, or directly measurable evaluation metrics such as support confidence, lift or chi-squared measures in the case of association rules [2].

On the contrary, subjective measures consider both the data and the user and characterize the patterns' surprise and novelty when compared to previous user knowledge or expected data distribution. The first work on the topic of subjective interestingness is certainly [26] that is restricted to the pattern mining domain. In [4, 6], the author extends this notion to any explorative data mining task and represents interestingness as a ratio between information content and complexity of a discovered pattern being it an itemset, a cluster or a query evaluation result (see Section3.2 for more formal details). In [6], De Bie defines the subjective interestingness as a situation where a

> "user states expectations or beliefs formalized as a 'background distribution'. Any 'pattern' that contrasts with this and is easy to describe is subjectively interesting".

The authors in [14] consider also semantic measures of interestingness, based on the semantics and explanations of the patterns like utility and actionability. This latter property of actionability is not meaningful in our case where, as stated by De Bie [6], we consider situations

> "where the user is interested in exploring without a clear anticipation of what to expect or what to do with the patterns found".

Recently, in [21] the authors propose a data exploration study based on De Bie's FORSIED framework [5, 6] that pairs a high level conjunctive query language to identify groups of data instances and expresses belief on some real-valued target attributes, based on location and spread patterns. This work is close to our proposal but expresses belief on a summary of the data.

In the context of data cube exploration, to the best of our knowledge there is no final and consensual interestingness measure or belief distribution elicitation method, while there exists measures that are closely related. Measures have been defined as unexpectedness of skewness in navigation rules and navigation paths [19] and computed as a peculiarity measure of asymmetry in data distribution [17]. In [13], the authors define interestingness measures in a data cube as a difference between expected and observed probability for each attribute-value pair and the the degree of correlation among two attributes. In [24], Sarawagi describes a method that profiles the exploration of a user, uses the Maximum Entropy principle and the Kullback-Leibler divergence as a subjective interestingness measure to recommend which unvisited parts of the cube can be the most surprising in a subsequent query.

In [10, 11] the authors use supervised classification techniques to learn two interestingeness measures for OLAP queries: (1) focus, that indicates to what extent a query is well detailed and connected to other queries in the current exploration and (2) contribution that indicates to what extent a query contributes to the interest and quality of the exploration.

Finally, interestingness and related principles have been studied in the context of recommendation but more widely used for evaluation rather than the recommendation itself [16]. Interestingness is reflected based on 4 main criteria such as diversity, serendipity, novelty, and coverage, in addition to traditional accuracy measures.

In the context of OLAP query recommendation, several recommendation algorithms have been proposed that take into account the past history of queries of a user either based on a Markov model [23] or on extracted patterns [1]. Noticeably, [1] quantifies how distant is the recommendation from the current point of exploration to evaluate the interestingness of each candidate query recommendation.

8 CONCLUSION

This paper describes a first attempt to model user belief as a probability distribution over query parts in the context of data cube exploration. The experiments conducted on several prototypical user templates generated with CubeLoad illustrate how a topic-specific PageRank can be used to approximate such probability distribution. Preliminary experiments show that it is already possible to use our belief model as an indicator for the type of exploration that a user favors (more global or more local exploration) or for the portion of the data cube that will be most certainly explored. Finally our belief model can be used to identify recommendations that are likely to trap user in a cognitive bubble and thus may help leveraging diversity in exploration.

This work opens up for further research avenues. Our long-term goal is the implementation and validation through user studies of the vision illustrated by Figure 2. Shorter-term research questions include: (1) how to refine our model of query parts, for example to better take into account measures in the schema? (2) How to express the user's belief beyond query parts, for instance over the cube's cells? (3) How to improve our model to better distinguish between less marked exploration patterns, for instance between the *goal-oriented* and *slice-and-drill* patterns, in terms of distribution? (4) How to deal with real noisy log usage? (5) How can we update the probability distribution when a new query is executed? (6) Finally, what interestingness measures can be devised from on our proposal?

REFERENCES

[1] Julien Aligon, Enrico Gallinucci, Matteo Golfarelli, Patrick Marcel, and Stefano Rizzi. 2015. A collaborative filtering approach for recommending OLAP sessions. *DSS* 69 (2015), 20–30.

[2] S. Alvarez. 2003. *Chi-squared computation for association rules: preliminary results*. Technical Report BC-CS-2003-01. Computer Science Dept. Boston College, Chestnut Hill, MA 02467 USA. 11 pages. http://www.cs.bc.edu/~alvarez/ChiSquare/chi2tr.pdf

[3] Tijl De Bie. 2011. An information theoretic framework for data mining. In *KDD*. ACM, 564–572.

[4] Tijl De Bie. 2013. Subjective Interestingness in Exploratory Data Mining. In *Advances in Intelligent Data Analysis XII - 12th International Symposium, IDA 2013, London, UK, October 17-19, 2013. Proceedings*. 19–31. https://doi.org/10.1007/978-3-642-41398-8_3

[5] Tijl De Bie. 2014 (accessed on December 2018). *The Science of Finding Interesting Patterns in Data*. http://www.interesting-patterns.net/forsied/

[6] Tijl De Bie. 2018. An information-theoretic framework for data exploration. From Itemsets to embeddings, from interestingness to privacy. In *Keynote presentation given at IDEA'18 @ the KDD'18 conference*. http://www.interesting-patterns.net/forsied/keynote-presentation-given-at-idea18-the-kdd18-conference/

[7] Tom Brijs, Koen Vanhoof, and Geert Wets. 2004. Defining Interestingness for Association Rules. *International Journal "Information Theories Applications"* 10 (2004), 370–375.

[8] Sergey Brin and Lawrence Page. 2012. Reprint of: The anatomy of a large-scale hypertextual web search engine. *Computer Networks* 56, 18 (2012), 3825–3833. https://doi.org/10.1016/j.comnet.2012.10.007

[9] Véronique Cariou, Jérôme Cubillé, Christian Derquenne, Sabine Goutier, Françoise Guisnel, and Henri Klajnmic. 2009. Embedded indicators to facilitate the exploration of a data cube. *IJBIDM* 4, 3/4 (2009), 329–349. https://doi.org/10.1504/IJBIDM.2009.029083

[10] Mahfoud Djedaini, Krista Drushku, Nicolas Labroche, Patrick Marcel, Verónika Peralta, and Willeme Verdeau. 2019. Automatic Assessment of Interactive OLAP Explorations. *To appear in Information Systems* (2019).

[11] Mahfoud Djedaini, Nicolas Labroche, Patrick Marcel, and Verónika Peralta. 2017. Detecting User Focus in OLAP Analyses. In *ADBIS*. 105–119.

[12] Krista Drushku, Julien Aligon, Nicolas Labroche, Patrick Marcel, Verónika Peralta, and Bruno Dumant. 2017. User Interests Clustering in Business Intelligence Interactions. In *Advanced Information Systems Engineering - 29th International Conference, CAiSE 2017, Essen, Germany, June 12-16, 2017, Proceedings*. 144–158.

[13] Carem C. Fabris and Alex Alves Freitas. 2001. Incorporating Deviation-Detection Functionality into the OLAP Paradigm. In *XVI Simpósio Brasileiro de Banco de Dados, 1-3 Outubro 2001, Rio de Janeiro, Brasil, Anais/Proceedings*. 274–285.

[14] Liqiang Geng and Howard J. Hamilton. 2006. Interestingness measures for data mining: A survey. *ACM Comput. Surv.* 38, 3 (2006), 9.

[15] Taher H. Haveliwala. 2003. Topic-Sensitive PageRank: A Context-Sensitive Ranking Algorithm for Web Search. *IEEE Trans. Knowl. Data Eng.* 15, 4 (2003), 784–796. https://doi.org/10.1109/TKDE.2003.1208999

[16] Marius Kaminskas and Derek Bridge. 2017. Diversity, Serendipity, Novelty, and Coverage: A Survey and Empirical Analysis of Beyond-Accuracy Objectives in Recommender Systems. *TiiS* 7, 1 (2017), 2:1–2:42.

[17] M. Klemettinen, H. Mannila, and H. Toivonen. 1999. Interactive exploration of interesting findings in the Telecommunication Network Alarm Sequence Analyzer (TASA). *Information and Software Technology* 41, 9 (1999), 557 – 567.

[18] Kleanthis-Nikolaos Kontonasios and Tijl De Bie. 2015. Subjectively interesting alternative clusterings. *Machine Learning* 98, 1-2 (2015), 31–56. https://doi.org/10.1007/s10994-013-5333-z

[19] Navin Kumar, Aryya Gangopadhyay, Sanjay Bapna, George Karabatis, and Zhiyuan Chen. 2008. Measuring interestingness of discovered skewed patterns in data cubes. *Decision Support Systems* 46, 1 (2008), 429 – 439.

[20] Patrick E. O'Neil, Elizabeth J. O'Neil, Xuedong Chen, and Stephen Revilak. 2009. The Star Schema Benchmark and Augmented Fact Table Indexing. In *Performance Evaluation and Benchmarking, First TPC Technology Conference, TPCTC 2009, Lyon, France, August 24-28, 2009, Revised Selected Papers*. 237–252. https://doi.org/10.1007/978-3-642-10424-4_17

[21] Kai Puolamäki, Emilia Oikarinen, Bo Kang, Jefrey Lijffijt, and Tijl De Bie. 2018. Interactive Visual Data Exploration with Subjective Feedback: An Information-Theoretic Approach. In *34th IEEE International Conference on Data Engineering, ICDE 2018, Paris, France, April 16-19, 2018*. 1208–1211. https://doi.org/10.1109/ICDE.2018.00112

[22] Stefano Rizzi and Enrico Gallinucci. 2014. CubeLoad: A Parametric Generator of Realistic OLAP Workloads. In *Advanced Information Systems Engineering - 26th International Conference, CAiSE 2014, Thessaloniki, Greece, June 16-20, 2014. Proceedings*. 610–624. https://doi.org/10.1007/978-3-319-07881-6_41

[23] Carsten Sapia. 2000. PROMISE: Predicting Query Behavior to Enable Predictive Caching Strategies for OLAP Systems. In *DaWaK*. 224–233.

[24] Sunita Sarawagi. 2000. User-Adaptive Exploration of Multidimensional Data. In *VLDB*. Morgan Kaufmann, 307–316.

[25] Sunita Sarawagi. 2001. User-cognizant multidimensional analysis. *VLDB J.* 10, 2-3 (2001), 224–239. https://doi.org/10.1007/s007780100046

[26] Abraham Silberschatz and Alexander Tuzhilin. 1995. On Subjective Measures of Interestingness in Knowledge Discovery. In *Proceedings of the First International Conference on Knowledge Discovery and Data Mining (KDD-95), Montreal, Canada, August 20-21, 1995*. 275–281. http://www.aaai.org/Library/KDD/1995/kdd95-032.php