The BAY-HIST Prediction Model for RDF Documents

Edna Ruckhaus and María-Esther Vidal

Universidad Simón Bolívar
Caracas, Venezuela
{ruckhaus, mvidal}@ldc.usb.ve

Abstract. In real-world RDF documents, property subject and object values are often correlated. The identification of these relationships is of significant relevance to many applications, e.g., query evaluation planning and linking analysis. In this paper we present the BAY-HIST Prediction Model, a combination of Bayesian networks and multidimensional histograms which is able to identify the probability of these dependencies. In general, Bayesian networks assume a small number of discrete values for each of the variables considered in the network. However, in the context of the Semantic Web, variables that represent the concepts in large-sized RDF documents may contain a very large number of values; thus, BAY-HIST implements multidimensional histograms in order to aggregate the data associated with each node in the network. We illustrate the benefits of applying BAY-HIST to the problem of query selectivity estimation as part of cost-based query optimization. We report initial experimental results on the predictive capability of this model and the effectiveness of our optimization techniques when used together with BAY-HIST. The results suggest that the quality of the optimal evaluation plan has improved over the plan identified by existing cost models that assume independence and uniform distribution of the data values.

1 Introduction

The number of controlled vocabularies and annotated data sources in the Web has exploded in the last few years. Individually, many of these documents contain a large number of concepts and instances, and additionally their growth rate is very high. Thus, in order to be capable of scaling up, Web architectures have to be tailored for query processing on large number of resources and instances. We apply BAY-HIST to the problem of query selectivity estimation as part of cost-based query optimization.

The Prediction Model BAY-HIST is a framework that combines Bayesian networks and multidimensional histograms with the purpose of determining dependencies between properties in RDF documents and the distribution of their values. Bayesian Networks are probabilistic models that allow a compact representation of the joint distribution of the concepts defined in an RDF document. In general, Bayesian networks assume a small number of discrete values for each of the variables considered in the network. However, in the context of RDF documents in the Semantic Web, variables that represent the concepts in large-sized RDF documents may contain a very large number of values; thus, BAY-HIST implements multidimensional histograms in order to aggregate the data associated with each node in the Bayesian network that represents the RDF document.

BAY-HIST has been included as a component of the OneQL System, an Ontology System that provides optimization and query evaluation techniques that scale up to large RDF/RDF(S) documents [4, 10]. We report initial experimental results on the predictive capability of this model and the effectiveness of our optimization techniques when used together with BAY-HIST. The results suggest that the quality of the optimal evaluation plan has improved compared to the plan identified by existing cost models that assume independence and uniform distribution of the data values, by up to two orders of magnitude.

The structure of this paper is as follows: first, we will give a motivating example. Following this, we will present the syntax and semantics of BAY-HIST. Next, we will explain the architecture of the BAY-HIST Prediction Model and its application to cost-based query optimization. Then, the experimental study will be described, and finally, the conclusions and future work will be presented.

2 A Motivating Example

The example that follows shows a query to the RDF repository published at `http://www.govtrack.us/`. In this example, besides information concerning the U.S. congress bills voting process, we consider information of the census such as religion and gender, and political information such as the party and the state that is represented by each representative that participates in the voting process. Consider the relationships between party, gender, religion, state and the way a representative votes. To discover if there is any correlation among the values of these five properties, we will try to determine if for different instantiations of the following query, different number of tuples are obtained: *Names of all the representatives of state ?S, that belong to party ?P, are of gender ?G, are of religion ?R and have voted for the winning option in the voting process of Bill ?B.* The SPARQL representation of this query is illustrated in Figure 1.

```
PREFIX pol:<tag:http://www.rdfabout.com/rdf/schema/politico/>
PREFIX vote:<tag:http://www.rdfabout.com/rdf/schema/vote/>
PREFIX foaf:<tag:http://xmlns.com/foaf/0.1/>
SELECT ?X
FROM <tag:http://www.examples.org/votesdataset/>
WHERE
        {?X pol:forOffice ?S . ?X pol:party ?P . ?Z pol:hasRole ?X . ?Z foaf:gender ?G .
        ?Z foaf:religion ?R . ?O vote:votedBy ?X . ?B vote:winner ?O}
```

Fig. 1. A SPARQL query

This query may have different subject and object instantiations (constants). For instance, we may want to explore for a certain Bill, the different combinations of instantations for party, religion, gender and state. While for a certain set of instantiations the query has 18 answers, for another one it has no answers. This behavior is due to the lack of uniformity in the property value distribution and the dependency between properties. For example, the probability that a representative has voted for the winning option in the voting process of Bill 1998-173 if he is Catholic, male, belongs to the Democratic party and represents the state of Massachussets is much higher than the probability that a representative has voted for the winning option in the voting process of Bill 1998-173 if he is Jewish, male, Republican and represents Oklahoma. The identification of these relationships is of significant relevance to many applications. For instance, in query evaluation planning, this information may provide the basis for the optimizer to discriminate between bad or good query plans.

3 The BAY-HIST Prediction Model

Consider the RDF repository presented in the previous example. Let us assume that there are certain causal relationships between the subjects and objects of properties that are represented as an RDF Bayesian Network (RBN), as shown in Figure 2. In this RBN, there are nodes that represent property subjects or objects. For example, node `o-religion` represents the values (objects) of property `religion`. We also represent the event of a combination between subjects or objects of related properties. Such is the case of node `s-s-foroffice-party` that represents the event that a subject that is representing a certain state, belongs to a certain party. The arcs in this network represent dependencies between nodes. In this network we model that the combination of voter and gender is conditioned not only by the gender itself, but also by the state he represents and the party to which he belongs to; thus, the probability that a person's gender is 'male', the state is 'Oklahoma' and that he belongs to the 'Republican' party is 0.033. This probability is related to the probabilities of all the rest of combinations of gender, state and party. Tables 1(a) and 1(b) show a portion of the conditional probability tables (CPT) of this RBN. An RBN represents all the conditional dependencies among prop-

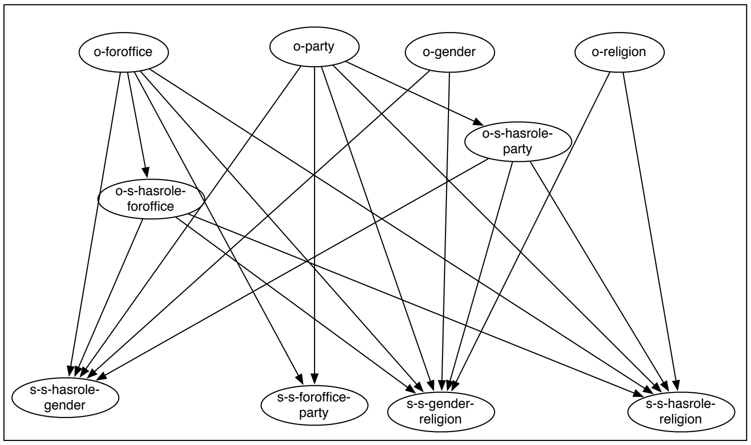


Fig. 2. RBN Votes

Table 1. CPT's Votes

(a) CPT o-party

o-party	prob(o-party)
Democratic	0.51
Independent	0.007
Republican	0.47

(b) CPT s-s-foroffice-party

s-s-foroffice-party	o-foroffice	o-party	prob(s-s-foroffice-party)
true	Democratic	ak	0
false	Democratic	ak	1
true	Independent	ak	0
false	Independent	ak	1
true	Republican	ak	0.03
false	Republican	ak	0.97
true	Democratic	ma	0.038
false	Democratic	ma	0.962
...	...	...	...

erty subjects and objects in an RDF document. Next, we will formally define an RDF Bayesian Network:

Definition 1 (RDF Bayesian Network) *Given an RDF directed graph $O_R = (V_R, E_R)$ where V_R and E_R are the nodes and arcs in the RDF graph. An* **RDF Bayesian Network** *R_B for O_R, is a pair $R_B = \langle O_B, CPT_B \rangle$, where $O_B = (V_B, E_B)$ is a DAG. V_B are the nodes in O_B and E_B are the arcs in O_B. CPT_B are the Conditional Probability Tables for each node. The homomorphism $f : \mathbb{P}(E_R) \rightarrow \mathbb{P}(V_B)$ establishes mappings between O_R and O_B:*

$$f(\{(sub, pro, obj)\}) = \{s\text{-}pro, o\text{-}pro\} \qquad \text{(Mapping 1)}$$

$$f(\{(sub_1, pro_1, obj), (sub_2, pro_2, obj)\}) = \{o\text{-}o\text{-}pro_1\text{-}pro_2, o\text{-}o\text{-}pro_2\text{-}pro_1\} \qquad \text{(Mapping 2)}$$

$$f(\{(sub, pro_1, obj_1), (sub, pro_2, obj_2)\}) = \{s\text{-}s\text{-}pro_1\text{-}pro_2, s\text{-}s\text{-}pro_2\text{-}pro_1\} \qquad \text{(Mapping 3)}$$

$$f(\{(sub, pro_1, obj_1), (sub_2, pro_2, sub)\}) = \{s\text{-}o\text{-}pro_1\text{-}pro_2, o\text{-}s\text{-}pro_2\text{-}pro_1\} \qquad \text{(Mapping 4)}$$

$V_C \subseteq V_B$, where V_C is the union of the sets of nodes established by mappings 2 to 4, and it is comprised of all the nodes that represent property combinations.

$E_B \subseteq V_B \times V_C$ is the set of arcs. An arc $(v_1, v_2) \in E_B$ iff there exist two sets of nodes in the RBN, $V_1 \subseteq V_B$ and $V_2 \subseteq V_C$ such that, $v_1 \in V_1$ and $v_2 \in V_2$ and when f^{-1} is applied to these sets, a subset of arcs in the RDF graph is obtained.

CPT_B is the probability $Pr(v/predecessors(v))$ for each node $v \in V_B$, i.e., the distribution on the values of v for each possible value assignment of its predecessors. The CPT_B are multidimensional histograms ordered by value. If a node v is a source node, the histogram will be one-dimensional, because in this case the CPT_B only represents the distribution of values taken up by the variable represented by the node. For each node v, according to the properties of the distribution of the values of v, CPT_B can be represented as an *equi-width* histogram or as an *equi-height* histogram.

Example 1 *Next, we illustrate the use of the homomorphism f. Figure 3 shows a portion of an RDF graph (O_R) and its corresponding RBN graph (O_B). Mapping 1 is applied to the sets of RDF arcs* `{(rep1,foroffice,va)}` *and* `{(rep2,party,democratic)}`*:*

f`({(rep1,foroffice,va)})={s-foroffice,o-foroffice}`

f`({(rep2,party,democratic)})={s-party,o-party}`

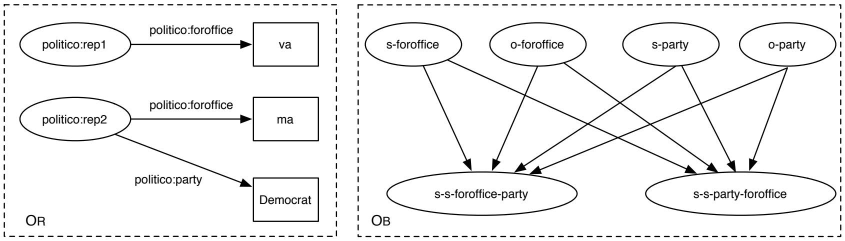


Fig. 3. Example Mapping RDF Graph - RBN Graph

Then, Mapping 3 is applied to the set of RDF arcs {`(rep2,foroffice,ma)`, `(rep2,party,democratic)`}

$$f(\{(\texttt{rep2,foroffice,ma}),(\texttt{rep2,party,democratic})\})=\{\texttt{s-s-foroffice-party,s-s-party-foroffice}\}$$

The arc `(o-foroffice,s-s-foroffice-party)` *belongs to E_B because the arcs obtained by applying the inverse of f are subsets of E_R:*

$$f^{-1}(\{\texttt{s-foroffice,o-foroffice}\})\cup f^{-1}(\{\texttt{s-s-foroffice-party,s-s-party-foroffice}\})=$$
$$\{(\texttt{rep1,foroffice,va}),(\texttt{rep2,foroffice,ma}),(\texttt{rep2,party,democratic})\}$$

Intuitively, an RBN is semantically valid if its arcs have been established between nodes that map to properties whose subjects and objects are of the same type, i.e., have some type of matching instantiations, subject-subject, subject-object or object-object. For example, an arc from node `o-s-hasrole-party` to node `s-s-gender-religion` is semantically valid because there are matching subject-subject instantiations between triples of property *hasrole* and triples of *religion*, i.e., both are "persons".

Given the symmetry property of the combinations between triple patterns, the set V_B may contain only one of the nodes in the sets defined with mappings 2, 3 and 4 in Definition 1; thus, the resulting RBN is minimal:

Definition 2 (Minimal RBN) *Given an RBN $R_B = \langle O_B, CPT_B \rangle$. R_B is a* **Minimal RBN** *if the set V_B contains exactly one node in sets* {*s-s-pro$_1$-pro$_2$, s-s-pro$_2$-pro$_1$*}, {*s-o-pro$_1$-pro$_2$, o-s-pro$_2$-pro$_1$*} *and* {*o-o-pro$_1$-pro$_2$, o-o-pro$_2$-pro$_1$*}.

4 Architecture

Figure 4 shows the architecture of the BAY-HIST Prediction Model System. BAY-HIST has two main components that generate and query the RBN: the RBN Analyzer and the RBN Inference Engine. Both components make use of the *SamIam* Bayesian Inference Tool [1].

The analyzer receives an RDF document and creates the RBN structure using the mappings presented in Definition 1 to establish the correspondence between the RDF graph and the nodes and arcs of the RBN structure. Once the RBN structure has been defined, the RDF data is loaded into relational tables, and a multi-dimensional histogram is generated for each node in the RBN structure through the stored procedures and the histogram option implemented by the DBMS Oracle [8]. Both, the RBN structure and CPT's are fed to the *SamIam* network editor, and a Bayesian network is generated in one of the internal formats recognized by the *SamIam* tool.

When a query is received, the RBN Inference Engine constructs the corresponding probability query (e.g., marginal probability and posterior marginal probability) and passes this query on to the *SamIam* inference engine which then returns an answer.

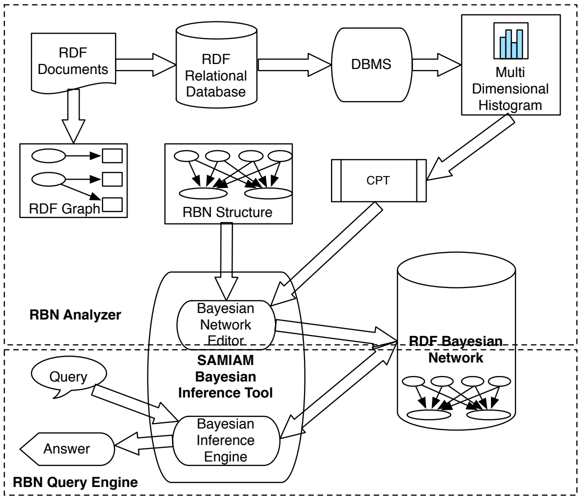


Fig. 4. Architecture of the BAY-HIST System

5 Application of BAY-HIST to Query Optimization

The BAY-HIST Prediction Model is applied to query selectivity estimation. These estimates are used within the cost model of a cost-based query optimizer as part of the formulas that compute the cost and cardinality of query sub-plans. We have developed a randomized optimization strategy based on the Simulated Annealing algorithm [7]. This algorithm explores execution plans of any shape (bushy trees) in contrast to other optimization algorithms that explore a smaller portion, e.g., left-linear plans. Random walks are performed in stages that consist of an initial random *plan generation step* followed by one or more *plan transformation steps*. An equilibrium condition or a number of iterations determines the number of transformation steps in each stage.

The probability of transforming a current plan p into a new plan p' is specified by an acceptance probability function $P(p, p', T)$ that depends on a global time-varying parameter T called the *temperature* which reflects the number of stages to be executed. Function P may be nonzero when $cost(p') > cost(p)$, meaning that the optimizer can produce a new plan even when it has a higher cost than the current one. This feature prevents the optimizer from becoming stuck in a local minimum. Temperature T is decreased during each stage, and the optimizer concludes when $T = 0$. Transformations applied to the plan during the random walks correspond to SPARQL axioms, e.g., commutativity and associativity of the '.' operator. The optimizer is able to identify near optimal solutions because of the precision of estimates that take into account correlations of values and non uniform distribution.

Using BAY-HIST, the selectivity of an RDF query execution plan that joins A and B over join arguments $\mathcal{J}$ $(A \bowtie_{\mathcal{J}} B)$ is expressed in terms of a probability query against the corresponding RBN:

$$fs(A \bowtie_{\mathcal{J}} B) = \prod_{J \in \mathcal{J}} Pr(JoinEvent_J / (JoinEvid_{\mathcal{J}_A} \wedge JoinEvid_{\mathcal{J}_B} \wedge instEvid_{\mathcal{I}_A} \wedge instEvid_{\mathcal{I}_B}))$$

This is a posterior marginal probability query, i.e., the probability that two pattern instantiations are combined, given the evidence of the instantiations and the joins in its left and right sub-trees.

The probability queries associated with an RDF pattern (the base case) correspond to marginal probabilities, i.e., to the probability that the value of subjects or objects of the property in the pattern is equal to the instantiation in the pattern: $Pr(\text{o-pro=obj})$, $Pr(\text{s-pro=sub})$ or $Pr(\text{s-pro=sub} \wedge \text{o-pro=obj})$.

An estimate of the selectivity of an RDF pattern A, carried out by using a probability query on the RBN is more precise than an estimate carried out by using the traditional cost model. The traditional cost model defines the following selectivity formula:

$$fs(A, \mathcal{J}) = \prod_{J \in \mathcal{J}} 1/nKeys(A, J) \tag{1}$$

where $nKeys(A, J)$ is the number of different values taken up by J in pattern A. Likewise, an estimate of the selectivity of a sub-plan $A \bowtie_{\mathcal{J}} B$ carried out through a probability query on the RBN is more precise than an estimate carried out through the traditional cost model. The selectivity formula in the traditional cost model is as follows:

$$fs(A, B, \mathcal{J}) = \prod_{J \in \mathcal{J}} 1/max(nKeys(A, J), nKeys(B, J)) \tag{2}$$

These traditional formulas do not compute a precise estimate of the query evaluation costs because they are based on the following assumptions: (a) the values of the subjects and objects in a triple pattern are uniformly distributed, (b) the values of the subjects and objects in a pattern are independent, and (c) the values of the subjects and objects in properties of the patterns that are combined in a query, are independent.

The example that follows shows the motivating example query with two different sets of instantiations:

- *Names of all the male representatives of the state of Massachussets that belong to the Democratic party, are Catholic and have voted for the winning option in the voting process of Bill 1998-173.*
- *Names of all the male representatives of the state of Oklahoma that belong to the Republican party, are Jewish and have voted for the winning option in the voting process of Bill 1998-173.*

```
PREFIX pol:<tag:http://www.rdfabout.com/politico/>
PREFIX vote:<tag:http://www.rdfabout.com/vote/>
PREFIX foaf:<tag:http://xmlns.com/foaf/0.1/>
SELECT ?X
FROM <tag:http://www.examples.org/votesdataset/>
WHERE
        {?X pol:forOffice senate:ma .
        ?X pol:party 'Democratic' .
        ?Z foaf:gender 'male' .
        ?Z pol:hasRole ?X .
        ?Z foaf:religion 'Catholic' .
        ?O vote:votedBy ?X .
        '1998-173' vote:winner ?O}
```

(a) SPARQL Query 1

```
PREFIX pol:<tag:http://www.rdfabout.com/politico/>
PREFIX vote:<tag:http://www.rdfabout.com/vote/>
PREFIX foaf:<tag:http://xmlns.com/foaf/0.1/>
SELECT ?X
FROM <tag:http://www.examples.org/votesdataset/>
WHERE
        {?X pol:forOffice senate:ok .
        ?X pol:party 'Republican' .
        ?Z foaf:gender 'male' .
        ?Z pol:hasRole ?X .
        ?Z foaf:religion 'Jewish' .
        ?O vote:votedBy ?X .
        '1998-173' vote:winner ?O}
```

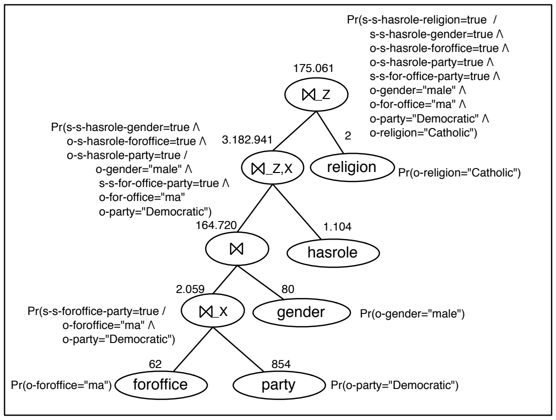
(b) SPARQL Query 2

Fig. 5. Two Queries with Different Instantiations

The SPARQL representation of these two queries is illustrated in Figure 5. Query 1 and Query 2 differ in their subject and object instantiations (constants), and their answers are different: while the first query has 18 answers, the second one has no answers. This behavior is due to the lack of uniformity in the property value distribution and the dependencies between properties. Based on this observation, we use an RBN to differentiate the selectivity of the sub-plans of each query execution plan taking into account the existing correlation between the various RDF properties. To estimate the selectivity of the sub-plan shown in Figure 6(a), a posterior marginal probability query is carried out in the RBN and the result of this probability query is 0.0275.

```
{?X pol:forOffice senate:ma .
?X pol:party 'Democratic' .
?Z foaf:gender 'male' .
?Z pol:hasRole ?X .
?Z foaf:religion 'Catholic'}
```

(a) Query Sub-Plan



(b) Sub-Plan Tree

Fig. 6. Probability Queries on an Execution Sub-plan (`http://www.govtrack.us/`)

For the corresponding sub-plan in the second query, i.e., the same sub-plan with different instantiations, the result of the inference on the RBN is 0, which is consistent

with the expectation that the cardinality of the first query is higher than the cardinality of the second query. Figure 6(b) shows the tree representation of the sub-plan in Figure 6(a). Each node is annotated with the probability query corresponding to the sub-plan (sub-tree) selectivity estimate, and with its cardinality. The cost estimate of the sub-plan, is equivalent to the total number of intermediate results that must be estimated to obtain the answer:
$cost(P) = 62 + 854 + 2.059 + 80 + 164.720 + 1.104 + 3.182.941 + 2 = 3.351.822$

6 Related Work

In [6], Bayesian networks are applied to the problem of imprecise estimates of the selectivity of a relational query; this framework is known as the Probabilistic Relational Model (PRM). This imprecision stems from the assumption of uniform distribution of values for attributes in a table, attribute independence in one table, and attribute independence in tables that are semantically related. The proposed solution uses a probabilistic model to represent the distribution of values of each attribute and the correlations between attributes. Thus, instead of computing the query selectivity in terms of the number of different values of each attribute in the *select* condition of the query, the selectivity is computed using the result of a probability query to the model. In [5], Statistical Relational Models (SRM) were developed. They are different from PRM because they represent a statistical model of a particular database state instead of representing any state. Thus, Conditional Probability Table (CPT) construction in SRMs is done through queries to the database whereas the structure and CPT construction in PRMs is conducted by using machine-learning techniques.

The difference between the solution proposed by Getoor, et. al. [5, 6] and the solution presented in our paper, is the scalability to large-sized RDF repositories by means of multidimensional histograms. The SRM, developed in [5] assume a low number of values for each variable in the model. On the other hand, although in our work, an RDF document is modeled similarly to an SRM, its nodes and arcs have a particular semantics based on the RDF graph semantics, i.e., subject, property and object triples. Besides this, in our proposed RBN model, there are also *Join* variables, but restricted to the possible combinations between subjects and objects. Additionally, the purpose of the Bayesian network proposed by Getoor, et. al., is the estimation of query selectivity. In our work, Bayesian networks are applied to RDF documents in order to estimate the selectivity of query evaluation plans and sub-plans.

The work described in [9, 11, 12] extends the Ontology Web Language (OWL) with constructs that allow the annotation of an ontology with probabilities and causal relationships. These annotations are done with the purpose of reasoning on uncertainty in ontologies. Once an ontology is annotated, it is translated to a Bayesian network, and Bayesian inference queries may be answered. The main difference between these models and our research is that since the information on subject an object values are kept in an aggregated form, our combinated approach of Bayesian networks and multidimensional histograms scales up to large RDF documents. Besides this, in our work we define random variables that represent the event that a property may be combined (*Join*) with another property; these type of variables are not considered in these approaches.

7 Experimental Study

The goal of the experimental study was to analyze the benefits of the proposed predictive model when applied to the problem of query optimization. First, the predictive capacity of the model was studied and then, the quality of the optimal query was compared to the original query and to the optimal plan identified by a cost model that assumes independence between properties and uniform distribution of values.

We used the real-world dataset on the US Congress bills voting process for the years 1998, 1999 and 2000 published at `http://www.govtrack.us/`. Besides the election results, we also consider census information about representatives such as religion and gender, and political information such as the party and their state. The number of triples in the dataset for years 1998, 1998-1999 and 1998-1999-2000 is $50,860$, $94,590$ and $128,852$, respectively.

The query benchmark is comprised of 112 queries with five instantiated patterns. The properties in the patterns and their ordering are the same for all queries, but the instantiations are different. Previously, we determined that these properties are correlated and thus, queries with different instantiations will have different selectivity.

We use the Bayesian inference tool, *SamIam* [1], to build the RBN based on the graph structure, and the CPT which is represented as a multidimensional histogram. Currently, the graph structure is built by hand, but this could be done semi-automatically. The graph in the RBN was built according to the properties represented in the ontology. Then, the CPT were developed using multidimensional histograms to aggregate the node values. The structure of this RBN was illustrated before as Figure 2. Each CPT for a target node is a multidimensional histogram, where the first dimension corresponds to a node itself, and the rest of the dimensions correspond to the predecessors of the node. The algorithm for multidimensional histogram generation constructs a histogram for the first dimension, and then for each bucket, it generates a histogram for the second dimension, and so on, until all dimensions are completed. These histograms were generated through the histogram options provided by the Oracle DBMS [8]. The default histogram option generates equal-width or equal-height histograms according to the number of different values of an attribute and its distribution.

In order to exploit the DBMS histogram mechanisms, we loaded a relational table for each property in the ontology. For each target node, we created a relational table that is a combination of the subject or object of the property that is represented by the node, with the subjects or objects of all its predecessors. We used methods in the Oracle package DBMS_STATS to generate an histogram on the column that represents the target node in the "combination" table. Then, for each bucket we created a table and again used DBMS_STATS to generate an histogram on the second dimension, and so on until all the dimensions had been covered. The histogram was completed with the computation of the frequency of each value of the target node given the different sets of values of its predecessors.

Bayesian inference queries are posed to the network through the *SamIam* tool in order to estimate the selectivity of each query based on the instantiations of its patterns. We use one of the algorithms implemented by *SamIam*, the Shenoy-Shafer exact inference algorithm [2]. Each query was also evaluated and we obtained the number of results. Thus, we compared the estimate of the selectivity with the actual number of answers. The correlation value is 0.95. This result indicates that there exists a linear relationship between the estimates and the actual values, so we may assert that the BAY-HIST model is capable of predicting the selectivity of a query plan or sub-plan, and therefore, we can have a precise estimate of this plan's evaluation cost.

The purpose of our next experiment was to study the effectiveness of our optimization techniques when used with the BAY-HIST prediction model. Given that the BAY-HIST model is capable of considering dependencies between properties and its distribution of values, the quality of the optimal plan identified by the optimizer using BAY-HIST should be better than the quality of a plan identified by an optimizer that uses a cost model that does not consider dependencies between properties and non-uniform distribution of values. We report on runtime performance, which corresponds to the *user time* produced by the *time* command of the Unix operation system.

We used the same dataset and RBN as the previous experiment. We also used the same query benchmark, but we shuffled the queries, evaluated them and chose the 21 queries that had the worst evaluation time. The experiment was performed using these 21 queries. The Simulated Annealing optimization algorithm was configured with an initial temperature of 700, and 20 iterations in the initial stage.

We compared the performance of the original query, the optimal plan identified by the optimizer with the model that assumes property independence and uniform distribution, and the optimal plan identified by the optimizer with the BAY-HIST model. These plans were evaluated with and without index structures[1].

The average evaluation time is reported in Figure 7. We can observe that the performance of the optimal plans without index structures exceeds the performance of the original queries by up to one order of magnitude. The improvement with the use of the index strucures with respect to the original plans is up to two orders of magnitude, but the improvement is even greater when the optimizer uses the BAY-HIST model. We also observed that this difference is proportional to the incremental size of the datasets.

These results indicate that the quality of the plan identified by the optimizer and the BAY-HIST model, is better than the quality of the optimal plan identified by the optimizer with the traditional prediction model and the benefits are even greater when index structures are used.

8 Conclusions and Future Work

We present the BAY-HIST Prediction Model, a combination of Bayesian networks and multidimensional histograms, which is able to estimate correlations between data values in an RDF document as well as their distribution. We study the benefits of applying BAY-HIST to the problem of query selectivity estimation as part of cost-based query optimization; also, we report initial experimental results that suggest that the quality of the optimal evaluation plans can be improved when selectivity is estimated using the BAY-HIST Prediction Model.

In the future we plan to use BAY-HIST on the RDF(S) and OWL formalisms; also, we will study the benefits of this prediction model when it is used to discover links be-

[1] Denoted as Bhyper according to the hypergraph RDF model that these index structures implement [3].

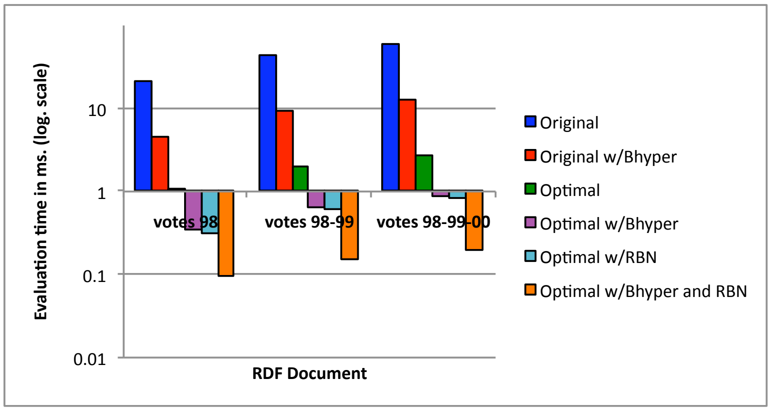


Fig. 7. Quality of the Optimal Plan

tween data terms. Currently, the optimization algorithm queries the RBN for the selectivity of all the sub-plans in each execution plan. Future work will also include keeping track of probability queries posed against an RBN in each execution plan, in order to improve the efficiency of the cost model.

References

1. *SamIam* - Sensitivity Analysis Modeling Inference and More. Automated Reasoning Group, University of California, Los Angeles. http://reasoning.cs.ucla.edu/samiam/.
2. Darwiche A. *Modeling and Reasoning with Bayesian Networks*. Cambridge University Press, 2009.
3. Martinez A. and Vidal M. A Directed Hypergraph Model for RDF. In *KWEPSY*, 2007.
4. Ruckhaus E., Ruiz E., and Vidal M. Query evaluation and optimization in the semantic web. *Theory and Practice of Logic Programming - TPLP*, 8(3):393–409, 2008.
5. Getoor L. Learning statistical models from relational data, 2001.
6. Getoor L., Taskar B., and Koller D. Selectivity estimation using probabilistic models. In *SIGMOD Conference*, pages 461–472, 2001.
7. Vidal M., Ruckhaus E., Lampo T., Martinez A., Sierra J., and Polleres A. Efficiently joining group patterns in SPARQL queries. In *Proceedings ESWC*, 2010.
8. *ORACLE*. Oracle Database Management System. http://www.oracle.com/.
9. Da Costa P., Laskey K., and Laskey K. PR-OWL: A bayesian ontology language for the semantic web. In *ISWC-URSW*, pages 23–33, 2005.
10. Lampo T., Ruckhaus E., Sierra J., Vidal M., and Martinez A. OnEQL: An Ontology-based Architecture to Efficiently Query Resources on the Semantic web. In *Proceedings of SSWS, collocated with ISWC*, 2009.
11. Yi Yang and Jacques Calmet. Ontobayes: An ontology-driven uncertainty model. In *Proceedings CIMCA '05*, 2005.
12. Ding Z., Peng Y., and Pan R. A Bayesian Approach to Uncertainty Modeling in OWL Ontology. In *Proceedings of the International Conference on Advances in Intelligent Systems - Theory and Applications*, 2004.