Impact Factor: 4.14 (Calculated by SJIF-2015)
e- ISSN: 2348-4470
p- ISSN: 2348-6406
International Journal of Advance Engineering and Research Development
Volume 3, Issue 4, April -2016

# Query Aware Determinization of Uncertain Objects

RESULT PAPER

Mr. Milind K.Patil[1], Assistant Prof. Yogesh S. Patil[2], H.O.D. & Associate Prof. Dinesh D. Patil[3]

[1]*Computer Science & Engg., J. T. Mahajan Polytechnic, Faizpur*
[2]*Computer Science & Engg., S.S.G.B.C.O.E.T, Bhusawal*
[3]*Computer Science & Engg., S.S.G.B.C.O.E.T, Bhusawal*

**Abstract —** *In this paper considers the problem of determinizing probabilistic data to enable such data to be stored in legacy systems that accept only deterministic input. Probabilistic data may be generated by automated data analysis/enrichment techniques such as entity resolution, information extraction, and speech processing. The legacy system may correspond to pre-existing web applications such as Flickr, Picasa, etc. The goal is to generate a deterministic representation of probabilistic data that optimizes the quality of the end-application built on deterministic data. We explore such a Determinization problem in the context of two different data processing tasks triggers and selection queries. We show that approaches such as thresholding or top-1 selection traditionally used for Determinization lead to suboptimal performance for such applications. Instead, we develop a query-aware strategy and show its advantages over existing solutions through a comprehensive empirical evaluation over real and synthetic datasets.*

***Keywords*** *— Determinization, uncertain data, data quality, query workload, branch and bound algorithm*

## I. INTRODUCTION

With the advent of cloud computing and the proliferation of web-based applications, users often store their data in various existing web applications. Often, user data is generated automatically through a variety of signal processing, data analysis/enrichment techniques before being stored in the web applications. For example, modern cameras support vision analysis to generate tags such as indoors/outdoors, scenery, landscape/portrait, etc. Modern photo cameras often have microphones for users to speak out a descriptive sentence which is then processed by a speech recognizer to generate a set of tags to be associated with the photo. The photo (along with the set of tags) can be streamed in real-time using wireless connectivity to Web applications such as Flickr [1].

Pushing such data into web applications introduces a challenge since such automatically generated content is often ambiguous and may result in objects with probabilistic attributes. For instance, vision analysis may result in tags with probabilities [2], [3], and, likewise, automatic speech recognizer (ASR) may produce an N-best list or a confusion network of utterances [4]. Such probabilistic data must be “determinate" before being stored in legacy web applications. We refer to the problem of mapping probabilistic data into the corresponding deterministic representation as the *Determinization* problem.

Many approaches to the *Determinization* problem can be designed. Two basic strategies are the Top-1 and all techniques, wherein we choose the most probable value / all the possible values of the attribute with non-zero probability, respectively. For instance, a speech recognition system that generates a single answer/tag for each utterance can be viewed as using a top-1 strategy. Another strategy might be to choose a threshold $\tau$ and include all the attribute values with a probability higher than $\tau$. However, such approaches being agnostic to the end-application often lead to suboptimal results as we will see later. A better approach is to design customized Determinization strategies that select a determinate Representation which optimizes the quality of the end-application.

Uncertain data are inherent in some important applications, such as environmental surveillance, market analysis, and quantitative economics research. Due to the importance of those applications and the rapidly increasing amount of uncertain data collected and accumulated, analyzing large collections of uncertain data has become an important task and has attracted more and more interest from the database community. Recently, uncertain data management has become an emerging hot area in database research and development. In this tutorial, we systematically review some representative studies on answering various queries on uncertain and probabilistic data [5].

Examples of such an end-app includes publishing/subscribing system such as Google Alert, where people put their subscriptions in the form of index keywords (e.g. Gujarat earthquake) and predicts over a database (e.g. this data is video). Google Alert finds all corresponding data sets to the user based on the subscriptions. Now for example a video about Gujarat Earthquake is to be uploaded on YouTube. The video has a set of tags that were decided using either by automatically vision processing and/or by information retrieval techniques put over transcribed speech.

Such tools which may create tags with probabilities (e.g., “Gujarat": 0.8, “earthquake":0.4, “election": 0.6), while the important tags of the video could be “Gujarat" and “earthquake”. The Determinization procedure should link the video with suitable tags such that subscribers or the users who are really very much involved in the video (i.e., whose subscription includes the words “Gujarat Earthquake") are notified while others are not overwhelmed by immaterial data.

 

Thus, in the given example, the Determinization process should minimize metrics called as false positives and false negatives that result from a determinize representation of data. Now take a example of different application such as Flickr, to which pictures are uploaded automatically from modern cameras along with the tags that may be generated based on speech recognition or image enrichment techniques. Flickr supports effective retrieval based on photo tags. In such an application, people may have interest in selecting determinize representation that optimizes set-based quality metrics such as F-measure instead of minimizing false positives/negatives.

In this paper, we study the difficulty of determinizing datasets with probabilistic attributes (usually generated by automatically by data analyses/enrichment). Our approach exploits a workload of triggers/queries to choose the top deterministic representation for two types of applications– one that chains triggers on generated content and another that supports effective retrieval. Interestingly, the trouble of Determinization has not been explored widely in the past. The most related research efforts are which explore how to give deterministic answers to a query (e.g. conjunctive selection query) over probabilistic database. Unlike the problem of determinizing an answer to a query, our aim is to determinate the data so as to enable it to be stored in legacy deterministic databases such that the determinize representation maximizes the anticipated performance of queries in the future. Solutions in cannot be straightforwardly applied to such a Determinization problem. Probabilistic data is studied in this paper; the works that are mostly related to ours is this project. They search how to determine answers to a query over a probabilistic data. In similarity, we have interest in best deterministic representation of data (and not Determinizing Probabilistic Data) so as to continue to use existing end-applications that take only deterministic input.

The conflicts in the two problem settings lead to many different challenges. Authors in the paper address a problem that chooses the set of uncertain objects to be cleaned, in order to achieve the best development in the quality of query answers. However, their aim is to improve quality of single query, while our aim is to optimize quality of overall query workload [6].

## II. RELATED WORK

Many advanced probabilistic data models were used in proposed systems. Here the centre of attention however was determinizing probabilistic objects, such as speech output and image tags, for which the probabilistic attribute model meet the requirements. It is to be noted that determining probabilistic data stored in more advanced probabilistic representation such as tree structures is also used. Several related research efforts that contract with the problem of selecting terms to index document for document retrieval.

A term-centric pruning method explains in keeps top postings for each term according to the individual score impact that each posting would have if the term appeared in a temporary search query. Here we propose a scalable term selection for text classification, is nothing but which is based on coverage of the terms. The centre of these research efforts is on significance – that is, getting the right set of terms that are most relevant to this paper. In our problem, a set of probably appropriate terms and their significance to the document are already specified by other data processing techniques. Thus, our objective is not to explore the significance of terms to documents, but to select keywords from the given set of terms to represent the paper, such that the quality of answers to triggers or queries is optimized. The main advantage of our proposed system is it will resolve the problem of determinization by reducing the expected cost of the answer to queries. Here we develop an efficient algorithm that achieves near-optimal quality. The algorithms which we are advice are very capable and reach high-quality results that are very close to those of the optimal solution [11].

Cutting edge information preparing strategies, for example, substance determination, information cleaning, data extraction, and mechanized labeling frequently deliver results comprising of items whose traits may contain instability. This vulnerability is every now and again caught as an arrangement of various fundamentally unrelated quality decisions for each questionable characteristic alongside a measure of likelihood for option values.

On the other hand, the lay end client, and some end-applications, won't not have the capacity to decipher the outcomes if yielded in such a structure. Along these lines, the inquiry is the manner by which to present such results to the client practically speaking, for instance, to bolster characteristic quality choice and article determination inquiries [12] the client may be keen on. Specifically, in this article we examine the issue of boosting the nature of these choice questions on top of such a probabilistic representation. The quality is measured utilizing the standard and generally utilized set-based quality measurements. We formalize the issue and after that create efficient approaches that give superb responses to these questions.

Uncertain data are inherent in some important applications, such as environmental surveillance, market analysis, and quantitative economics research. Uncertain data in those applications are generally caused by factors like data randomness and incompleteness, limitations of measuring equipment, delayed data updates, etc [5]. Due to the importance of those applications and the rapidly increasing amount of uncertain data collected and accumulated, analyzing large collections of uncertain data has become an important task and has attracted more and more interest from the database community.

 

### *2.1.* Determinizing Probabilistic Data

While we do not know of any previous work that directly addresses the problem of determinizing probabilistic data as studied in this paper, the works that are very related to ours are [1],[7]. They search how to determinize answers to a query over a *probabilistic* database. We are only concerned in top deterministic representation of data so as to keep on using accessible end-applications that take only deterministic input. The differences in the two problem settings lead to different challenges. Authors in [8] deal with a problem that chooses the list of uncertain objects to be cleaned, in order to realize the best development in the class of query answers. However, their aim is to get better value of single query, while ours is to optimize quality of overall query workload. Also, the focus is on how to choose the most excellent sets of objects and each chosen object is cleaned by human clarification, whereas we determinize all objects automatically. These differences effectively lead to different optimization challenges. Another allied area is MAP inference in graphical model [8], [9], whose goal is to discover the assignment to each variable that together maximizes the probability defined by the model. The determinization problem for the cost-based metric can be seen as a case of MAP inference problem. If we look the problem that way, the test in front of us is to develop a fast and high-valued inexact code to solve the equivalent NP-hard problem.

### *2.2.* Probabilistic Data Model

A range of highly developed data models have been proposed in the past. Our focus however was determinizing probabilistic objects, example image tags and speech output, for which the probabilistic attribute model suffices. We observe that determining probabilistic data stored in more highly advanced probabilistic models such as tree might also be interesting and can be possible [1]. Furthermore, our work to deal with data of such high complexity is an interesting future direction of work. There are many research efforts related that deals with the problem of selecting terms to number a document for document retrieval.

### 2.3. Key Term Selection

There are many research efforts related that deals with the problem of selecting terms to number a document for document retrieval. A term-centric pruning method explained in keeps topmost postings for each and every term according to the individual score impact that each and every posting will have if the term is seen in an for the function search query [1]. We propose a scalable term selection for categorization of text, which is based upon coverage of the terms coverage of the terms The focus of these research efforts is based on relevance – that is, finding the correct set of terms that are most relevant to document. In our problem, a set of possibly relevant terms and their relevance to the document are already given by other data dealing out techniques. Thus, our goal is not to find the relevance of terms to documents, but to find and select keywords from the given set of terms to represent the document, such that the quality of answers to triggers/queries is optimized.

### 2.4. Query intent disambiguation

Query information in such type of works is used to calculate many appropriate terms for queries, of queries. However, our aim is not to guess correct terms, but to find the correct keywords from the terms that are automatically generated by automated data generation tool [1].

### 2.5. Query and tag suggestions

Another related explore area is that of query suggestion and tag suggestion. On the basis of query-flow graphical representation of query information, authors in develop a measure of semantic similarity between queries, which is used for the task of producing diverse and useful recommendations. Rae et al. introduces an extendable structure of tag suggestion, using co-incidence examination of tags used in user detailed contents such as personal, social contact, social group and non user specific contents. The main objective of this is on how to make similarities and correlations between queries/tags and recommend queries/tags based on that information. However, our aim is not to measure similarity between object tags and queries, but to select tags from a given set of uncertain tags to optimize certain quality metric of answers to multiple [10].

## III. EXISTING SYSTEM

Many approaches to the *determinization* problem can be designed. The basic strategies are the all techniques, where in we choose the most probable value / all the possible values of the attribute with non-zero probability, respectively .For instance, a speech recognition system that generates a single answer/tag for each utterance can be viewed as using a top-1 strategy. Another strategy might be to choose a threshold and include all the attribute values with a probability higher than threshold. Existing system works address a problem that chooses the set of uncertain objects to be cleaned, in order to achieve the best improvement in the quality of query answers. There are several related research efforts that deal with the problem of selecting terms to index document for document retrieval. A term-centric pruning method described in existing system retains top postings for each term according to the individual score impact that each posting would have if the term appeared in a search query. Often lead to suboptimal results. They

 

explore how to determinize answers to a query over a *probabilistic* database. In contrast, we are interested in best deterministic representation of data (and not that of an answer to a query) so as to continue to use existing end-applications that take only deterministic input. Their goal is to improve quality of single query, while ours is to optimize quality of overall query workload. The main drawback of this approach is that it is *unaware* of the query workload ("query-unaware") and thus does not necessarily optimize the given quality metrics, which leads to lower quality.

## IV. PROPOSED SYSTEM

In this paper, we propose the problem of determinizing datasets with probabilistic attributes (possibly generated by automated data analyses/enrichment).Our approach exploits a workload of triggers/queries to choose the "best" deterministic representation for two types of applications – one, that supports triggers on generated content and another that supports effective retrieval. Interestingly, the problem of determinization has not been explored extensively in the past. The most related research efforts are which explore how to give deterministic answers to a query (e.g. conjunctive selection query) over probabilistic database. Unlike the problem of determinizing an answer to a query, our goal is to determinize the data to enable it to be stored in legacy deterministic databases such that the determinize representation optimizes the expected performance of queries in the future. Solutions cannot be straight forwardly applied to such a determinization problem.

A variety of advanced probabilistic data models have been proposed in the past. Our focus however was determinizing probabilistic objects, such as image tags and speech output, for which the probabilistic attribute model suffices. We note that determining probabilistic data stored in more advanced probabilistic models such as and/or tree might also be interesting. Extending our work to deal with data of such complexity remains an interesting future direction of work. There are several related research efforts that deal with the problem of selecting terms to index document for document retrieval. A term-centric pruning method described in retains top postings for each term according to the individual score impact that each posting would have if the term appeared in an adhoc search query. Authors in propose a scalable term selection for text categorization, which is based on coverage of the terms.

The focus of these research efforts is on relevance – that is, getting the right set of terms that are most relevant to document. In our problem, a set of possibly relevant terms and their relevance to the document are already given by other data processing techniques. Thus, our goal is not to explore the relevance of terms to documents, but to select keywords from the given set of terms to represent the document, such that the quality of answers to triggers/queries is optimized.

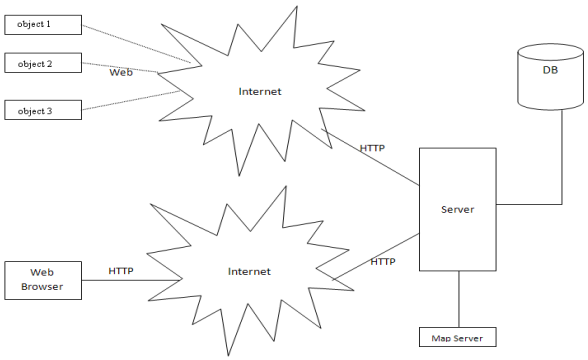


**Figure 1: System Architecture**

### 4.1. Branch and Bound Algorithm

As an alternative of performing a brute-force enumeration, we can make use of a faster branch and bound (BB) technique. The move towards will discovers response sets in a greedy fashion so that answer sets with lower cost tend to be discovered first. A branch-and-bound algorithm consists of a systematic enumeration of candidate solutions by means of state space search: the set of candidate solutions is notion of as forming a rooted tree with the full set at the root. The algorithm investigates branches of this tree, which symbolize subsets of the solution set. Before specifying the candidate solutions of a branch, the branch is checked against upper and lower estimated bounds on the optimal solution, and is leftover if it cannot produce a better solution than the best one found so far by the algorithm. The algorithm depends on the capable estimation of the lower and upper bounds of a region/branch of the search space and approaches comprehensive enumeration as the size (n-dimensional volume) of the region tends to zero. We will utilize to demonstrate the future BB algorithm.Instead of performing a brute-force enumeration; we can employ a faster branch and bound (BB) technique. The approach discovers answer sets in a greedy fashion so that answer sets with lower cost tend to be discovered first.

Branch and bound (BB or B&B) is an algorithm design paradigm for discrete and combinatorial optimization problems, as well as general real valued problems. A branch-and-bound algorithm consists of a systematic enumeration of candidate solutions by means of state space search: the set of candidate solutions is thought of as forming a rooted tree with the full set at the root. The algorithm explores *branches* of this tree, which represent subsets of the solution set.

 

Before enumerating the candidate solutions of a branch, the branch is checked against upper and lower estimated *bounds* on the optimal solution, and is discarded if it cannot produce a better solution than the best one found so far by the algorithm. The algorithm depends on the efficient estimation of the lower and upper bounds of a region/branch of the search space and approaches exhaustive enumeration as the size ($n$-dimensional volume) of the region tends to zero.

**4.2. Outline of the Branch Bound Algorithm**

The benefit of a unique model for all types of discrete optimization problems is that a general purpose Branch and Bound method is available. The two basic stages of a general Branch and Bound method:

1. Branching: splitting the problem into sub problems.
2. Bounding: calculating lower and/or upper bounds for the objective function value of the sub problem.

The branching is performed in the following algorithm by separating the current subspace into two parts using the internality requirement. Using the bounds, unpromising sub problems can be eliminated. Our general method for branch and bound algorithms involves modelling the solution space as a tree and then traversing the tree exploring the most promising sub trees first. This will continuous until either there are no sub trees into which to advance break the problem, or we have inwards at a point where, if we continue, only inferior solutions will be found. Let us have a look on a general algorithm for branch and bound searching is presented.

Search (A,B,best)
Pre: A=Solution space tree
B=Vertex in A
best = the solution which obtained as best so far

Post: best= the solution which obtained as best so far after searching sub tree rooted at B
If B is a complete solution more optimum than best=B
Generate the children of B
Compute Bound for vertices in sub tree of children X1....XK
X1....XK =feasible children with good lower bound for i=1 to k
If X i has a promising upper bound then search (A,X,best)

**Branch and bound searching**

Let us look at this technique more directly and discover that what is required to explain problems with the branch and bound method. We first need to define the objects that formulate the original problem and possible solutions to it.

**Problem instances:**

For the knapsack problem this would consist of two lists, one for the weights of the items and one for their values. Here we need an integer for the knapsack capacity. For chromatic numbers (or graph coloring), this is just a graph that could be accessible as an adjacency matrix, or better yet, an adjacency edge list.

**Solution tree:**

This must be an ordered edition of the solution search space, perhaps containing partial and infeasible solution candidates as well as all feasible solutions as vertices. For knapsack we built a depth-first search tree for the coupled integer programming problem with the objects ordered by weight. In the chromatic number solution tree we offered partial graph colorings with the first k nodes colored at level k. These were ordered so that if a node had a particular color at a vertex, then it remained the same color in the sub tree.

**Solution candidates:**

For knapsack, a list of the items placed in the knapsack will be sufficient. Chromatic numbering involves a list of the colors for each vertex in the graph. Other than, it is a little more complex since we use partial solutions in our search, so we must indicate vertices yet to be colored in the list. A necessary rule to be followed in essential solution spaces for branch and bound algorithms as follows. If a solution tree vertex is not part of a feasible solution, then the sub tree for which it is the root cannot contain any feasible solutions. This rule assures that if we cut off search at a vertex due to impracticality, then we have not unnoticed any optimum solutions.

**Lower bound at a vertex:**

The Smallest value of the intention function for any node of the sub tree rooted at the vertex.

**Upper bound at a vertex:**

The largest value of the intention function for any node of the sub tree rooted at the vertex.

For chromatic number we used the number of colors for the lower bound of a partial or complete solution. The lower bound for knapsack vertices was the current load, while the upper bound was the possible weight of the knapsack in the sub tree. Branch-and-bound may furthermore be a base of various heuristics. For instance, one may desire to prevent branching while the gap among the upper and lower bounds becomes smaller than a certain threshold. This is act as a solution and can greatly reduce the computations required. This type of solution is particularly applicable when the cost function used is noisy or is the result of statistical estimates and so is not known exactly but rather only known to lie within a range of values with a specific probability. The main advantage of Branch & Bound algorithm is it finds an optimal solution (if the problem is of limited size and enumeration can be done in reasonable time).

 

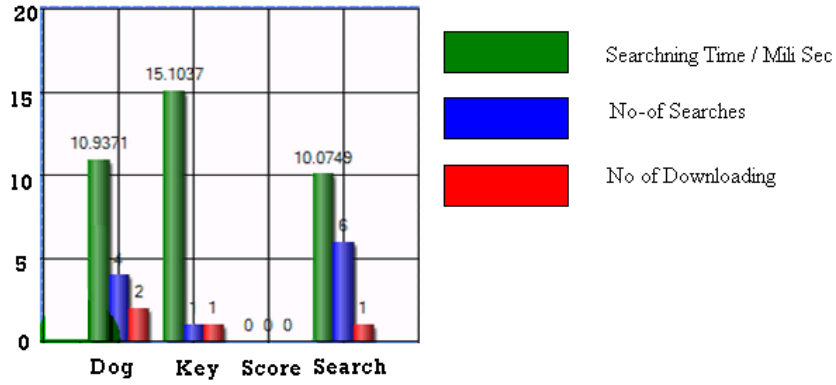


**Figure: Reasonable Searching time**

### 4.3. Iterative Algorithm

In this section, define efficient iterative approach to the Determinization problem for the set-based metric. These are methods which compute a sequence of progressively accurate iterates to approximate the solution. We need such methods for solving many large linear systems. Sometimes the matrix is too large to be stored in the computer memory, making a direct method too difficult to use.It first determinizing all objects, using a query unaware algorithm, such as threshold-based or random algorithm, followed by an iterative procedure. The algorithm picks one object Oi. It then treats other objects O\ {Oi} as already determinate, and determinisms Oi again such that the overall expected F-measure E (Fα (O, Q)) is maximized. In this way, E (Fα (O, Q)) will either increase or remain the same in each iteration. For every |O| iterations, the algorithm checks the value of E (Fα (O, Q)), and stops if the increase of the value since last check-point is less than certain threshold. The main question is how to, in each iteration, determinizing the chosen object O such that the overall expected F-measure is maximized.

### A. Determinizing Individual Object

Having updated negative and positive F-measures for all queries, we are left with the problem of how to determinizing the chosen object *Oi* such that the overall expected F-measure of the query workload is maximized. This problem is virtually the same as the EDCM problem, where the goal is to determinizing an object such that the overall expected cost of a query workload is minimized. Thus, we can employ the Branch. More specifically, the BB algorithm can be applied with small modifications:

Since the original BB algorithm is to find the minimum, while our task here is to find the maximum, the BB algorithm needs to be changed in a symmetric fashion (for example, exchanging the ways to compute lower bound and upper bound). The main structure of the algorithm stays unchanged.

### B. Picking Next Object

Another question is how to pick next object to determinizing. One strategy is for each object *O*, *O* to look ahead the overall expected F-measure resulted from choosing this object. The object that leads to the maximum value is chosen as the object to determinizing. This strategy, though ensuring maximum increase of the overall expected F measure in each iteration, will add a linear factor to the overall complexity of the algorithm. Thus, it is not suitable for large datasets. Another strategy is to simply loop over the dataset or choose objects in a random order. Although this strategy is not necessarily the best one in terms of leading the algorithm towards convergence, it is a constant operation. We thus employ the second strategy.

### C. Other Set-Based Metrics

While we illustrate the algorithm using F-measure, the iterative framework can also be used for optimizing other set-based metrics, such as Jaccard distance and Symmetric distance.

## CONCLUSION

Hence, from this paper we have considered problem of determinizing uncertain objects in order to organize and store such data in already existing systems example Flickr which only accepts deterministic value. Our aim is to produce a deterministic depiction that optimizes the quality of answers to queries/triggers that execute over the deterministic data representation .As in future work, we plan to perform project on efficient determinization algorithms that are orders of scale faster than the enumeration based best solution but achieves almost the same excellence as the optimal solution and search determinization techniques as per the application context, wherein users are also involved in retrieving objects in a ranked order.

 

**REFERENCES**

[1] Jie Xu, Sharad Mehrotra," Query Aware Determinization of Uncertain Objects" ,IEEE Transactions on knowledge and data engineering, VOL. 27, NO. 1, January 2015.

[2] J. Li and J. Wang, "Automatic linguistic indexing of pictures by a statistical modeling approach," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 9, pp. 1075–1088, Sept. 2003.

[3] C. Wangand, F. Jing, L. Zhang, and H. Zhang, "Image annotation refinement using random walk with restarts," in *Proc. 14th Annu. ACM Int. Conf. Multimedia*, New York, NY, USA, 2006.

[4] B. Minescu, G. Damnati, F. Bechet, and R. de Mori, "Conditional use of word lattices, confusion networks and 1-best string hypotheses in a sequential interpretation strategy," in *Proc. ICASSP*, 2007.

[5] Jian Pei, Ming Hua," Query Answering Techniques on Uncertain and Probabilistic Data" In *VLDB*, pages 1151-1154, 2006.

[6] Umesh Gorela1, Bidita Hazarika2, Abhinesh Tiwari3, Priti Mithari," Survey on Query Aware Strategy for Determining Uncertain Probabilistic Data", in (IJSETR), Volume 4, Issue 10, October 2015 3510

[7] R. Nuray-Turan, D. V. Kalashnikov, S. Mehrotra, and Y. Yu, "Attribute and object selection queries on objects with probabilistic attributes," *ACM Trans. Database Syst.*, vol. 37, no. 1, Article 3, Feb. 2012.

[8] V. Jojic, S. Gould, and D. Koller, "Accelerated dual decompositionfor MAP inference," in *Proc. 27th ICML*, Haifa, Israel, 2010.

[9] D. Sontag, D. K. Choe, and Y. Li, "Efficiently searching for frustrated cycles in map inference," in *Proc. 28th Conf. UAI*, 2012.

[10] I. Bordino, C. Castillo, D. Donato, and A. Gionis, "Query similarity by projecting the query-flow graph," in *Proc. 33rd Int. ACM SIGIR*, Geneva, Switzerland, 2010.

[11] P.Jhancy, K.Lakshmi ,Dr.S.Prem Kumar," Query Aware Determinization of Uncertain Objects" in ijcert Volume 2, Issue 12, December-2015, pp. 904-907

[12] R. Nuray-Turan, D. V. Kalashnikov, S. Mehrotra, and Y. Yu,"Attribute and object selection queries on objects with probabilistic attributes," *ACM Trans. Database Syst.*, vol. 37, no. 1, Article 3, Feb. 2012.

[13] B. Sigurbjornsson and R. V. Zwol, "Flickr tag recommendation based on collective knowledge," in *Proc. 17th Int. Conf. WWW*, New York, NY, USA, 2008.

[14] A. Rae, B. Sigurbjornsson, and R. V. Zwol, "Improving tag recommendation using social networks," in *Proc. RIAO*, Paris, France, 2010.

[15] D. Carmel *et al.*, "Static index pruning for information retrieval systems," in *Proc. 24th Annu. Int. ACM SIGIR*, New Orleans, LA, USA, 2001.

[16] Milind K.Patil,Prof. Yogesh S.Patil,Prof. Dinesh D.Patil," Query Aware Determinization of Uncertain Objects", in IJARCSSE ,Volume 6, Issue 1**,** January 2016

 