A Signaling Game Approach to Databases Querying and Interaction

Arash Termehchy
Oregon State University
termehca@oregonstate.edu

Behrouz Touri
University of Colorado, Boulder
behrouz.touri@colorado.edu

Abstract

As most database users cannot precisely express their information needs, it is challenging for database querying and exploration interfaces to understand them. We propose a novel formal framework for representing and understanding information needs in database querying and exploration. Our framework considers querying as a collaboration between the user and the database system to establish a *mutual language* for representing information needs. We formalize this collaboration as a signaling game, where each mutual language is an equilibrium for the game. A query interface is more effective if it establishes a less ambiguous mutual language faster. We discuss some equilibria, strategies, and the convergence rates in this game. In particular, we propose a reinforcement learning mechanism and analyze it within our framework. We prove that this adaptation mechanism for the query interface improves the effectiveness of answering queries stochastically speaking, and converges almost surely [1].

1 Introduction

Most users do not know the structure and/or content of their databases. Hence, they cannot express their information needs according to how the databases represent their desired information [4, 8, 5, 3]. Hence, it is challenging for database query interfaces to understand and satisfy users' information needs. Database query interface may improve its understanding of how users express their intents through further interaction with users. Similarly, users may gain a better understanding of how the database represent information by submitting queries and exploring their results. For example, to find the answers for a particular information need, the user may submit some initial and ill-specified SQL query, observes its results, and reformulates it according to her observations.

Ideally, we would like the user and the database query interface to develop gradually a mutual understanding over the course of several search tasks and interactions: the query interface should better understand how the user expresses her intent and the user may get more familiar with how the database represents information in the domain of interest. For example, consider a relational database with relation $restaurant$ with many attributes including the price level and number of stars for restaurants in the US. Because the user is not aware of the existence *price-level* attribute and/or its values, she may not put any restriction on this attribute in her queries. Given the user's feedback on the results of several information needs, database query interface may infer that the user is more likely to search for high or medium level price restaurants, less likely to seek very high and low level price restaurants, and almost never to ask for restaurant with very low price level. On the other hand, after working with the database for some time and observing the information about her preferred restaurants, user may learn that she can express her preference for nice restaurants using the attribute *number-of-stars*.

[1]This paper has been presented at SIGIR Conference on Theory of Information Retrieval, ICTIR'05, 2015

Intuitively, both user and query interface may leverage their experiences from the current and the past search tasks to reach a precise mutual understanding fast so they can communicate more effectively. Hence, it is interesting to find the strategies that the query interface and/or the user should follow to achieve a common rapport quickly. Further, one should also explore the properties of this common rapport, e.g. is a perfect and near perfect rapport possible?

To the best of our knowledge, there is not any investigation on the emergence and/or maintenance of such a common representation of information needs between the user and database query interface. In this paper, we propose a novel formal framework that explores the representation of information needs in database querying. It considers the user and query interface as active and potentially rational agents whose goal is to develop a common representation of information needs by communicating certain signals. We formalize this setting as a *signaling game* [6, 13]. We propose a reinforcement learning rule to update the database strategies. We prove that this learning strategy for the query interface improves the effectiveness of answering queries stochastically speaking, and converges almost surely.

2 Related Work

Researchers have proposed querying and exploration interfaces over structured and semi-structured databases that help users to express their information needs and find reasonably accurate results [8, 5, 3]. We extend this body of work by considering users as active and potentially rational agents whose decisions and strategies impact the effectiveness of database exploration. To the best of our knowledge, there is not any game-theoretic approach to database querying and exploration. We also formally explore the information needs representations and the emergence of common representations in database querying and exploration.

Researchers have recently applied game theoretic approaches to model the actions taken by users and document retrieval systems in a single session [12]. In particular, they propose a framework to find out whether the user likes to continue exploring the current topic or move to another topic. We, however, seek a deeper understanding of information need representations and the emergence of common representations between the user and query interface. Further, we investigate the querying and interactions that may span over multiple queries and sessions. A reasonably precise mutual understanding between the user and query interface also improves the effectiveness of ad-hoc and session querying. Moreover, we analyze some equilibria and convergence rates of reaching them for some strategies in the game. Finally, we focus on structured rather than unstructured data.

Researchers in other scientific disciplines, such as economics and sociology, have used signaling games to formally model and explore communications between multiple (rational) agents [11, 6]. Avestani et al. have used signaling games to create a shared lexicon between multiple autonomous systems [2]. We, however, focus on modeling users' information needs and emergence of mutual language between users and database query interfaces. In particular, query interfaces and user may update their information about the interaction in different time scales. We also provide a rigorous analysis of the success of our proposed strategy. Researchers have proposed methods to discover complex search tasks that involve multiple sessions [10, 1, 14]. We focus on establishing a common understanding between the user and query interface in the long run and over a rather large number of possibly simple search tasks.

3 Framework

3.1 Basic Definitions

To simplify our model and proofs, we consider a database to be a single relational table. Our results extend for databases with multiple relations. A *database* DB is a finite relation of a fixed arity. Each member of database DB is called an *entity* and shown as $(v_1, \dots, v_k)$ where $v_1, \dots, v_k$ are values in DB. Each entity in DB represents information about a single named-entity. To simplify our model, we assume that all values in DB are of type string. A query s over DB is an expression $(u_1, \dots, u_k)$ where u_i is either a value in DB or a variable. The answer of query $s = (u_1, \dots, u_k)$ over DB is the set of tuples $(v_1, \dots, v_k)$ in DB such that if u_i is a value, v_i is lexicographically equal to u_i.

To simplify our model, we assume that users like to find a single entity in the database. Users may submit under-specified queries whose answers contain a lot of non-relevant entities. The query interface may use a ranking function to return top-k entities for the query [4]. Clearly, the set of all under-specified queries over a database is finite. Let $S = \{s_i, 1 \leq i \leq n\}$ and $I = \{e_i, 1 \leq j \leq m\}$ be the set of all possible queries and entities, i.e. intents, respectively. We assume that the set of intents have a prior probability π, i.e. $\pi \in \mathbb{R}^m$, $\pi_i \geq 0$ for all i, and $\sum_{i=1}^{m} \pi_i = 1$. In this case π_i is the probability that the user has intent e_i in mind. Without loss of generality we can assume that $\pi_i > 0$ for all $i \in [m]$, otherwise, we can restrict our analysis on the set $\tilde{I} = \{e_i \in I \mid \pi_i > 0\}$.

Throughout this paper, we use the following mathematical notations: For a positive integer $m \geq 1$, we denote $[m] := \{1, \ldots, m\}$. For a vector $u \in \mathbb{R}^m$, we denote the ith entry of u by u_i. Similarly, we denote the (i, j)th entry of an $m \times n$ matrix ($i \in [m], j \in [n]$) by A_{ij}. We also say that A is a row-stochastic matrix (or simply a stochastic matrix) if it is non-negative (i.e. $A_{ij} \geq 0$ for all i, j) and $\sum_{j=1}^{n} A_{ij} = 1$ for all $i \in [m]$. We denote the set of all $m \times n$ stochastic matrices by $\mathcal{L}_{mn}$. For an event A of a probability space Ω, we use 1_A for the indicator function on the set A, i.e. $1_A(\omega) = 1$ of $\omega \in A$ and $1_A(\omega) = 0$ if $\omega \notin A$.

3.2 A Signaling Game Model

First, let us discuss how we model the interaction between the user and the database as a game. The intent of each user in an interaction is an entity e_i in the database. However, the user does not necessarily know how to formulate the right query for this intent and hence, uses an over- or under-specified query, i.e. signal, s_j to convey her intent. On the other hand, the database is not aware of the intent of the user and it can only base its decision on the signal observed from the user. Therefore, the strategy of the database is also a random mapping from the set of signals to the set of intents to map s_j to an intent e_ℓ. Such an interaction is successful if $e_\ell = e_i$, in other words the database has been successful in decoding the user's signal s_j. Otherwise, the interaction is unsuccessful.

This signaling can be modeled as a signaling game with with identical interests played between the user and the database. In this game, the set of strategies of a user is the set $\mathcal{L}_{mn}$ of row-stochastic matrices and the set of strategies of the database is the set $\mathcal{L}_{nm}$ of $n \times m$ row-stochastic matrices. The payoff of the user and the database in this case are

$$u_1(P, Q) = u_2(P, Q) = \sum_{i=1}^{m} \pi_i \sum_{j=1}^{n} P_{ij} Q_{ji}, \tag{1}$$

where u_1 is the payoff of the user and u_2 is the payoff of the database. The payoff function (1) is an expected payoff of the interaction between the user and the database when the user maps the (random) intent e_i to a signal s_j with probability P_{ij} and the database maps back the signal s_j to e_ℓ with probability $Q_{j\ell}$. The larger the value of this payoff function is, the more likely it is that the query interface returns the desired answers to more users' queries. This payoff structure reflects the widely used performance metric for effective database querying. This setting is very similar to the setting of language games which have been studied extensively in the past [13]. The major difference here is the existence of a none-uniform prior over the intents.

A Nash equilibrium for a game is a combination of strategies where the database system (user) will not do better by unilaterally deviating from its strategy, i.e., $u_1(P, Q) \geq u_1(P, Q')$ for all Q'. A strict Nash equilibrium for a game is a Nash equilibrium in which the database system (user) will do worse by changing its equilibrium strategy, i.e., $u_1(P, Q) > u_1(P, Q')$ for all $Q' \neq Q$. Clearly, the players would like the collaborations to result in equilibria with maximum payoff. The database system and user may adapt some rules and algorithms that update their strategies based on the observed signals to improve their payoff and reach a desired equilibrium. These rules may depend on their degrees of rationalities and/or amount of available resources [11, 6]. Further, the players prefer the collaboration to result in a desired equilibrium in smaller number of interactions.

4 An Adaptation Mechanism

We consider the case that the user is not adapting to the signaling scheme of the database. In many relevant applications, the user's learning is happening in a much slower time-scale compared to the learning of the database. So, one can assume that the user's strategy is fixed compared to the time-scale of the database adaptation. When dealing with the game introduced in Section 3, many questions arise:

i. How can a database learn or adapt to a user's signaling scheme?

ii. Mathematically, is a given learning rule effective?

iii. What would be the limiting behavior of a given learning rule?

Here, we address the first and the second questions above. Dealing with the third question is far beyond the page limits of this short paper. As in [9], we consider Roth-Erev reinforcement learning mechanism for adaptation of the database adaption. For the case that both the database and the user adapt their strategies, one can use the results in [9]. Let us discuss the database adaptation rule. The learning/adaptation rule happens over discrete time $t = 0, 1, 2, 3, \ldots$ instances where t denoted the tth interaction of the user and the database. We refer to t simply as the iteration of the learning rule. With this, the reinforcement learning mechanism for the database adaptation is as follows:

a. Let $Q(1) = R$ be the initial database strategy with $Q_{j\ell}(0) > 0$ for all $j \in [n]$ and $\ell \in [m]$.

b. For iterations $t = 1, 2, \ldots$, do

   i. If the user's signal at time t is $s(t)$, return a list $E \subseteq I$ of the entities whose cardinality is k with probability:

$$P(E(t) = \{i_1, \ldots, i_k\} \mid s(t)) = Q_{s(t)i_1}(t) \cdots Q_{s(t)i_k}(t).$$

   ii. If the user is satisfied with the list, set

$$R_{ji} = \begin{cases} R_{ji} + 1 & \text{if } j = s(t) \text{ and } j \in E(t) \\ R_{ji} & \text{otherwise} \end{cases}. \tag{2}$$

   iii. Update the database strategy by

$$Q_{ji}(t+1) = \frac{R_{ji}}{\sum_{\ell=1}^{m} R_{j\ell}}, \tag{3}$$

   for all $j \in [n]$ and $i \in [m]$.

In the above scheme R is simply the reward matrix. When the decoding of the signal $s(t)$ using the chosen list $E(t)$ is successful, the database *reinforces* all the pairs $R_{s(t)i}$ for $i \in E(t)$.

Few comments are in order regarding the above adaptation rule:

- One can use available ranking functions, e.g. [4], for the initial conditions $R(1) = Q(1)$ which possibly leads to an intuitive initial point for the learning dynamics. One may normalize and convert the scores returned by these functions to probability values.
- In step b.ii., if the database have the knowledge of the user's intent after the interactions (e.g. through a click), the database set $R_{ji} + 1$ for the known intent e_i. The mathematical analysis of the both cases will be similar.
- In the initial step, as the query interface uses a ranking function to compute the probabilities, it may not materialize the mapping between the intents and queries. As the game progresses, it should maintain the reward values for the seen signals and their returned intents. The number of under-specified queries that return a certain entity is less than to the arity of the database relation, which is normally around 30-50 for most databases. Further, the query interface do not need to store the values in the queries and can store the queries using pointers to the values in their associated entities in the database. Hence, the query interface may efficiently maintain and search the reward table for medium size databases. In practice, the maximum number of under-specified queries is smaller than the arity of the database relation.

5 Analysis of the Learning Rule

In this section, we provide an analysis of the reinforcement mechanism provided above and will show that, statistically speaking, the adaptation rule leads to improvement of the efficiency of the interaction. The extensive study of this interaction is well beyond the page limits set here. As a result we provide the following simplifying assumption.

Assumption 5.1 *We assume that the cardinality of the list k is 1.*

Indeed the analytical work provided in this work is extendable to lists with arbitrary cardinality.

For the analysis of the the reinforcement learning mechanism in Section 4 and for simplification, denote

$$u(t) := u_1(P, Q(t)) = u_2(P, Q(t)), \tag{4}$$

where $u_1 = u_2$ is defined in (1).

We recall that a random process $\{X(t)\}$ is a submartingale [7] if

$$E(X(t+1) \mid \mathcal{F}_t) \geq X(t),$$

where $\mathcal{F}_t$ is the σ-algebra generated by $X_1, \ldots, X_t$. In other words, a process $\{X(t)\}$ is a submartingale if the expected value of $X(t+1)$ given the past, is not strictly less than the value of X_t.

The main result here is that the random process $u(t)$ defined by (4) is a submartingale when the reinforcement learning rule in Section 4 is utilized. To show this, we discuss an intermediate result. For simplicity of notation, we use superscript $+$ to denote variables at time $(t+1)$ and drop the dependencies at time t for variables depending on time t. Throughout the rest of our discussions, we let $\{\mathcal{F}_t\}$ be the natural filtration for the process $\{Q(t)\}$, i.e. $\mathcal{F}$ is the σ-algebra generated by $P(t)$.

Lemma 5.2 *For any $i \in [m]$ and $j \in [n]$ (and any time $t \geq 0$), we have*

$$E(Q^+_{ji} \mid \mathcal{F}_t) - Q_{ji} = \frac{Q_{ji}}{\sum_{\ell'=1}^m R_{j\ell'} + 1} \left(\pi_i P_{ij} - u^j(P, Q)\right),$$

where

$$u^j(P, Q) = \sum_{\ell=1}^m \pi_\ell P_{\ell j} Q_{j\ell},$$

is the average efficiency of signal j on conveying messages.

Proof: Fix $i \in [m]$ and $j \in [n]$. Let A be the event that at the t'th iteration, we reinforce a pair (j, ℓ) for some $\ell \in [m]$. Then on the complement A^c of A, $Q^+_{ji}(\omega) = Q_{ji}(\omega)$. Let $A_1 \subseteq A$ be the subset of A such that the pair (j, i) is reinforced and $A_2 = A \setminus A_1$ be the event that some other pair (j, ℓ) is reinforced for $\ell \neq i$.

We note that

$$Q^+_{ji} = \frac{R_{ji} + 1}{\sum_{\ell=1}^m R_{j\ell} + 1} 1_{A_1} + \frac{R_{ji}}{\sum_{\ell=1}^m R_{j\ell} + 1} 1_{A_2} + Q_{ji} 1_{A^c}.$$

Therefore, we have

$$\begin{aligned} E(Q^+_{ji} \mid \mathcal{F}_t) &= \pi_i P_{ij} Q_{ji} \frac{R_{ji} + 1}{\sum_{\ell=1}^m R_{j\ell} + 1} \\ &+ \sum_{\ell \neq j} \pi_\ell P_{\ell j} Q_{j\ell} \frac{R_{ji}}{\sum_{\ell'=1}^m R_{j\ell'} + 1} + (1-p) Q_{ji}, \end{aligned}$$

where $p = P(A_2 \mid \mathcal{F})$. Note that $Q_{ij} = \frac{R_{ji}}{\sum_{\ell=1}^m R_{j\ell}}$ and hence,

$$\begin{aligned} E(Q^+_{ji} \mid \mathcal{F}_t) - Q_{ji} = \frac{1}{\sum_{\ell'=1}^m R_{j\ell'} + 1} \Bigg(& \pi_i P_{ij} Q_{ji} \sum_{\ell \neq i} Q_{j\ell} \\ & - \sum_{\ell \neq i} \pi_\ell P_{\ell j} Q_{j\ell} Q_{ji} \Bigg). \end{aligned}$$

Replacing $\sum_{\ell \neq i} Q_{j\ell} = 1 - Q_{ji}$ and adding/subtracting $\pi_i P_{ij} Q_{ji} Q_{ji}$ in the term inside the parenthesis in the above equality, we get

$$E(Q_{ji}^+ \mid \mathcal{F}) - Q_{ji} = \frac{Q_{ji}}{\sum_{\ell'=1}^m R_{j\ell'} + 1} \left(\pi_i P_{ij} - u^j(P, Q)\right).$$

□

Using Lemma 5.2, we show that the process $\{u(t)\}$ is a sub-martingale.

Theorem 5.3 *Let $\{u(t)\}$ be the sequence given by* (4). *Then, $\{u(t)\}$ is a submartingale sequence.*

Proof: Let $u^+ := u(t+1)$, $u := u(t)$, $u^j := u^j(P(t), Q(t))$ and also define $\tilde{R}^j := \sum_{\ell'=1}^m R_{j\ell'} + 1$. Then, using the linearity of conditional expectation and Lemma 5.2, we have:

$$\begin{aligned} E(u^+ \mid \mathcal{F}_t) - u &= \sum_{i=1}^m \sum_{j=1}^n \pi_i P_{ij} \left(E(Q_{ji}^+ \mid \mathcal{F}_t) - Q_{ji}\right) \\ &= \sum_{i=1}^m \sum_{j=1}^n \pi_i \frac{P_{ij} Q_{ji}}{\sum_{\ell'=1}^m R_{j\ell'} + 1} \left(\pi_i P_{ij} - u^j\right) \\ &= \sum_{j=1}^n \frac{1}{\tilde{R}^j} \left(\sum_{i=1}^m Q_{ji} (\pi_i P_{ij})^2 - (u^j)^2\right). \end{aligned} \tag{5}$$

Note that Q is a row-stochastic matrix and hence, $\sum_{i=1}^m Q_{ji} = 1$. Therefore, by the Jensen's inequality [7], we have:

$$\sum_{i=1}^m Q_{ji} (\pi_i P_{ij})^2 \geq \sum_{i=1}^m (Q_{ji} \pi_i P_{ij})^2 = (u^j)^2.$$

Replacing this in the right-hand-side of (5), we conclude that $E(u^+ \mid \mathcal{F}_t) - u \geq 0$ and hence, the sequence $\{u(t)\}$ is a submartingale. □

The above result implies that the effectiveness of database, stochastically speaking, increases as time progresses when the learning rule in Section 4 is utilized. This is indeed a desirable property for any adapting/learning scheme for database adaptation.

An immediate consequence of Theorem 5.3 is that the efficiency sequence $\{u(t)\}$ is convergent almost surely.

Corollary 5.4 *The sequence $\{u(t)\}$ given by* (4) *converges almost surely.*

Proof: Note that $0 \leq u(t) \leq mn$ (indeed, a simple application of Hölder's inequality give the bound $u(t) \leq 1$) and hence, $\{u(t)\}$ is a bounded submartingale. Therefore, by the Martingale Convergence Theorem [7], it follows that $\lim_{t \to \infty} u(t)$ exists almost surely. □

6 Conclusion & Future Work

We modeled the interaction between the user and the database query interface as a repeated signaling game, where the players starts with different mapping between signals, i.e. queries, and objects, i.e. desired entities, and like to reach a common mapping. We proposed an adaptation mechanism for the query interface to learn the signaling strategy of the user and prove that this mechanism increases the expected payoff for both user and the query interface in average and converges almost surely. We plan to explore the possible equilibria of this game where the user modifies her signaling strategy with a different rate from the query interface. Further, it may be challenging to efficiently maintain and updated the signaling strategy of the query interface for very large databases. We plan to investigate and address these challenges.

References

[1] E. Agichtein, R. W. White, S. T. Dumais, and P. N. Bennet. Search, interrupted: understanding and predicting search task continuation. In *SIGIR*, 2012.

[2] P. Avesani and M. Cova. Shared lexicon for distributed annotations on the Web. In *WWW*, pages 207–214, 2005.

[3] T. Beckers et al. Report on INEX 2009. *SIGIR Forum*, 44(1):38–57, 2010.

[4] S. Chaudhuri, G. Das, V. Hristidis, and G. Weikum. Probabilistic information retrieval approach for ranking of database query results. *ACM Trans. Database Syst.*, 31(3):1134–1168, 2006.

[5] Y. Chen, W. Wang, Z. Liu, and X. Lin. Keyword search on structured and semi-structured data. In *SIGMOD*, 2009.

[6] I. Cho and D. Kreps. Signaling games and stable equilibria. *Quarterly Journal of Economics*, 102:179–221, 1987.

[7] R. Durrett. *Probability: theory and examples*. Cambridge university press, 2010.

[8] N. Fuhr and T. Rolleke. A probabilistic relational algebra for the integration of information retrieval and database systems. *TOIS*, 15, 1997.

[9] Y. Hu, B. Skyrms, and P. Tarrès. Reinforcement learning in signaling game. *arXiv preprint arXiv:1103.5818*, 2011.

[10] A. Kotov, P. N. Bennett, R. W. White, S. T. Dumais, and J. Teevan. Modeling and analysis of cross-session search tasks. In *SIGIR*, 2011.

[11] D. Lewis. *Convention*. Cambridge: Harvard University Press, 1969.

[12] J. Luo, S. Zhang, and H. Yang. Win-win search: Dual-agent stochastic game in session search. In *SIGIR*, 2014.

[13] M. A. Nowak and D. C. Krakauer. The evolution of language. *Proceedings of the National Academy of Sciences*, 96(14):8028–8033, 1999.

[14] H. Wang, Y. Song, M.-W. Chang, X. He, R. W. White, and W. Chu. Learning to extract cross-session search tasks. In *WWW*, 2013.