ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 11, November 2018

Continuous Top-K Monitoring on Document Streams

Rahul V. Mundhe[1], Prof. K. B. Manwade[2]

PG Student, Dept. of Comp. Engg. Ashokrao Mane Group of Institutions, Vatharla tarf Vadgaon, Kolhapur., Maharashtra, India[1]

Professor, Dept. of Comp. Engg. Ashokrao Mane Group of Institutions, Vatharla tarf Vadgaon, Kolhapur., Maharashtra, India[2]

ABSTRACT: In current years, there has been a strong development of incorporating results from structured data source into keyword based web hunt systems such as Amazon or Google. When presenting ordered data, facets are a potent tool for navigate, refinement, and group the results. For a given structured data source, a fundamental problem in supporting faceted search is finding an ordered assortment of attributes and standards that will occupy the facets. This creates two sets of challenges: First, because of the restricted screen real estate, it is vital that the top a small number of facets best match the probable user intent. Second, the massive scale of existing data to engines like Amazon or Google demands an automated unsupervised resolution. In this work we proposed efficient evaluation techniques of continuous top-k queries over text and feedback streams featuring generalized scoring functions which capture dynamic ranking aspects. As a first contribution, we generalize state of the art continuous top-k query models, by introducing a general family of non-homogeneous scoring functions combining query-independent item importance with query-dependent content relevance and continuous score decay reflecting information freshness. Our second contribution consists in the definition and implementation of efficient in-memory data structures for indexing and evaluating this new family of continuous top-k queries. Our experiments show that our solution is scalable and outperforms other existing state of the art solutions, when restricted to homogeneous functions. Going a step further, in the second part of this thesis we consider the problem of incorporating dynamic feedback signals to the original scoring function and propose a new general realtime query evaluation framework with a family of new algorithms for efficiently processing continuous top-k queries with dynamic feedback scores in a real-time web context.

I. INTRODUCTION

In context, a central server monitors a document stream and hosts CTQDs from various users. Each CTQD specifies a set of keywords, as explicitly given by the issuing user or extracted from user's online behavior. The task of the server is to continuously refresh - for every CTQD - the top-k most relevant documents to the keywords, as new documents stream in and old ones become too stale to be of interest. Stock news notifications are an application domain for CTQDs. The investment decisions of a stock broker are very sensitive to news about the stocks in user's portfolio. To enable timely decisions, presenting user with the most relevant news as soon as they become available is key to the success of the notification system. Similar application scan be found in monitoring live Web content, such as RSS/news feeds, blog entries, posts on social media, etc. Widely available notification systems, such as Google Alerts (google.com/alerts) and Yahoo! Alerts(alerts.yahoo.com), attest to the significance of these applications. On the other hand, these systems either work in a semi-offline manner by delivering periodic updates (e.g., daily) or allow for coarse filtering based only on general topics, rather than set of specific keywords. Another application domain for CTQDs are micro blog real-time search services e.g. Twitter Search, Friend Feed etc.

In order to filter the published information items, users generally issue catchphrase addresses that can either be clearly evaluated by the crucial web lists over the appropriation focal point of secretly set away things or submitted to a

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 11, November 2018

disturbing organization (e.g., Google Alert, Yahoo Alert) always illuminating customers about as of late circulated things planning their filtering criteria. In the two circumstances, the amount of filtered things can rapidly end up being too high to be in any way exploitable by the customer in time. Along these lines, filtering must be connected by appropriate thing packing and situating limits. Situating includes in assessing of how basic information things are as demonstrated by some quality criteria (e.g., freshness, master, et cetera.), and moreover, that they are so imperative to isolating conditions communicated in customer participations. For clustering, information things (news articles, customer posts) are ordinarily dealt with separated into imperative gatherings sharing fundamental properties (e.g., news articles referring to the same real-world event, called stories).

Advanced solutions also assess both, the thing story and the thing substance amid catchphrase based organizing through adequate scoring limits. To pass on as severely dislike information as could sensibly be normal, business news accumulation structures rely upon a blend of time spoil and sliding time window frameworks. For example, Google News keeps up a window of articles over the latest 30 days regardless of the way that the best k comes to fruition showed without questions are now and again conveyed over multi day sooner. Such articles could be returned just in circumstances when request don't restore various related late results. Additionally, the helpfulness of business web alerts systems is reminiscent of predictable best k printed request appraisal. For example, the Google Alerts advantage once in a while evaluates freely every submitted inquiry on the Google News engines according to a predefined stimulating system, recoups the best k comes to fruition without a second's pause and illuminates customers for as of late dispersed things. Changing a consistent request to a movement of discontinuously executed delineation questions gains bona fide repressions. For expansive quantities of client questions and high entry rates it is viable difficult to over and again assess all inquiries at all new data things. Hence, business frameworks more often than not diminish the recurrence of depiction question evaluation and thus important news may be missed.

A crucial factor in the formalization of the continuous query evaluation problem and the solutions proposed, is the underlying filtering technique, i.e. the conditions under which an item from the stream is inserted into a query’s result set. After presenting the most important filtering techniques, we focus on the continuous top-k ranking model. In this model, the definition of a scoring function assigning a score to each query-item pair is also required. The definition of an adequate scoring function depends both on the quality of information streams, but also on the expected value of obtained results in concrete applications contexts. In order to propose a generic solution to the continuous top-k query evaluation problem, we study several ranking parameters proposed to accommodate specific application needs and abstract from these a generalized function capturing both static and dynamic aspects of information arriving in streams.

II. LITERATURE SURVEY

A crucial factor in the formalization of the continuous query evaluation problem and the solutions proposed, is the underlying filtering technique, i.e. the conditions under which an item from the stream is inserted into a query's result set. After presenting the most important filtering techniques, we focus on the continuous top-k ranking model. In this model, the definition of a scoring function assigning a score to each query-item pair is also required. The definition of an adequate scoring function depends both on the quality of information streams, but also on the expected value of obtained results in concrete applications contexts. In order to propose a generic solution to the continuous top-k query evaluation problem, the study several ranking parameters proposed to accommodate specific application needs and abstract from these a generalized function capturing both static and dynamic aspects of information arriving in streams.

According to Leong Hou U, Junjie Zhang, Kyriakos Mouratidis, and Ye Li, “Continuous Top-k Monitoring on Document Streams [1] system proposed a interested in efficient evaluation techniques of continuous top-k queries over text and feedback streams featuring generalized scoring functions which capture dynamic ranking aspects. As a first contribution, this generalize state of the art continuous top-k query models, by introducing a general family of non-homogeneous scoring functions combining query-independent item importance with query-dependent content relevance and continuous score decay reflecting information freshness. This second contribution consists in the definition and implementation of efficient in-memory data structures for indexing and evaluating this new family of continuous top-k queries. Systems experiments show that solution is scalable and outperforms other existing state of

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 11, November 2018

the art solutions, when restricted to homogeneous functions. Going a step further, in the second part of this work system consider the problem of incorporating dynamic feedback signals to the original scoring function and propose a new general real time query evaluation framework with a family of new algorithms for efficiently processing continuous top-k queries with dynamic feedback scores in a real-time web context. Finally, putting together the outcomes of these works, system present Meows Reader, a real-time news ranking and filtering prototype which illustrates how a general class of continuous top-k queries offers a suitable abstraction for modeling and implementing continuous online information filtering applications combining keyword search and real-time web activity.

Oren E., Delbru R. and Decker S., "Extending faceted navigation for RDF data [2]" The paper analyzes the predicate balance, object cardinality and predicate frequency to rank facets. The predicate balance is referred to as the balance of the faceted navigation tree composed of faceted attributes If the predicate frequency of a faceted attribute is low, when a user selects a value of this attribute, only a small number of data items are affected.

According to [3] Giunchiglia F., Dutta B., and Maltese V., "Faceted Lightweight Ontologies, in Conceptual Modeling: Foundations and Applications" The paper proposes a lightweight ontology which has a rooted tree structure where each node is associated with a natural language label. This model gives the definitions of category ontology, lightweight ontology, and faceted lightweight ontology, but does not provide interactive operations.

According to Stoica E., Hearst M.A., and Richardson M., [4] "Automating Creation of Hierarchical Faceted Metadata Structures" Castanet algorithm presented can automatically generate Hierarchical Structure of faceted metadata from textual descriptions of data items by exploiting the "is-a"relationship in WordNet. The algorithm selects candidate terms from textual descriptions of data items. This algorithm solely depends on WordNet to obtain the "is-a" relationship and therefore has limited scalability.

Anick P.G. and Tipirneni S., "The paraphrase search assistant: terminological feedback for iterative information seeking" in [5].System proposed a facet term extraction algorithm based on the lexical dispersion of words in text. The algorithm consists of two stages. In the indexing stage documents are parsed so the lexical compounds can be extracted. In the querying stage the compounds appearing in the top n documents of a ranked result list are used to compute the lexical dispersion of each term occurring within these compounds. The disadvantage of this algorithm is that the extraction of facet terms depends on the specific lexical structure and therefore can be hardly extended to new domains.

Ling X. et al., "Mining multi-faceted overviews of arbitrary topics in a text collection" [6] proposed a paper proposes a two-stage probabilistic method to extract facet terms based on topic model. Given the original keywords from a user, this method first applies a bootstrapping algorithm to the document collection to get more correlated terms. Probabilistic mixture models are applied to these expanded terms to estimate the term distribution of every facet.

Zeng H.-J., et al., "Learning to cluster web search results" [7] the hierarchical relation extraction algorithm proposed by takes the search keywords as the root node. A Support Vector Machine regression model is learned from human labeled training data to search facet terms in the search results. The final groups are generated by merging the initial candidate groups. This method cannot generate multidimensional taxonomies. Also the extracted relations may be neither " is-a " nor "part-of ".

According to Chen J, and Li Q. [8], "Concept Hierarchy Construction by Combining Spectral Clustering and Subsumption Estimation. The paper presented a method of automatic term hierarchies acquisition based on subsumption estimation and spectral clustering. First, each term is considered as a vertex in an undirected weighted graph. The problem of hierarchical relation construction is then modeled as a modified graph-partitioning problem and is solved by spectral clustering methods. This method can extract facet terms based on compound words such as "probabilistic clustering," but the hierarchical taxonomies may be significantly different from the manually obtained taxonomies.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 11, November 2018

Xing D. et al. [9], "Deep classifier: automatically categorizing search results into large-scale hierarchies" The paper proposes a novel algorithm that groups search results into detailed hierarchical categories by pruning online web directories. The web directories, such as Yahoo! Directory and the Open Directory Project, are always too complex to construct an appropriate taxonomy directly. Therefore, this algorithm first searches for the web directory by using keywords from a user query. Then a hierarchical structure related to user keywords is generated from the search results. Compared to the online web directories, the taxonomic structure contains fewer nodes and is closely related to the user keywords.

According to Zhicheng Dou and ZhengbaoJiang,"Automatically Mining Facets for Queries from Their Search Results [10] proposed a system Query facets provide interesting and useful knowledge about a query and thus can be used to improve search experiences in many ways. First, this system can display query facet together with the original search results in an appropriate way. Thus, users can understand some important facets of a query without browsing tens of pages. For example, a user could learn different brands and categories of watches. This is system can also implement a faceted search based on query facets. User can clarify their specific intent by selecting facet items. Then search results could be restricted to the documents that are relevant to the items. These multiple groups of query facets are in particular useful for vague or ambiguous queries, such as "apple". Accordingcould show the products of Apple Inc. in one facet and different types of the fruit apple in another. Second, query facets may provide direct information or instant answers that users are seeking. For example, for the query "lost season 5", all episode titles are shown in one facet and main actors are shown in another. In this case, displaying query facets can save browsing time. Third, query facets may also be used to improve the diversity of the ten blue links. Accordingcan re-rank search results to avoid showing the pages that are near-duplicated in query facets at the top. Query facets also contain structured knowledge covered by or related to the input keywords of a query, and thus they can be used in many other fields besides traditional web search, such as semantic search or entity search. There has been a lot of recent work on automatically building knowledge ontology on the Web . Query facets can become a possible data source for this. IT shows supervised method based on a graphical model to recognize query facets from the noisy facet candidate lists extracted from the top ranked search results. This proposed two algorithms for approximate inference on the graphical model. Accordingdesigned a new evaluation metric for this task to combine recall and precision of facet terms with grouping quality. Experimental results showed that the supervised method significantly out-performs other unsupervised methods, suggesting that query facet extraction can be effectively learned.

Preference based queries rank the items of a database according to the significance of their attributes. In addition to hard constraint (eg price<100) the result must satisfy some additional specific properties related to the attribute values associated with each tuple [11]. The preference queries are broadly classified into top k query, skyline query and top k dominating query. In a top k query user defined ranking function is used which assign a value to each tuple. It bound the output size. If two or more tuples have the matching score then all these tuples added or use a tie-breaking criterion. The most important limitation of the top k query is that a user defined ranking function is used.

The skyline consists of the tuples not dominated by other tuple. The skyline computation has received considerable attention in relational database but the existing algorithms of the skyline computation are inapplicable to stream application. The irst reason is they assume static data that are stored in the disk. The second reason is they focus on "one-time" execution that returns a single skyline .The third reason is they aim at reducing the i/o overhead. The skyline computation in streaming environment is performed by using a sliding window. Y.Tao [12] proposes algorithms that continuously monitor the incoming data and maintain the skyline incrementally. These algorithms utilize several interesting properties of stream skylines to improve space/time efficiency by expunging data from the system as early as possible. The skyline computation in data stream system that consider only the tuples that arrived in a sliding window covering the W most recent timestamps, where W is a system parameter called the window length.

Consider a dataset D and a preference function f, a top-k query contains the k tuples with the highest scores according to f. The problem is well-studied in conventional databases but the existing methods are incompatible to highly dynamic environments involving numerous long running queries. K.Mouratidis et.al [13] proposed algorithms for the continuous monitoring of top-k queries over a fixed-size window W. The sliding window size can be articulated either

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 11, November 2018

in terms of the count based or time units. To achieve real-time query estimation the suitable tuples are stored in main memory. The valid records are arranged by using grid based index schema. Grid based index preserves a book keeping structure.

The top k query is important for several online applications such as communication and sensor networks, stock market trading, and profile based marketing etc. Top k query evaluation can be performed by using the count based and time based sliding window. The count based window W contains the most recent items and the time based window W contains all tuples that arrived within a fixed time instances. The task of the query processor is to constantly report the top k set of every monitoring query among the valid data. When a query q first arrives at the system, its result is computed by the top-k computation module which searches the minimum number of cells that may contain result records. Two algorithms are used for the continuous evaluation of Top k monitoring. The algorithms are Top k Monitoring Algorithms (TMA) and Skyband Monitoring Algorithm (SMA).The Top k Monitoring Algorithm re-computes the answer of a query whenever some of the current top-k points expire. The Skyband Monitoring Algorithm (SMA) partially precomputes future results by reducing the problem to sky band maintenance over a subset of the valid records.

Skyline query is one of the most widely used preference query. The result of a skyline query is composed of the points that are not dominated by any other point. A dominant tuple is defined as tuple tx dominates another tuple ty ,if and only if tx is smaller than or equal to ty.in all dimensions [15].The dominance relationship counts on the semantics of each attribute. In some cases, small values are preferred (e.g., distance) but in other cases large values are suitable (e.g., quality). The key advantage of the skyline query is that it does not require any user-defined information or parameter. The limitation of the skyline query is that it does not bind the output size and therefore in extreme cases it is possible that all tuples be the part of skyline tuples.

III. SYSTEM DESIGN

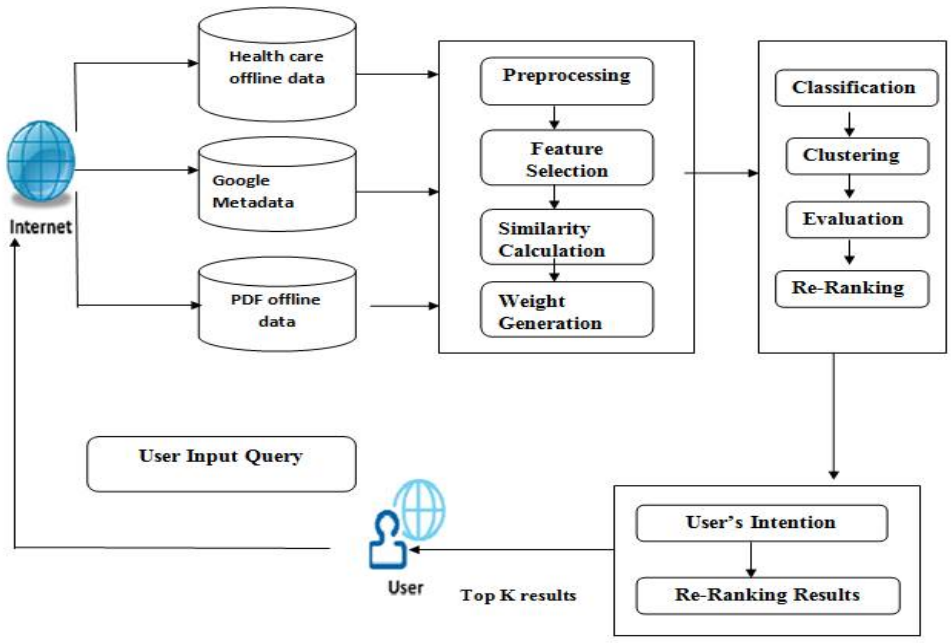

Figure 1: Block Diagram of proposedsystem

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 11, November 2018

User Authentication and query submission Module:
This module can be done user authentication module with user can submit the search query with this form.
Document retrieval module:
This second module can express the document retrieval approach from different web pages and extract the data.
List extraction module
This phase can work with extract the list from each page and store into database simultaneously.
List weighting module
In that phase list weighting has done suing vector base cosine similarity algorithm. Na process each list with specific weight.
Clustering module
This module can done the list clustering approach, it can make the multiple clusters of available lists.
News ranking and analysis module
Finally system provide the ranking of new uploaded documents uploaded by user with comparative analysis graphs of with some existing systems.

Algorithm Design

Incoming Document Streaming Algorithm

Step 1: set all cursors ci to the beginning of their lists

Step 2: whilethe relevant lists are not exhausted do

Step 3: if sum of all non-exhausted lists <=1 then

Step 4: return

Step 5: decide execution order of relevant lists

Step 6: for i = 1 to m do

Step 7: set if UB(i) > 1 then

Step 8: advance c1, c2,…., ci- until their IDs>= ci

Step 9: if c1, c2 , …. ci point at same query q then

Step 10: ifS(q, d) > 1 then

Step 11: insert d into the top-k result of q

Step 12: reflect new Sk(q) to wj values of q

Step 13: advance c1, c2 , …. ci to next position

Step 14: break (go to line 2)

The input set all cursors to list.And list are not exhausted check sum of all non-exhausted list less than 1.Condition true order of relevant order list return.The insert d into top k list result.Finally Top-K result return.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 11, November 2018

Document retrieval Algorithm

Input: Users query as Q , Network Connection N;

Output: result from relevancy calculation top k pages base on Q.

Step 1: User provide the Q to system.

Step 2: if (N!=Null)

Process

Read each attribute A from ith Row in D

Res[i]=Calcsim(Q,A)

Else No connection

Step 3: For each(k to Res)

Step 4: Arraylist Objarray to bind Q to Res[i] or k

Step 5: Return to users Objarray

Step 6: Display Objarray

System performs a novel diversity-aware service ranking algorithm to find the optimal top-k Web services based on a proposed comprehensive ranking measure. It is re-lasted work on service recommendation in these three categorists, and on diversity-based ranking algorithms.

Weight Calculation Algorithm

Input: Query generated from user Q, each retrieved list L from webpage.

Output: Each list with weight.

Step 1: Read each row R from Data List L

Step 2: for each (Column c from R)

Step 3: Apply formula (1) on c and Q

Step 4: Score=Calc(c, Q)

Step 5: calculate relevancy score for attribute list.

Step 6: assign each Row to current weight

Step 7: Categorize all instances

Step 8: end for end procedure

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 11, November 2018

Here systems have to find similarity of two lists: ā= (a1, a2, a3 …) and Б= (b1, b2, b3 …) where an and bn are the components of the vector

(Features of the document or values for each word of the comment) and the is the dimension of the vectors:

Clustering Algorithm

Input: input list of group which contains the list item LI, Facet list FL, var weight

Output: Classify all the items into different clusters

Step 1: For each (item I to LI)

Step 2: For each (item j to FL)

Step 3: Define weight as double [], Hashmap<double, string>

Step 4: weight[i]=Similairty(LI[i],Fl[j])

Step 5: put into hashmap<weight[i], LI[i]Fl[j]>

End for

End for

Step 6: Sort Hashmap with desc order

Step 7: Select first value from Hasmap

Step 8: Move LI[i] to Fl[j]

Here system Classify all the items into different clusters. Groups to use and randomly initialize their respective center points. Each data point is classified by computing the distance between that point and each group center, and then classifying the point to be in the group whose center is closest to it.

Ranking Algorithm

Input :Hashmap<double, string>,

Output : URL list with weight

Step 1: Read each (k to Hashmap)

Step 2: evaluate each Li=∑_(k=0)^n(Hashmap[k])

Step 3: Display Li with maximum weight

Step 4: end for

Step 5: all Li asec order

System performs a ranking algorithm to find the url list with weight. All list with maximum weight display in asec order.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 11, November 2018

IV. RESULTS AND DISCUSSION

In experiment 1 we have collected some existing approaches time complexity with proposed. The time required for data extraction for different search engines with this system The below table 10.1 shows the time complexity comparison of keyword base object search [3], Concept Hierarchy Construction by Combining Spectral Clustering [5], semantic search using RDF data [6] etc.

Table 1: Comparative analysis between proposed vs Existing approaches (Accuracy)

Method	Accuracy in %	False ratio in %
Keyword base object search [3]	86.15	13.85
Hierarchy base spatial clustering [5]	84.16	15.84
Semantic base RDF search [6]	82.77	17.23
Facet search using TF-IDF [1]	90.15	9.85
Top-K Monitoring using VCS (Proposed)	95.02	4.98

In the second experiment, it given some theoretical as well estimated results. With the comparison of above given three system how TOP-k is better, the below graph as well table shown in deafly. The below table 6.4the time required for searching in specific query with proposed as well as existing approach.

Table 2 : Comparative analysis between proposed V/S existing approaches (Time complexity)

Method	5d	10d	15d	20d
Keyword base object search [3]	246	488	723	975
Hierarchy base spatial clustering [5]	310	602	923	1178
Semantic base RDF search [6]	840	1520	2310	3125
Facet search using TF-IDF [1]	580	952	1533	2701
Top-K Monitoring using VCS (Proposed	235	400	650	800

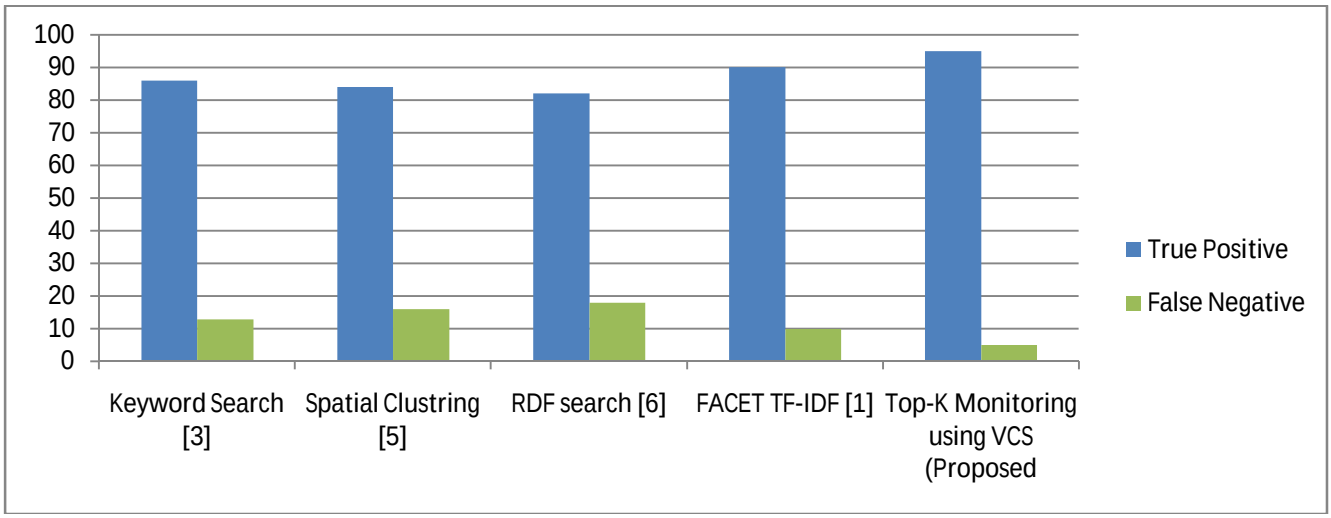

Figure 2: Accuracy between proposed vs Existing systems

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 11, November 2018

V. CONCLUSION

In this work we analyzed the different approaches, first, the temporal prevalence of a particular topic in the news media is a factor of importance, and can be considered the media focus of a topic. Second, the temporal prevalence of the topic in social media indicates its user attention. Last, the interaction between the social media users who mention this topic indicates the strength of the community discussing it, and can be regarded as the user interaction toward the topic. We propose an unsupervised framework approach which identifies any type of search query, and then ranks them by relevance using their weight as well as their number of visit by users. For the implementation phase we need to work different type of dataset, which can read like buffer pool. It also need to evaluate the operation cost query as well as data updation time on GUI.The main aim of this survey is to analysis of the methods used by preferences queries to find out the top k result. The sliding window technique is used for continuous monitoring of data streams. In top k query the continuous monitoring of the data is performed by using Top k Monitoring Algorithm and Skyband Monitoring Algorithm. In skyline query continuous monitoring of skyline points is performed by lazy method and eager method. In top k dominating query the advanced algorithm and approximate algorithms are used to return the top k result. Finally all preference queries methods are compared and concluded that top k dominating query with advanced algorithm show the best performance.

FUTURE WORK

To implement a system with different dataset and measure the time complexity with different existing algorithms in distributed environment.The system also address on personalize search on user feedback sessions as well as recommendation base on user point of interest with database security is the interesting part of system.

REFERENCES

[1] Leong Hou U, Junjie Zhang, KyriakosMouratidis, and Ye Li, "Continuous Top-k Monitoring on Document Streams" in IEEE Transactions on Knowledge and Data Engineering,Year: 2017, Volume: 29, Issue: 5, pp.1-14
[2] Oren E., Delbru R. and Decker S., "Extending faceted navigation for RDF data", in Proceedings of the 5th International Semantic Web Conference (ISWC), 2006. p. 559-572, 1001.
[3] Giunchiglia F., Dutta B., and Maltese V., "Faceted Lightweight Ontologies, in Conceptual Modeling: Foundations and Applications", A. Borgida, et al., Editors 2009, Springer Berlin /Heidelberg. p. 36-51.
[4] Stoica E., Hearst M.A., and Richardson M., "Automating Creation of Hierarchical Faceted Metadata Structures"in Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL), 2007: p.244-251.
[5] Anick P.G. and Tipirneni S., "The paraphrase search assistant: terminological feedback for iterative information seeking" in Proceedings of the 22nd annual International ACM SIGIR Conference on Research and Development in Information Retrieval. 1999, ACM: Berkeley,California, United States. p. 153-159.
[6] Ling X. et al., "Mining multi-faceted overviews of arbitrary topics in a text collection" in Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. 2008, ACM: Las Vegas, Nevada, USA. p. 497-505.
[7] Zeng H.-J., et al., "Learning to cluster web search results" in Proceedings of the 27th annual international ACM SIGIR conference on Research and Development in Information Retrieval.2004, ACM: Sheffield, UK. p. 210-217.
[8] Chen J, and Li Q., "Concept Hierarchy Construction by Combining Spectral Clustering and Subsumption Estimation" in Web Information Systems – WISE 2006, K. Aberer, et al., Editors.2006, Springer Berlin / Heidelberg. p. 199-209.
[9] Xing D. et al., "Deep classifier: automatically categorizing search results into large-scale hierarchies" in Proceedings of the international conference on Web search and web data mining 2008, ACM: Palo Alto, California, USA. p. 139-148.
[10] Zhicheng Dou and ZhengbaoJiang,"Automatically Mining Facets for Queries from Their Search Results"in IEEE 2016,VOL. 28, NO. 2, p. 385-397
[11] W. Kiessling, "Foundations of Preferences in Database Systems,"Proc. 28th Int'l Conf. Very Large Data Bases (VLDB), pp. 311-322,2002.
[12] Y. Tao and D. Papadias, "Maintaining Sliding Window Skylines on Data Streams," IEEE Trans. Knowledge and Data Eng., vol. 18, no. 3, pp. 377- 391, Mar. 2006.
[13] K. Mouratidis and S. Bakiras, , "Continuous Monitoring of Top-k Queries over Sliding Windows," Proc. ACMSIGMOD Conf. Management of Data ,pp. 635-646, 2006.
[14] M. Kontaki, A.N. Papadopoulos, and Y. Manolopoulos, "Continuous Top-k Dominating Queries in Subspaces, " Proc. Panhellenic Conf. Informatics (PCI), pp. 675-689, 2008.
[15] D. Papadias, Y. Tao, G. Fu, and B. Seeger, "Progressive Skyline Computation in Database Systems" ACM Trans. Database Systems,vol. 30, no. 1, pp. 41-82, 2005.