EFFICIENT MANAGEMENT OF TOP-K QUERIES OVER UNCERTAIN DATA STREAMS WITH DYNAMIC SLIDING WINDOW MODEL

By

RAJA AZHAN SYAH BIN RAJA WAHAB

Thesis Submitted to the School of Graduate Studies, Universiti Putra Malaysia, in Fulfillment of the Requirements for the Degree of Doctor of Philosophy

July 2024

FSKTM 2024 12

Abstract of thesis presented to the Senate of Universiti Putra Malaysia in fulfilment of the requirements for the degree of Doctor of Philosophy

EFFICIENT MANAGEMENT OF TOP-K QUERIES OVER UNCERTAIN DATA STREAMS WITH DYNAMIC SLIDING WINDOW MODEL

By

RAJA AZHAN SYAH BIN RAJA WAHAB

July 2024

Chair : Siti Nurulain binti Mohd Rum, PhD
Faculty : Computer Science and Information Technology

Today, the advancement of information technology has led to a growing need for continuous processing of significant events, such as enhanced methods for monitoring road speed and mobile computing. The Uncertain Data Stream (*UDS*) utilized for query processing can provide challenges in many technical contexts owing to its inherent inconsistency, ambiguity, and time delay in interpreting information. The large amount of data generated and frequent changes in a short time make conventional processing methods insufficient. The main issues are minimizing redundant scans of the whole data set, improving uncertainty computation, and only processing the most recent tuple items. In *UDS*, the number of possible world instances grows exponentially, and understanding what is required to achieve Top-*k* query processing in the shortest possible time can be extremely challenging. However, there is a need

to increase the number of studies investigating the issue of UDS using the Sliding Window Model (*SWM*). An inefficient approach to processing continuous queries on *UDS* with uncertainty over the *SWM* increased the complexity of semantic trade-offs between answering maximum probability and high-scoring result sets. Current research on tackling uncertainty revolves around creating specifically tailored algorithms that can operate in the presence of value uncertainty using both a count-based and a time-based approach. This study aims to propose a framework for processing Top-k queries in *UDS,* where the focus is on leveraging the efficiency of the *SWM*, achieved through the *SWMTop-kDelta* algorithm. After establishing this model's rules and probability theory, a method was designed to support the Top-*k* processing algorithm over the *SWM* until the Top-*k* potential candidates expired. This study also provides an overview of an improved optimization method for tackling computational redundancy in the context of *SWM* and Top-*k* query computation. This method reduces computational costs by efficiently handling the insertion and exit policy for the appropriate tuple candidates within a specified window frame. The experiments in this study compare the *SWMTop-kDelta* algorithm with two previous researchers and two baseline approach algorithms to evaluate their effectiveness. The algorithm development combines the frameworks from Phases 1 to 3, evaluating real and synthetic datasets. It assesses efficiency by comparing the number of possible worlds and processing times. The experiment was conducted in triplicate and recorded the mean value of these iterations. As the data set size increases, *SWMTop-kDelta* consistently performs well, regardless of the data set size and the measurement of the number parameter k. Even if the initial

improvement is only slight, performance can consistently improve by making certain adjustments, such as increasing the number of window segmentations, decreasing the window size, reducing the number of queries, and adjusting the probability threshold (d) more frequently. It demonstrates a significant improvement of 30%–90% compared to other methods, thanks to its consistent performance and strong scalability. This study effort will make a valuable contribution to the field of Top-k computational query processing.

Keywords: Top-k, Uncertain Data Stream, Sliding Window Model, Tuple Items, Possible World

SDG: GOAL 4: Quality Education

Abstrak tesis yang dikemukakan kepada Senat Universiti Putra Malaysia sebagai memenuhi keperluan untuk ijazah Doktor Falsafah

PENGURUSAN PERTANYAAN *TOP-K* YANG CEKAP KE ATAS ALIRAN DATA TIDAK PASTI DENGAN MODEL DINAMIK TETINGKAP GELONGSOR

Oleh

RAJA AZHAN SYAH BIN RAJA WAHAB

Julai 2024

Pengerusi : Siti Nurulain binti Mohd Rum, PhD
Fakulti : Sains Komputer dan Teknologi Maklumat

Pada hari ini, kemajuan teknologi maklumat telah membawa kepada keperluan yang semakin meningkat untuk pemprosesan berterusan peristiwa penting, seperti kaedah yang dipertingkatkan untuk memantau kelajuan jalan raya dan pengkomputeran mudah alih. Aliran Data Tidak Pasti (*ADTP*) yang digunakan untuk pemprosesan bahasa pangkalan data boleh memberikan cabaran dalam banyak konteks teknikal disebabkan ketidakkonsistenan yang wujud, kekaburan dan kelewatan masa dalam mentafsir maklumat. Jumlah besar data yang dijana dan perubahan yang kerap dalam masa yang singkat menjadikan kaedah pemprosesan konvensional tidak mencukupi. Isu utama ialah meminimumkan imbasan berlebihan bagi keseluruhan set data, menambah baik komputasi ketidakpastian dan hanya memproses item tupel

terbaharu. Dalam *ADTP*, bilangan kejadian dunia yang mungkin berkembang dengan pesat, dan memahami perkara yang diperlukan untuk mencapai pemprosesan pertanyaan *Top-k* dalam masa yang sesingkat mungkin boleh menjadi sangat mencabar. Pertanyaan *Top-k* direka untuk menyusun item tuple dalam masa yang minimum dengan cekap. Namun begitu, bilangan kajian yang dijalankan untuk menyiasat masalah *ADTP* yang menggunakan Model Tetingkap Gelongsor (*MTG*) perlu ditambahbaik. Pendekatan yang tidak cekap untuk memproses pertanyaan berterusan pada *ADTP* dengan ketidakpastian terhadap *MTG* menyebabkan kerumitan pertukaran semantik antara menjawab kebarangkalian maksimum dan set hasil skor tinggi. Penyelidikan semasa mengenai menangani ketidakpastian berkisar tentang mencipta algoritma yang disesuaikan secara khusus yang boleh beroperasi dengan kehadiran nila ketidakpastian menggunakan kedua-dua pendekatan asas-kiraan dan asas-masa *MTG*. Objektif utama kajian ini adalah untuk mencadangkan rangka kerja pemprosesan pertanyaan *Top-k* untuk ADTP yang memfokuskan kepada penggunaan kecekapan model tingkap gelongsor. Selepas mewujudkan peraturan dan teori kebarangkalian model ini, satu kaedah telah direka untuk menyokong algoritma pemprosesan *Top-k* ke atas *MTG* sehingga calon berpotensi *Top-k* tamat tempoh. Kajian ini juga menyediakan gambaran keseluruhan kaedah pengoptimuman yang lebih baik untuk menangani lebihan komputasi dalam konteks pemprosesan pertanyaan *MTG* dan *Top-k*. Kaedah ini mengurangkan kos komputsai dengan mengendalikan dasar sisipan dan keluar dengan cekap untuk calon item tuple yang sesuai dalam durasi tetingkap yang ditentukan. Eksperimen dalam kajian ini membandingkan algoritma *SWMTop-kDelta* berdasarkan dua algoritma

dari penyelidik terdahulu dan dua dasar asas algoritma *MTG* untuk menilai keberkesanannya. Pembangunan Algoritma ini menggabungkan rangka kerja dari Fasa 1 hingga 3 dengan menggunakan set data sebenar dan sintetik. Ia menilai kecekapan berdasarkan perbandingan bilangan kejadian dunia yang berkemungkinan dan masa pemprosesan. Eksperimen telah dijalankan sebanyak tiga kali dan merekodkan nilai secara purata terhadap setiap lelaran ini. Apabila saiz set data bertambah, *SWMTop-kDelta* secara konsisten menunjukkan prestasi yang baik, tanpa mengira saiz set data dan ukuran nombor parameter-k. Walaupun peningkatan awal hanya sedikit, prestasi boleh meningkat secara konsisten dengan membuat pelarasan tertentu, seperti meningkatkan bilangan pembahagian tetingkap, mengurangkan saiz tetingkap, mengurangkan bilangan pertanyaan dan melaraskan ambang kebarangkalian (*d*) dengan lebih kerap. Ia menunjukkan peningkatan ketara sebanyak 30%–90% berbanding kaedah lain, terima kasih kepada prestasi yang konsisten dan kebolehskalaan yang kukuh. Eksperimen telah dijalankan dalam tiga kali ganda dan merekodkan nilai min lelaran ini. Usaha penyelidikan ini akan memberi sumbangan yang berharga kepada bidang pemprosesan pertanyaan komputasi Top-k.

Kata Kunci: Top-k, Aliran Data Tidak Pasti, Model Tetingkap Gelongsor, Item Tupel, Bilangan Kejadian Dunia

SDG: MATLAMAT 4: Pendidikan Berkualiti

ACKNOWLEDGEMENTS

Having finished this thesis, we would like to begin by expressing our gratitude to the kind and caring deity. We acknowledge the grace of Allah and express our gratitude for the numerous benefits, challenges, and chances that have made it possible for me to finish this job. As a result of this process, I have gained a tremendous deal of intellectual and personal experience. Throughout the PhD program, Dr. Siti Nurulain Mohd Rum served as both our advisor and mentor. I am extremely grateful to her for all of the hard work, invaluable research skills, helpful support, and valuable criticism that she provided.

In addition, I would like to express my appreciation to Professor Dr. Hamidah Ibrahim and Associate Professor TS. Dr. Iskandar Ishak, both of whom were members of our advisory group, for the insightful criticism, positive reinforcement, smart counsel, and creative suggestions that they provided. We would like to express our gratitude to our cherished family and friends, who have never failed to support us. We would like to express our appreciation for their unflinching support and encouragement during this entire process. In conclusion, we would like to express our gratitude to everyone who has assisted us in this project, whether it be financially, morally, or in any other way. I am grateful to each of you for your contributions.

This thesis was submitted to the Senate of Universiti Putra Malaysia and has been accepted as fulfilment of the requirement for the degree of Doctor of Philosophy. The members of the Supervisory Committee were as follows:

Siti Nurulain binti Mohd Rum, PhD
Senior Doctor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Chairman)

Hamidah binti Ibrahim, PhD
Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Member)

TS. Iskandar bin Ishak, PhD
Associate Professor
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Member)

_____
ZALILAH MOHD SHARIFF, PhD
Professor and Dean
School of Graduate Studies
Universiti Putra Malaysia

Date: 12 September 2024

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

UPM	Universiti Putra Malaysia
UDS	Uncertain Data Stream
SWM	Sliding Window Model
PW	Possible World
DSMS	Data Stream Management System
ALU	Attribute-Level Uncertainty
TLU	Tuple-Level Uncertainty
CQL	Continuous Query Language

CHAPTER 1

INTRODUCTION

1.1 Overview

Query processing has been extensively utilized across several domains in recent years. It is a crucial tool that users use to organize their data through analytical requests. Traditional query processing approaches that rely on in-memory algorithms cannot manage the vast volume of data streams. Handling uncertain data streams is crucial in many applications due to the inherent lack of certainty in real data streams. Top-*k* query processing entails the compilation of information in a manner that promotes optimal and efficient scanning and retrieval. Therefore, it is crucial to create efficient Top-*k* query processing methods that are time-efficient and then integrate a competent Data Stream Management System (*DSMS*) to facilitate query processing across many data sources.

This is because the results might be considered "relevant," and users frequently own a vested interest in a few pertinent outcomes, which may encompass streaming data. A Top-*k* query processing determines the final results, a commonly used approach to address issues related to efficiency and scalability. This component is essential in ranked retrieval, from centralized data to the distributed data environment. The Top-*k* query processing approach, as proposed by (Ilyas et al. 2008), only computes the "best" *k* results instead of all results. Ranking requirement, in which an algorithm will provide k tuple items with the highest scores based on a user-defined scoring function

and a specific query (R. Fagin et al., 2003). The key challenge with the Top-*k* query is determining the ranking of tuple items based on preferences to discover the potential candidate of Top-*k*. Each tuple item selected in the Top-*k* should possess a level of acceptability in the application equal to or greater than any other entity in the list. The notion that a comprehensive ranking should incorporate subject expertise and that ranking results are susceptible to interpretation is also noticeable.

Figure 1.1 : Top-*k* example

Consider a simple tabular dataset that includes pricing information for several things. Figure 1.1 illustrates the semantics of Top-*k* queries through each record. To enable Top-*k* queries, one may easily enhance SQL by introducing a new phrase that explicitly restricts the number of results. Let's examine a Top-*k* query that includes a provision to stop after retrieving the Top-*k* results. Conceptually, the remaining part of the query is processed as anticipated, resulting in the creation of a table *T*. Hence, the outcome consists of the initial k tuples from *T*. If *T* includes fewer than or equal to *k* tuples, the "STOP AFTER *k*" instruction becomes irrelevant. If several sets of tuples satisfy the "ORDER BY" criterion, both of these sets are considered valid answers (non-deterministic semantics).

When assessing the Top-*k* queries, it is crucial to consider two key factors. **Factor 1:** The query structure encompasses single relationships, multiple relationships, aggregate performance, and other associated elements. **Factor 2:** The access path is the route that does not have an index or has indexes for all or partial aspects of the ranking. The most straightforward scenario is the Top-*k* selection query, which involves a single relation. A noteworthy instance establishes how a top operator can be employed in this case. The tuple items entering the leading operator are sorted based on *S* by utilizing the full scan, which may be implemented in the pipeline. In this scenario, it is adequate to acquire the initial k tuple items from the input stream and promptly return a tuple once it has been read. The syntax for implementing a logical top operator indicated as *k*, *S*, that produces the *k* highest-ranked tuple items based on *S* is as follows:

```
SELECT      <some attributes>
FROM        R
WHERE       <Boolean conditions>
ORDER BY    S(…) [DESC]
STOP AFTER k
```

In recent years, significant research has focused on Top-*k* query processing over data streams due to its relevance in many developing applications. Due to advancements in Industrial Revolutionary 4.0 technology and the rapid growth of big data technology, a growing amount of information is being collected in several fields, including object tracking, RFID technology, sensor networks, information extraction, and data integration (Mingyi & Yinju, 2015). Within several application fields, the data sources utilized for processing may be doubtful due to the uncertainty resulting from the inherent unreliability of sensor devices or external data manipulations such as privacy-preserving data

transformations. Probability distributions are commonly used when describing data records in uncertain administration rather than deterministic values (Aggarwal et al., 2009). Therefore, a conventional Top-*k* query is only sometimes capable of handling an uncertain Top-*k* query and necessitates a redefinition of the Top-*k* query semantics for uncertain data stream queries.

1.2 Motivation

The data shown in Figure 1.1 is definitive. However, when the data in a data stream is uncertain or exceedingly challenging to determine, the issue of a Top-*k* query becomes even more complex regarding the meaning of Top-*k* over uncertainty. For example, it may not be feasible to identify the exact price of a vehicle, but the probability distribution of prices can be determined. One may need to familiarize themselves with the height range of a person between a height of 1.6 meters and 1.7 meters and beyond. Interpreting this type of information into measurements is typically challenging. Uncertain data stream processing presents challenges for several reasons: (i) the tuple items in the stream are generated in real-time, (ii) the application lacks control over the order in which tuple items arrive, both within a single data stream and across multiple data streams, (iii) the scale of data streams is inherently unlimited, and (iv) the data stream objects are discarded once they have been processed (Minh et al., 2020; Ren et al., 2021). The properties of the tuple items might vary over time and be continually created. A larger window size can lead to overestimating the necessary window size, ultimately leading to the undesirable and unexpected return of tuple items (Jin et al., 2008; Liu et al.,

2018). This study examines how the interplay between scoring and probability impacts the final results of an uncertain top-k query. The gap between these two aspects will give rise to numerous uncertain doubt interpretations, mostly centered around possible world semantics events.

Over the past several years, multiple additions have been made to the Top-*k* query issue, and various methods have been developed specifically for handling uncertain data streams. Applying the Top-*k* query processing to uncertain data streams using a Sliding Window Model (*SWM*) is a recent and growing database trend. Although the execution of Top-*k* queries on data streams with sliding windows has not been thoroughly studied, the *SWM* on uncertain data streams has received little attention. This is due to the numerous challenges that arise from the time requirements for processing incoming and expiring tuple items on high-speed streams and the increasing number of possible worlds caused by the uncertain nature of the data stream (Aggarwal, Member, & Yu, 2009). This can also contribute to the fact that the expected generation from tuple items for both circumstances is more likely to be volatile. In the field of semantics, numerous studies have been carried out to develop models for describing uncertain data streams (Sarma et al., 2006). These models can be utilized as a sequence of events of a relationship, such as the relational data model (Fuhr & Ro, 1997; Lakshmanan et al., 1997), the semi-structured data model (Burdick et al., 2008), the stream data model (Jin et al., 2008; Ré et al., 2008), the multidimensional data model (Burdick et al., 2005; Ré et al., 2008), and several other models. Within the field of study on uncertain data streams, the theory of a possible world is a significant idea. The

prevailing semantics of these models are all based on the possible world (Abiteboul, 1991; Green & Tannen, 2006). Therefore, the theory of a possible world is an important notion.

Uncertainty in a stream tuple item might arise from two reasons: (i) uncertainty in its value and (ii) existential uncertainty. An item in a tuple is considered to have an uncertain value if its value is represented either by a Probability Density Function (PDF) (Minh et al., 2020; Ye, 2018) or by discrete datasets (Yeh et al., 2009). This thesis explores the challenge associated with the sliding windows model and proposes a new sliding window method called the delta-based method. This method offers an adaptable technique for creating a segmentation division that manages the tuple items entering and exiting the window while computing score and probability values within the window frame. The expansion of the window semantics to describe the notion of the proposed sliding windows model is thoroughly examined, and it also provides both reliable and similar window management techniques. Several methods have been proposed to handle the processing of Top-*k* queries in uncertain data streams. However, a few studies have been on processing uncertain data in a *SWM*, explicitly focusing on Count-based and Time-based approaches. Processing continuous Top-*k* queries over a sliding window is a challenging problem in the uncertain data stream setting (Yang et al., 2011).

Motivated by this, the study concentrates on centralized Top-*k* query processing, specifically the continuous querying of Top-*k* tuple items. Furthermore, it explores using a continuous query process to determine the highest scores and maximum probability values within centralized uncertain

data streams. This is done within a proposed delta-based *SWM*. Each data stream consists of a collection of elements associated with a tuple item, and its numerical values are derived from the uncertain data streams. The evaluated Top-*k* continuous query is bound to the most recent segment of the uncertain data stream, and the numerical values of the tuple items are adjusted correspondingly as the sliding window slides.

1.3 Research Problem

rocessing Top-*k* queries over uncertain data streams has emerged as a promising approach in developing an intuitive information system, as explained in Section 1.1. In addition to certain data, Top-*k* query processing is essential for uncertain data streams, where data requires both reliability and consistency. Data's underlying and unavoidable uncertainty is typically attributed to unexpected and unreliable signal readings, sensor update delays, or insufficient expertise. When dealing with data uncertainty, it's not appropriate to eliminate uncertain values immediately. This is because doing so can result in inaccurate or misleading outcomes. Uncertainty data mainly refers to factors such as indeterminacy, unreliability, unpredictability, randomness, inconsistency, variability, incompleteness, unknown bounding, and irregularity (Diao et al., 2009; Kawashima, 2010; Hu et al., 2017). The query processing results may be reliable and correct if the tuple item's uncertainty is appropriately addressed. Hence, it is crucial to eradicate confusion by thoroughly considering the data uncertainty model, which may adequately represent knowledge about facts using uncertain data. It is often

challenging to handle Top-*k* queries on certain data, let alone uncertain ones. This is due to the complexity of Top-*k* computation, which is affected by various factors such as score value, probability, data interval, and data set size. Consequently, the performance issue becomes even more severe for Top-*k* queries with independent uncertain data stream results.

Our current research focuses on the challenges and uncertainties surrounding top-k query processing in Real-Time Traffic Management applications. These issues are particularly prominent and require careful examination. This application focuses on managing traffic flow in urban and suburban areas by monitoring the speed of vehicles that violate road rules. The data from traffic sensors can be prone to noise and interruptions, with the rate at which the data arrives often showing significant variations. In addition, unforeseen circumstances such as accidents or road closures contribute to the need for more predictability. One example of similarity is the air pollution index, which is determined by the uncertainty of data collected from multiple sensors simultaneously.

Following the above research problem, we have identified three challenges that need to be addressed in this thesis, focusing on analyzing and computing Top-*k* on uncertain data streams.

***Challenge* 1: An efficient implementation of a sliding window approach is needed to handle Top-k queries on uncertain data streams. If the sliding window model is not employed correctly, candidate tuple items within the window frame may cause significant overlapping computing**

costs. Therefore, it is essential to ensure that the sliding window approach is used effectively.

An efficient method is required to determine the type of uncertain data model before processing Top-k queries. Three categories for uncertain data models have been identified: independent, correlated, and locally correlated (Mohamud et al., 2023). Multiple items that are not connected by any rules are said to be independent. The uncertainty implies the ambiguity of the tuple's presence. When both are independent, there are additional relationships between tuple items that should not be detected simultaneously, and so forth. Every method for top-k queries has constraints, regardless of whether it is applied to deterministic or probabilistic data. It gets more complex when candidate tuple items are generated from expired tuple items and when they overlap within the sliding window frame. When an event is triggered, for instance, tuples may be generated. The new tuple item may not exist in some possible world creations if the occurrence is uncertain. Thus, the tuple items will be handled using the exact uncertain data stream mechanism, with an equal amount of tuple items in each sliding window. No comprehensive research has been conducted on combining *SWM* with possible world computing for the Top-k to adapt. Yet, managing a variety of tuple items might be so demanding that, in the least successful circumstance, it gets progressively more complicated linearly as the sliding window size grows. Inefficient processing of each tuple item might result in unnecessary top-k computation and resource depletion. The approach mentioned below explains how to determine candidates generically without specifying the efficiency of processing top-k queries during possible world computation.

This thesis examines top-k query algorithms with mutual independence and tuple item existence uncertainty in uncertain data streams. Which tuple item from uncertain data streams should be identified as top-k potential candidates considering the uncertainty in the data stream? How can we efficiently handle increasing computation complexity caused by overlapping candidate tuple elements in a sliding window frame? In the literature review section, various top-k preference query algorithms have been proposed using a sliding window. Based on our research, (Ren et al., 2021) is the only study that has tackled the questions posed in Section 1.2. They developed a *Topk-iDS* algorithm using Count-based and Time-based *SWM* to analyze incomplete and uncertain data streams. This algorithm utilizes a score-probability (possible world) setting. Hence, it aids in identifying alterations in the results of the top-k potential candidates and ensures the continuous updating of the most recent sliding window. Because of their distinct methodologies, this study will analyze and assess (Ren et al., 2021) instead of (Glavic & Kennedy, 2023). In 2023, Glavic's research centers around domain approximation ranking. This study necessitates the assessment of accuracy distribution probability before computing time. However, the (Ren et al., 2021) approach focuses on computing possible world tuple items within a sliding window frame, which aligns with this study's proposed contribution.

***Challenge* 2: An efficient algorithm is needed to improve the retrieval of the top-k query results from uncertain data streams, particularly when computing the top-k score and probabilities expected to have significant computational expenses for generating the set of possible worlds.**

Top-k queries are the most popular queries in databases and several data stream applications. The traditional top-k query retrieves k items with the highest scores according to specific scoring criteria. The significance rests not only in the ranking but also in its probability of membership, scores, and probabilities of other tuples, all of which are influenced by including a tuple in a top-k answer (Minh et al., 2020; Zarko & Zi, 2015). The interaction between tuple scores and information uncertainty determines the best solutions (Guo et al., 2022). It is indisputable that computing a potential Top-k result utilizing maximum probability and high scoring is time-consuming and challenging in an uncertain data stream environment. It becomes more complex, mainly because these concerns frequently do not align with the immediate application requirements in practice (Z. Zhang et al., 2018). For example, the uncertain top-k query might be employed to ascertain the case of an "athlete" with the highest probability of participation in an upcoming competition. However, a previous study (Carbone et al., 2019; Mao et al., 2017; Qiao et al., 2020; Vouzoukidou, 2016) demonstrated that a substantial amount of needless visits to the listings might still result in considerable rises in execution costs. The performance matter, measured in terms of processing time, is critical for Top-k queries operating on uncertain data streams (Malki et al., 2020).

Quantifying the uncertainty in data streams based on possible world models is accomplished using the probability of all candidate tuple items (Minh et al., 2020; Yijie Wang et al., 2013). This model can translate the possibility of an uncertain data stream into the probability of possible world occurrence (Glavic & Kennedy, 2023). The number of possible world instances will exponentially

expand with a growing amount of uncertain data, even though this model is feasible when considered (G. Xiao et al., 2016). Therefore, existing top-k query processing methods cannot be directly used (Minh et al., 2020). Within a *SWM*, representing tuple items using an inefficient strategy for managing continuous queries with uncertain data streams results in complex trade-offs between maximizing probability queries and generating result sets with high scores. Due to the dynamic nature of the window frame size, processing many tuple elements within the window may pose a challenge (Jacques-Silva et al., 2015). It is crucial to analyze how to retrieve the correct tuple items that meet the Top-k query requirement using scoring or probability (J. Chen & Feng, 2017). In addition to addressing the primary concern of top-k computation, it is essential to determine the critical generation rules of possible world semantics that arise from the combination of score and probability (Glavic & Kennedy, 2023; Ren et al., 2021; Z. Zhang et al., 2018). More research on the top-k query should involve score-probability values and their variants. Work has yet to be done on brief top-k score-probability computation using a *SWM* segmentation that considers uncertainty from possible world vector rules.

Due to the nature of an uncertain data stream, how can we determine continuous top-k queries on an uncertain data stream? Should we compute score values (without probability) or probability values (without score values) or combine score-probability values to improve computation cost? Is it possible to build effective computational methods to minimize query processing time by enhancing the operation of scanning all potential top-k vector rules generated by selected possible worlds? The field of top-k query processing has addressed the first question to some extent. Studies by (Z. Zhang et al., 2018)

focused on the top-k dominant relationship score and its associated range probabilities. However, their work needs to encompass the techniques for computing top-k in conjunction with the *SWM* approach. Our work aims to address the problem of top-k computation on continuously uncertain data streams by considering the trade-offs between reconciling maximum probability queries and providing result sets with high scores.

***Challenge* 3: The number of possible world instances grows exponentially, affecting top-k continuous query processing time to be high.**

Although past studies (Payton et al., 2012; Minh et al., 2020) recommend techniques for deciding the number of instances a system will execute changes (execute the snapshot queries), no assurance will be given that no results would be missed by utilizing the standard execution approach. However, using the naive baseline method to evaluate all the stored queries on each entry will be impractical. Up to now, data stream analysis methods, both in the literature and in commercial environments, still need to handle this dynamic scoring function and probabilities directly instead of applying an approach to conducting continuous queries regularly (Vouzoukidou, 2016). The event-driven model used for query responses ensures that no results are missed. Incoming data streams are indexed, while periodic continuous queries are performed using the pull model (L. Li & Wang, 2020). The process executes stored user queries periodically and provides new results from the top-k lists. As new items arrive, objects are dynamically changed in the sliding windows, and the old ones are removed (Jacques-Silva et al., 2015). Some

users prefer to keep the results current, while others only select the top-k potential candidates from the latest sliding window.

Uncertain data tuples (G. Xiao et al., 2017) can create multiple possible worlds. However, these worlds are structured and regulated by established rules of generation. For instance, inter-company tuples that depict the same situation as a world entity are excluded. The possible world can be represented by a range of tuples generated in the *SWM* (Glavic & Kennedy, 2023). Tuple items within the *SWM* can be generated with multiple possible worlds for a specific ,./timestamp, resulting in exponential growth within the sliding window as the size expands. For instance, if the window size is six (6) and the last tuple item exits the window, the newest tuple item will be inserted into the sliding window frame. Recompute the latest five tuples in *SWM* together with the possible world, which may result in resource depletion. When the window size is too large, new possible worlds must be created to complete the top-k tuple items process. This is due to the requirement to segment tuple items and simultaneously update and manage the probabilistic top-k tuple items across sliding windows as time progresses. Thus, how can we establish a search space that avoids unnecessary top-k computations while constantly selecting and determining possible world vector rules without prior knowledge? Given that the underlying number of tuple items retained at one time in the sliding window frame must match the sliding window size, can we exploit optimization techniques involving continuous top-k computation to manage the performance issue due to the possible world? To the best of our knowledge, the study from previous researchers has devised better algorithms for five Top-

k ranking queries: U-Topk (Khosla, 2015), PT-k (Hua et al., 2008), PTkS (G. Xiao & Li, 2015), Global-Topk (Cormode & Li, 2009), and *DRA* (Z. Zhang et al., 2018). These methods are based on various semantics and query criteria. The main issue these algorithms address is the exponential increase in possible world instances in a scenario with massive tuple items that include non top-k potential candidates. Instead of using a sliding window, their methods only use sorting-reduction procedures to handle the problem, and this can be improved because the reduction of applicable vector rules that do not belong to top-k potential candidates can still be done before and after the possible world-generated occurs. Therefore, this study suggests efficient optimization techniques to prevent redundant scans of whole datasets, optimize memory utilization, and process only the latest tuple item.

1.4 Research Objectives

The main goal of this research is to propose a framework capable of efficiently performing a Top-k query computation on tuple items that incorporate uncertainty. To accomplish the goal, the following objectives are established concerning the challenges presented in Section 1.3:

i. To propose an efficient approach for processing the top-k query over the sliding window model while avoiding excessive overlapping computation caused by selecting the same candidate tuple items within the window frame.

ii. To propose an extension algorithm to the approach above that can retrieve top-k potential results, eliminating unnecessary computations of scores and probabilities before generating a collection of possible world rules.

iii. To propose an efficient optimization approach to reduce the top-k computation time and complexity.

Therefore, this study will conclusively demonstrate specific metrics for measuring the desired improvements in Top-k computational efficiency, allowing for the evaluation of its success. The experiment that will be implemented minimizes the average time required to process and update the top-*k* result list by reducing processing time. The experiments were conducted in triplicate, and the mean value of these iterations was reported. The required target is to achieve a optimization of 40% reduction in average top-k processing time compared to existing solutions. Improving scalability ensures the process scales efficiently with the increasing tuple item volumes and arrival rates where studies are needed to decrease the generation of possible worlds from the number of tuple items within sliding segmentation to reduce throughput top-k processing compared to baseline methods. We thoroughly compared our proposed model algorithm, two previous researcher algorithms, and two baseline approaches to evaluate the experiments. This evaluation covered the frameworks used in Phases 1 to 3. We chose six performance metrics to assess the effectiveness of the proposed frameworks. These metrics consider various factors such as the size of the data set, parameter count (k), window size (W) and segmentation, probability threshold (d), and the number of queries for the top-k during the sliding window process.

The initial parameter comparison measurement aims to investigate how the dataset size affects the performance of our proposed algorithm. Our algorithm effectively reduces the number of potential scenarios generated by implementing an eviction policy called delta. As a result, this leads to a significant decrease in overlapping computation segments and prevents the

generation of unnecessary possible world rules, demonstrating its superior performance. As the data set's size grows, the algorithm's processing time will also increase. Our *SWMTop-kDelta* algorithm has successfully decreased the possible worlds needed to complete the top-k query processing more efficiently than previous methods. In second comparison measurement regarding the number parameter *k* on the performance of different algorithms, our proposed algorithm consistently outperformed others in terms of possible world number and processing time comparison. It demonstrated robust scalability and maintained consistent performance even when the value of k was changed up to 10. This is because the parameter k that affects the top-10 values takes a lot of time to compute, which leads to more potential candidates being chosen.

The third comparison measurement will analyze the impact of window size and segmentation. Our proposed algorithm showcased exceptional performance compared to previous methods. Fewer window sizes would increase computational work when working with large quantities. This is because creating a larger number of window segments requires more work, which can limit the computational possibilities within smaller segments. Similarly, when measuring the processing time result, it is essential to determine the combination of score and probability before the uncertain data stream time disrupts the delta invariant. During this iteration, pruning focuses more on intra-window segmentation and less on inter-window segmentation comparisons. Whenever the number of window segmentations increase while decreasing the window size, the performance remains intact and even improves slightly.

In the fourth comparison measurement, we analyze the impact of varying probability thresholds on the number of possible world comparisons. The number of comparisons increases as the probability thresholds decrease from 0.5 and below. The algorithm we propose surpasses others in terms of reducing the generation of possible worlds and processing time. These factors are greatly influenced by the frequency of changes to the probability threshold and the state of the sliding window. Based on the above 40% percentage improvement, *SWMTopk-Delta* eliminates unnecessary computations by focusing on less possible world vector rules and optimizing sorting for remaining tuples using a sliding window approach based on a combination score and probability function. The fifth comparison measurement examines the impact of the number of queries. Our proposed algorithm demonstrated superior performance to other algorithms regarding the number of possible worlds and processing time. Initially, it showcases a steady performance with a consistent increase as the number of queries increases. However, when the number of queries reaches 20 seconds, SWMTop-kDelta demonstrates superior performance compared to the other methods. The number of possible worlds generated during the continuous query time frame depends on the tuple items maintained, affecting the percentage improvement. If the query execution time is decreased, it will result in a reduction in processing time.

1.5 Research Scope

Preference queries mitigate issues encountered by traditional queries by utilizing preference assessment approaches that leverage uncertainty

characteristics to choose the most desired tuple items for effective selection. Various preference queries in uncertain stream database systems are managed by a DataStream Management System (*DSMS*), each with distinct methodologies, advantages, and disadvantages. Below, several preference queries, such as top-k, top-k dominant, and skyline, are discussed in depth, focusing primarily on top-k query processing. Figure 2.1 illustrates the taxonomy that categorizes the pertinent topics presented in this chapter.

Each query type differs in its approach to determining the user-defined connection. Top-k employs a user-defined function, whereas skyline uses dominance theory to identify dominated and dominant tuple elements. A detailed discussion of the various forms of preference questions may be found in Section 2.3. Numerous research studies have focused on multiple-preference questions, mainly aiming to provide a practical approach for processing these queries. Research is conducted to address difficulties specific to the type of data being managed. Various factors, such as data uncertainty, data created through data streams, data dynamism, etc., necessitate distinct techniques for processing queries. This thesis primarily focuses on top-k query processing, specifically addressing challenges associated with uncertainty and dynamism in fast data streams. The study categorizes prior efforts based on the following criteria: (i) Works that focus on the assumption that all tuple items in uncertain data streams carry the semantics of the possible worlds. This thesis introduces the Data Uncertainty Model, which encompasses Attribute-Level Uncertainty (ALU) and Tuple-Level Uncertainty (TLU) of data or tuple items. Works that address the challenges of

data uncertainty in a centralized environment. This thesis examines the latest items from the uncertain data stream. (ii) Works that focus on tackling data uncertainty issues in a centralized environment. In this thesis, the use of the term considers the most recent items from the uncertain data stream. The dynamic, uncertain data stream will be deleted after processing. (iii) Works dealing with uncertain data streams are naturally unlimited in size. The sliding window model (*SWM*) approach tracks the tuple items entering and exiting the window frame during top-k query processing. The potential research avenues are (i), (ii), and (iii). The routes described earlier are illustrated in Figure 2.1, with the thick green line indicating the focus of this thesis.

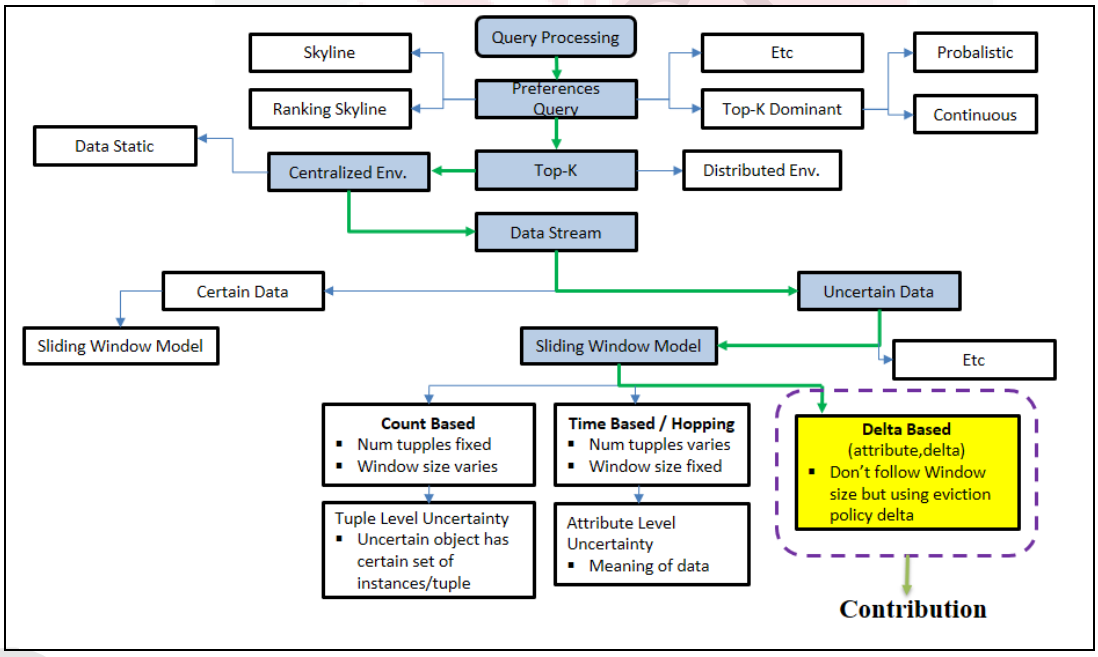

Figure 1.2: Preference queries taxonomy

To improve the relevance and applicability of processing top-*k* queries in a Delta-based *SWM* for evaluating *UDS*, this study may also be used to be integrated with the most recent data processing technologies, specifically:

Leveraging Stream Processing Frameworks: Processing large-scale data streams in real-time by implementing simplified SWM to obtain incremental top-k result updates so that it can comprehensively model high-velocity data streams efficiently. For instance, the Spark Streaming processing ecosystem provides powerful tools for data analytics and machine learning.

Leveraging Solutions for Modern Data Storage: Efficiently managing time-based tuple items within the *SWM* is crucial for optimizing handling in time-series data streams. This allows *DSMS* to provide efficient querying and aggregation, resulting in significant advantages for modern data storage solutions. A current example is data stream management for *NoSQL*, *InfluxDB*, and *TimescaleDB*. These databases are designed to handle large amounts of unstructured data streams that may contain uncertainties.

Visualization/Dashboarding from Microservices: The primary objective is to enhance the system's modularity, scalability, and maintainability by dividing the main application into distinct microservices. Each service can be deployed, scaled, and maintained independently, resulting in improved integration and execution of top-k query processing. Following this, a dashboard equipped with visualization tools can display real-time top-k results and other metrics obtained from UDS processed through microservices. This assists in monitoring and troubleshooting the system.

Applications in various domains have been considerably improved due to recent developments in top-*k* query processing for *UDS*. These breakthroughs

have enhanced data security, accuracy, and operational efficiency. Encrypted top-*k* queries are used in business intelligence to offer safe real-time analysis and customer segmentation. One example of this is identifying products that are selling the best while still protecting the confidentiality of the data. Secure processing in health informatics makes it possible to practice personalized medicine and epidemiology. This is accomplished by identifying afflicted regions and finding the most effective treatment plans through probabilistic models. Probabilistic models, for instance, help determine the top-k result list where effective treatment plans for patients are based on uncertain patient data. When successfully controlling traffic congestion and pollution levels, real-time monitoring systems, such as smart cities and environmental monitoring, can reap the benefits of top-*k* processing. The identification of congested routes through the use of real-time traffic sensor data and the subsequent upgrading of these routes through the utilization of *SWM* evaluations is one example. Implementing these improvements guarantees the reliable management of *UDS*, broadening the practical impact of research in various diverse applications. These technological breakthroughs provide comprehensive solutions for real-world difficulties in managing large-scale, uncertain data settings. These solutions are achieved by integrating encrypted processing, crowdsourcing, and probabilistic analysis methodologies.

The scope of this research work is outlined in the following items:

i. The type of query shown in this research is preference queries, which are limited to uncertain top-k query processing as a typically used method in most of the multi-criteria decision making of centralize environment (Akbarinia et al., 2007, 2011; Gong et al., 2009; Ilyas et al., 2008; Ilyas & Chang, 2007; Khosla, 2015; S. Lin, 2010; X. Lin et al., 2017; Theobald et al., 2008; Vlachou et al., 2008; Xie & Wood, 2013; Zhao et al., 2007). The

top-k queries consider by these researchers fall in different areas, like centralized environment without sliding window approach (Hua et al., 2008; Bousnina et al., 2017; S. Ge et al., 2013; T. Ge et al., 2009; C. Lin et al., 2017; Mingyi & Yinju, 2015; Minh et al., 2013; Papadopoulos et al., 2021; G. Xiao et al., 2017; Z. Zhang et al., 2018; J. Chen & Feng, 2013).

ii. This study emphasizes the issue of representing independent uncertainty data models, which contains an essential characteristic that combines both Attribute Level Uncertainty (ALU) and Tuple Level Uncertainty (TLU), along with their related relationships with possible worlds (Aggarwal et al., 2009; Khosla, 2015; Pei et al., 2013; G. Xiao & Li, 2015).

iii. The top-k query algorithm of the proposed methods are utilized using an approach of sliding window strategy (G. Xiao et al., 2016) (B. Chen et al., 2017) (Khosla, 2015) (Bhushan et al., 2017) (Jacques-silva et al., 2015) (Gedik, 2014).

iv. The performance evaluation of the top-k query processing over uncertain data streams is conducted primarily on extensive experiments to demonstrate the efficiency of the proposed framework in continuous query categories with sliding window approach (Dai et al., 2012; H. Jiang et al., 2020; Jin et al., 2008; H. Liu et al., 2018; Ren et al., 2021; Shen et al., 2014; N. Xiao et al., 2013; Zarko & Zi, 2015).

v. This research work involves top-k optimization for uncertain data stream processing and requires an effective type of data indexing and pruning (J. Chen & Feng, 2017; Jason et al., 2015; Kar, 2017; Malki et al., 2020; Rahul & Tao, 2019; Ren et al., 2021; Z. Zhang et al., 2018; N. Xiao et al., 2013; Zarko & Zi, 2015; {Formatting Citation}John et al., 2016).

vi. This study utilizes both synthetic and real datasets. The synthetic data sets are generated independently, while the real data sets are collected from the Chicago Trip Taxi and Airbeijing. The data sets commonly used in this context include references to (Minh et al., 2013; Ren et al., 2021; G. Xiao, Wu, et al., 2016; Z. Zhang et al., 2018; Lee et al., 2009; Peng & Wong, 2014; S. Ge et al., 2013).

1.6 Research Contributions

Our study employs a sliding window model to analyze and compute the top-k query on uncertain data streams. To attain the desired contribution of the thesis, it is essential to thoroughly examine several concerns before implementing the recommended framework. These issues include:

i. Further exploration and analysis of uncertain data stream model categories are needed.

ii. The efficiency of utilizing the proposed sliding window model on the continuous query.

iii. The query processing used to get suitable Top-k based on scoring and probability.

iv. The optimization to effectively reduce the top-k computation time and complexity.

Despite the contributions made by (Mingyi & Yinju, 2015) in processing top-k queries on uncertain data streams, these works have some drawbacks due to their limited applicability when dealing with complexity in both scores and probability values of the tuple (caused by difficulty in handling continuous query). Efficient stream processing strategies are challenging to design due to the unbounded characteristics of data streams (Vouzoukidou, 2016). The main contribution of this study is the improvement regarding the effectiveness of the proposed top-k framework when dealing with query processing on uncertain data streams over the sliding window model. Motivated by this element, the following contributions were made to answer the challenges addressed in this thesis. Explanations of each contribution are as follows:

i. This study defined the uncertain top-k query processing and modeled the type of uncertain data streams. The key features of uncertain data stream types are focused on the data types, and the application details are used to change the definition, storage, initial indexing, query processing, and performance. This thesis will use the independent uncertainty data model and their correspondence with possible worlds. As proposed by this study to target an independent uncertainty model, assumptions have been made that uncertain objects are independent of each other.

ii. Efficient strategies for uncertain top-k query processing are built by analyzing the results of an experimental evaluation using sliding windows. This study has incorporated the proposed delta-based sliding window model as an improvement to replace the common use of count-

based and time-based methods used in existing datastream processing engines (*SPEs*).

iii. Our study proposes a systematic method to achieve high scores and maximum probabilities across all possible world instances. It is a reasonable model, but the number of possible world instances would grow exponentially with the increase in uncertain data stream. Based on the above relations, this study defines a new problem of processing continuous queries that search for the top-k tuple items of different fundamental characteristics.

iv. To optimize *SWM* processing for top-k, appropriate approaches must be developed, as outlined in items ii and iii. Clearing, indexing, and pruning techniques are interrelated, improving query response times by lowering processing time and eliminating unnecessary top-k computations.

v. A systematic empirical study on a real-world and synthetic dataset.

1.7 Organisation of the Thesis

This thesis is organized as follows:

Chapter 1 is an introductory chapter that begins with an overview before discussing the motivations behind this research study, the challenges, the objectives, the scope, and the research contributions.

Chapter 2 reviews the background and the literature review relevant to the research study. The background presents the fundamental concepts of preference queries and the preference evaluation techniques in the database and data streams. It reviews relevant works proposed by previous researchers of preference queries, including Top-k, Top-k dominating, and skyline. The concept of top-k query processing over the sliding window model is given focus, and the method is categorized based on data certainty in a centralized environment, namely, an uncertain data stream. Indexing and query processing methods are interrelated, and this chapter also studied how

optimization of indexing and pruning can help boost continuous query processing time.

Chapter 3 defines the conduct of this research. The different phases of this study and the methods used for each phase are discussed in this chapter. The performance metrics, the experimental settings, and the data sets used in this research study are also discussed.

Chapter 4 details the depiction of the proposed framework for processing top-k queries on uncertain data streams over the sliding window model. This chapter explains and discusses each **Phase I** of the proposed framework. It begins with the identification of the data uncertainty model. Then, it proceeds to implement a sliding window model (*SWM*) approach to establish a segmentation split to attain the tuple items that enter and exit the window, using a selected dataset as an example.

Chapter 5 presents a detailed explanation of the proposed extension steps to the previous framework for handling top-k. This is done through a **Phase II** framework that focuses on establishing the most efficient processing time using the proposed algorithm. The algorithm considers continuous queries, score value, probability value, timestamp interval (delta-attribute), and possible world rules. The size of the data set has an exponential impact on these factors.

Chapter 6 elucidates that various techniques, such as indexing and pruning, are incorporated within the **Phase III** framework to improve computation efficiency. This is achieved by focusing on selected possible world vector rules and considering only the top-k result list, which may or may not have overlapping tuple items within the same window frames. Acknowledging that specific computations may not have been executed because of expired or removed tuple items is crucial. Consequently, it is necessary to propose efficient optimization approaches to prevent multiple scans of the entire data sets, reduce computation, and process only the most recent tuple-specified items.

Chapter 7 presents the results of the systematic performance study conducted to evaluate the algorithm performance of the proposed frameworks compared to the most relevant existing algorithms. This chapter also discusses the results concerning different parameters, including the dataset size, the score and probabilities value, the distributions of exact/uncertainty values, and the execution of the sliding window model to accommodate continuous queries.

Chapter 8 concludes this study with conclusions and the contributions of the research study. In addition, the recommendations with remarks about the achievements made and possible future research are presented in this final chapter.

REFERENCES

Agarwal, P. K., Sintos, S., & Steiger, A. (2020). Efficient Indexes for Diverse Top-k Range Queries. Proceedings of the 39th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems. https://doi.org/10.1145/3375395.3387667

Aggarwal, C. C., & Yu, P. S. (2009). A Survey of Uncertain Data Algorithms and Applications. IEEE Transactions on Knowledge and Data Engineering, 21(5), 609–623. https://doi.org/10.1109/tkde.2008.190

Akbarinia, R., Pacitti, E., & Valduriez, P. (2007). Processing Top-k Queries in Distributed Hash Tables. Euro-Par 2007 Parallel Processing, 489–502. https://doi.org/10.1007/978-3-540-74466-5_53

Akbarinia, R., Pacitti, E., & Valduriez, P. (2011). Best position algorithms for efficient top- k query processing. *Information Systems*, *36*(6), 973–989. https://doi.org/10.1016/j.is.2011.03.010

Anceaume, E. (2018). Finding Top- k Most Frequent Items in Distributed Streams in the Time-Sliding Window Model. *2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*, 61–62. https://doi.org/10.1109/DSN-W.2018.00030

Anebo, N. P., & M, O. E. C. (2023). Comparative Studies on Intelligent Swarming Network (*iSWAN*) Geno- Generative Algorithm and Top-K Query Processing Algorithm. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, *3307*(3), 581–594. Internet Archive. https://doi.org/10.32628/cseit23903140

Bhushan, A., Bellur, U., Sharma, K., Deshpande, S., & Sarda, N. L. (2017). Mining Swarm Patterns in Sliding Windows over Moving Object Data Streams. Proceedings of the 25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, 1–4. https://doi.org/10.1145/3139958.3139988

Bousnina, F. E., Chebbah, M., Anis, M., Tobji, B., Hadjali, A., & Yaghlane, B. Ben. (2017). On Top- k Queries over Evidential Data. *Proceedings Ofthe 19th International Conference on Enterprise Information Systems (ICEIS 2017)*, *1*(Iceis), 106–113. https://doi.org/10.5220/0006317701060113

Carbone, P., Katsifodimos, A., & Haridi, S. (2019). Stream Window Aggregation Semantics and Optimization. Encyclopedia of Big Data Technologies, 1615–1623. https://doi.org/10.1007/978-3-319-77525-8_154

Chen, B., Lv, Z., Yu, X., & Liu, Y. (2017). Sliding Window Top- K Monitoring over Distributed Data Streams. *Data Science and Engineering*, *2*(4), 289–300.

https://doi.org/10.1007/s41019-017-0053-1

Chen, D., & Chen, L. (2020). Sliding-Window Probabilistic Threshold Aggregate Queries on Uncertain Data Streams. *Information Sciences*, *520*, 353–372. https://doi.org/10.1016/j.ins.2020.02.029

Chen, J., & Feng, L. (2017). Efficient pruning for top-K ranking queries on attribute-wise uncertain datasets. *Journal of Intelligent Information Systems*, *48*(1), 215–242. https://doi.org/10.1007/s10844-016-0403-x

Cormode, G., Li, F., & Yi, K. (2009). Semantics of Ranking Queries for Probabilistic Data and Expected Ranks. 2009 IEEE 25th International Conference on Data Engineering, 16, 305–316. https://doi.org/10.1109/icde.2009.75

Costanzo, M. (2022). Getting the best from skylines and top-k queries. arXiv preprint arXiv:2202.12618. https://doi.org/10.48550/arXiv.2202.12618

Dai, C., Chen, L., Chen, Y., & Tang, K. (2012). An Efficient Algorithm for top-k Queries on Uncertain Data Streams. 2012 11th International Conference on Machine Learning and Applications, 19, 294–299. https://doi.org/10.1109/icmla.2012.57

Dtar, M., & Motwani, R. (2016). The Sliding-Window Computation Model. *Data Stream Management. Data-Centric Systems and Applications. Springer*, 149–165. https://doi.org/10.1007/978-3-540-28608-0

Dzolkhifli, Z., Ibrahim, H., & Hassin, M. H. B. M. (2021). Review on Skyline Query Processing Techniques over Data Stream. In *2021 International Conference on Software Engineering & Computer Systems and 4th International Conference on Computational Science and Information Management (ICSECS-ICOCSIM)* (pp. 443–446). https://doi.org/10.1109/ICSECS52883.2021.00087

Fattah, H. M. A., Hasan, K. M. A., & Tsuji, T. (2022). Indexed Top-k Dominating Queries on Highly Incomplete Data. *Proceedings of the International Conference on Big Data, IoT, and Machine Learning*, (December), 0–10. https://doi.org/10.1007/978-981-16-6636-0

G.Komarasamy , B.Priyachandru , V.GuruMoorthy, M. K. (2017). Dominant Top K Query Consequence Integrity. *International Journal on Applications in Information and Communication Engineering*, *3*(1), 1–5.

Gaihre, A., Weitze, S., & Song, S. L. (2021). Top-k : Delegate-Centric Top- k on GPUs. *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, *November*(39), 1–14. https://doi.org/10.1145/3458817.3476141

Garcelon, E., Avadhanula, V., Lazaric, A. & Pirotta, M.. (2022). Top K Ranking for Multi-Armed Bandit with Noisy Evaluations. Proceedings of The 25th International Conference on Artificial Intelligence and Statistics, in Proceedings of Machine Learning Research 151:6242-6269. Available from https://proceedings.mlr.press/v151/garcelon22b.html.

Ge, S., U, L. H., Mamoulis, N., & Cheung, D. W. (2013). Efficient All Top- k Computation — A Unified Solution for All Top- k , Reverse Top- k and Top- m Influential Queries. *IEEE Transactions on Knowledge and Data Engineering*, *25*(5), 1015–1027. https://doi.org/10.1109/tkde.2012.34

Ge, T., Zdonik, S., & Madden, S. (2009). Top-k queries on uncertain data. Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data, 375–388. https://doi.org/10.1145/1559845.1559886

Gedik, B. (2014). Generic windowing support for extensible stream processing systems. *Wiley Online Library*, *44*(March 2013), 1105–1128. https://doi.org/10.1002/spe

Glavic, B., & Kennedy, O. (2023). Efficient Approximation of Certain and Possible Answers for Ranking and Window Queries over Uncertain Data. *Proceedings of the VLDB Endowment*, *16*(6), 1346–1358. https://doi.org/10.14778/3583140.3583151

Gong, Z., Sun, G.-Z., Yuan, J., & Zhong, Y. (2009). Efficient Top-k Query Algorithms Using K-Skyband Partition. Scalable Information Systems, 288–305. https://doi.org/10.1007/978-3-642-10485-5_21

Gu, F., & Hu, X. (2022). Research on Efficient Top- k Query Based on ARIMA Time Series Model. *Wireless Communications and Mobile Computing - HINDAWI*, 29 April *2022*. https://doi.org/10.1155/2022/4510625

Guo, H., Li, J., Gao, H., & Zhang, K. (2022). PSATop- k : Approximate range top- k computation on big data. *Knowledge-Based Systems Journal*, *235*(107614), 1–17. https://doi.org/10.1016/j.knosys.2021.107614

Gupta, S., Cheng, Y., & Lud, B. (2019). Possible Worlds Explorer : Datalog and Answer Set Programming for the Rest of Us. *CEUR Workshop Proceedings*, *2368*(June 4-5), 44–55. https://doi.org/10.1007/978-3-030-24658-7

Hao, K., Xin, J., Wang, Z., Yao, Z., & Wang, G. (2022). On efficient top-k transaction path query processing in blockchain database. *Data & Knowledge Engineering*, *141*, 102079. https://doi.org/https://doi.org/10.1016/j.datak.2022.102079

Hidayat, A., Cheema, M. A., Lin, X., Zhang, W., & Zhang, Y. (2021). Continuous monitoring of moving skyline and top-k queries. The VLDB Journal, 31(3), 459–482. https://doi.org/10.1007/s00778-021-00702-4

Hua, M., Pei, J., Zhang, W., & Lin, X. (2008). Ranking queries on uncertain data. Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, 3, 673–686. https://doi.org/10.1145/1376616.1376685

Hua, M., Pei, J., Zhang, W., & Lin, X. (2008). Efficiently Answering Probabilistic Threshold Top-k Queries on Uncertain Data. 2008 IEEE 24th International Conference on Data Engineering, 1403–1405. https://doi.org/10.1109/icde.2008.4497570

Ilyas, I. F., Beskales, G., & Soliman, M. A. (2008). A survey of top-k query processing techniques in relational database systems. ACM Computing Surveys, 40(4), 1–58. https://doi.org/10.1145/1391729.1391730

Ilyas, I. F., & Chang, K. C. (2007). Top- k Query Processing in Uncertain Databases. *Proceedings of the IEEE 23rd International Conference on Data Engineering (ICDE 2007), Istanbul (Turkey)*, 896–905. https://doi.org/10.1109/icde.2007.367935

Issa, O., Bonifati, A., Toumani, F., Issa, O., Bonifati, A., Toumani, F., … Bonifati, A. (2020). Evaluating Top-k Queries with Inconsistency Degrees. *Proceedings of the VLDB Endowment (PVLDB)*, *13*(12), 2146–2158. https://doi.org/10.14778/3407790.3407815

Jacques-silva, M. D. G., & Themis, K. W. (2015). Sliding windows over uncertain data streams. *Knowl Inf Syst*, *45*, 159–190. https://doi.org/10.1007/s10115-014-0804-5

Jason, C., Lei, Z., Yongxin, C., & Zheng, T. (2015). Cleaning Uncertain Data with a Noisy Crowd, 6–17. https://doi.org/10.1109/icde.2015.7113268

Jiang, H., Zhu, R., & Wang, B. (2019). EPF: A General Framework for Supporting Continuous Top-k Queries Over Streaming Data. Cognitive Computation, 12(1), 176–194. https://doi.org/10.1007/s12559-019-09661-z

Jiang, W., Wang, T., & Wang, Z. (2020). A Top- K Query Scheme With Privacy Preservation for Intelligent Vehicle Network in Mobile IoT. *IEEE Access*, *8*, 81698–81710. https://doi.org/10.1109/ACCESS.2020.2990932

Jin, C., Yi, K., Chen, L., Yu, J. X., & Lin, X. (2008). Sliding-window top-k queries on uncertain streams. Proceedings of the VLDB Endowment, 1(1), 301–312. https://doi.org/10.14778/1453856.1453892

John, A., Sugumaran, M., & Rajesh, R. S. (2016). Indexing and Query Processing Techniques In Spatio-Temporal Data. ICTACT Journal on Soft Computing, 06(03), 1198–1217. https://doi.org/10.21917/ijsc.2016.0167

Kar, D. C. (2017). On Pruning of Data in a Sliding Window for Computing a Rank-Order Element. *IEEE Signal Processing Letters*, *24*(7), 1005–1009. https://doi.org/10.1109/lsp.2017.2695399

Khosla, C. (2015). Top-k Query Processing Techniques in Uncertain Databases : A Review. *International Journal of Computer Applications (0975 – 8887)*, *120*(20), 33–37. https://doi.org/10.5120/21345-4358

Lai, C., Lin, H., & Liu, C. (2021). Highly Efficient Indexing Scheme for k - Dominant Skyline Processing over Uncertain Data Streams. *IEEE Wireless and Optical Communications Conference (WOCC)*, 1, 97–101. https://doi.org/10.1109/wocc53213.2021.9603141

Lai, C., Wang, T., & Liu, C. (2019). Probabilistic Top- k Dominating Query Monitoring over Multiple Uncertain IoT Data Streams in Edge. *IEEE Internet of Things Journal*, *PP*(c), 2327–4662. https://doi.org/10.1109/JIOT.2019.2920908

Lai, Z., Han, C., Liu, C., Zhang, P., Lo, E., & Kao, B. (2021). *Top-K Deep Video Analytics : A Probabilistic Approach*. *Proceedings of the 2021 International Conference on Management of Data (SIGMOD '21), June 20â•fi25, 2021, Virtual Event , China* (Vol. 1). Association for Computing Machinery. https://doi.org/10.1145/3448016.3452786

Lee, J., You, G., & Ã, S. H. (2009). Personalized top-k skyline queries in high-dimensional space $, *34*, 45–61. https://doi.org/10.1016/j.is.2008.04.004

Li, L., Wang, H., Li, J., & Gao, H. (2018). *A survey of uncertain data management. Frontiers of Computer Scienc*e, 14(1), 162–190. https://doi.org/10.1007/s11704-017-7063-z

Li, W., & Li, P. (2020). Balanced Dominating Top-k Queries over Uncertain Data. *Proceedings of the 2020 5th International Conference on Cloud Computing and Internet of Things*, *September*, 69–76. https://doi.org/10.1145/3429523.3429533

Li,Yafei. Wan, J., Chen, R., Xu, J., Fu, X., Gu, H., Lv, P., & Xu, M. (2019). Top-k Vehicle Matching in Social Ridesharing: A Price-aware Approach. *IEEE Transactions on Knowledge and Data Engineering*, 1–1. https://doi.org/10.1109/tkde.2019.2937031

Li, Yan, Wang, H., Meng, N., Leong, K., & Zhiguo, H. U. (2020). Crowdsourced top- k queries by pairwise preference judgments with confidence and budget control. *The VLDB Journal*, *123*(30), 189–213. https://doi.org/10.1007/s00778-020-00631-8

Lian, X., & Chen, L. (2008). Probabilistic Ranked Queries in Uncertain Databases. *EDBT '08: Proceedings of the 11th International Conference on Extending Database Technology: Advances in Database Technology*, *March 25-3*, 511–522. https://doi.org/10.1145/1353343.1353406

Lin, C., Lu, J., Wei, Z., Wang, J., & Xiao, X. (2017). Optimal algorithms for

selecting top- k combinations of attributes : theory and applications. *The VLDB Journal*, *27*, 27–52. https://doi.org/10.1007/s00778-017-0485-2

Lin, S. (2010). Rank aggregation methods. *John Wiley & Sons, Inc*, *2*(September/October 2010), 555–570. https://doi.org/10.1002/wics.111

Lin, X., Xu, J., Hu, H., & Fan, Z. (2017). Reducing Uncertainty of Probabilistic Top- k Ranking via Pairwise Crowdsourcing. *IEEE Transactions on Knowledge and Data Engineering*, *29*(10), 1041–4347. https://doi.org/10.1109/tkde.2017.2717830

Liu, C.-M., Wang, T.-C., Lai, C.-C., & Wang, L.-C. (2018). An effective method for top-k dominating query processing over multiple uncertain data streams. *2018 27th Wireless and Optical Communication Conference (WOCC)*, 63, 1–5. https://doi.org/10.1109/wocc.2018.8372693

Liu, H., Zhou, K., Zhao, P., & Yao, S. (2018). Mining frequent itemsets over uncertain data streams. *Int. J. High Performance Computing and Networking*, *11*(4), 312–321. https://doi.org/10.1504/ijhpcn.2018.10014443

Mackenzie, J., & Moffat, A. (2020). Examining the Additivity of Top- k Query Processing Innovations. *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, (Oktober), 19–23. https://doi.org/10.1145/3340531.3412000

Mackenzie, J., Petri, M., Moffat, A., & Mackenzie, J. (2022). Efficient query processing techniques for next - page retrieval. *Information Retrieval Journal*, *25*(1), 27–43. https://doi.org/10.1007/s10791-021-09402-7

Malki, A., Benslimane, S., & Malki, M. (2020). Top-k Query Optimization Over Data Services. *Future Generation Computer Systems*, *113*(December 2020), 1–12. https://doi.org/10.1016/j.future.2020.06.052

Mao, Y., Zhong, H., Chen, H., & Li, X. (2017). Two-phase PT-Topk Query Processing Algorithm for Uncertain IOT Data in Dam Safety Monitoring Two-phase PT-Top k Query Processing Algorithm for Uncertain IOT Data in Dam. *Intelligent Automation & Soft Computing*, *8587*(May), 1–8. https://doi.org/10.1080/10798587.2017.1316070

Marwa B. Swidan, Ali A. Alwan, Yonis Gulzar, A. Z., & Abualkishik. (2021). An Overview of Query Processing on Crowdsourced Databases. *Scientific Journal of King Faisal University*, *1*(22), 5–12. https://doi.org/10.37575/b/cmp/2410

Mei, X., Zhang, P., Han, X., & Gao, Y. (2023). Stream Data Event Acquisition Method Based on Multi-Granularity Top-k Query. In *2023 3rd International Symposium on Computer Technology and Information Science (ISCTIS)* (pp. 947–951). https://doi.org/10.1109/ISCTIS58954.2023.10213165

Mencagli, G., Torquati, M., Lucattini, F., Cuomo, S., & Aldinucci, M. (2018). Harnessing Sliding-Window Execution Semantics for Parallel Stream Processing. *Journal of Parallel and Distributed Computing*, (September 2019). https://doi.org/10.1016/j.jpdc.2017.10.021

Milano, P. (2022a). Comparing modern techniques for querying data starting from top-k and skyline queries. *Computer Science (Database)*, *6 Jun*.

Milano, P. (2022b). Flexible Skyline : one query to rule them all. *Computer Science (Database)*, *14 Jan*, 1–10.

Milano, P. (2022c). Giving the Right Answer : a Brief Overview on How to Extend Ranking and Skyline Queries. *Computer Science (Database)*, *13 Feb*, 1–7.

Milano, P. (2022d). Trying to bridge the gap between skyline and top-k queries. *Computer Science (Database)*, *March*, 1–12.

Mingyi, D., & Yinju, L. (2015). An Effective Uncertain Data Streams Top-K Query Algorithm. *The Open Automation and Control Systems Journal*, *7*(1), 1549–1553. https://doi.org/10.2174/1874444301507011549

Minh, T., Le, N., Cao, J., & He, Z. (2013). Data & Knowledge Engineering Top-k best probability queries and semantics ranking properties on probabilistic databases. *Data & Knowledge Engineering*, *88*, 248–266. https://doi.org/10.1016/j.datak.2013.04.005

Minh, T., Le, N., Ngoc, T., Nguyen, T., Hoang, D., & Ngoc, T. (2020). Uncertain data : Forms , Semantics , Processing Queries. *International Journal of Information and Knowledge Management (JIKM)*, *10*(1), 30–42.

Mo, L., Cheng, R., Li, X., Cheung, D., & Yang, X. (2013). Cleaning Uncertain Data for Top- k Queries. *2013 IEEE 29th International Conference on Data Engineering (ICDE)*, 134–145. https://doi.org/10.1109/icde.2013.6544820

Mohamud, M. A., Ibrahim, H., Sidi, F., Nurulain, S., Rum, M., Dzolkhifli, Z., Lawal, M. (2023). A Systematic Literature Review of Skyline Query Processing over Data Stream. *IEEE Access*, *PP*(January), 1. https://doi.org/10.1109/ACCESS.2023.3295117

Mouratidis, K., & Tang, B. (2018). Exact Processing of Uncertain Top-k Queries in Multi-criteria Settings. *The 44th International Conference on Very Large Data Bases*, *11*(8), 866–879. https://doi.org/10.14778/3204028.3204031

Moratidis, K., Tang, B., Mouratidis, K., Li, K., & Tang, B. (2021). Marrying top-k with skyline queries : Relaxing the preference input while producing output of controllable size. *Proceedings of the 2021 International Conference on Management of Data*, *June*, 1317–1330.

https://doi.org/10.1145/3448016.3457299

Nguyen, H. T. H., & Cao, J. (2014). Trustworthy answers for top-k queries on uncertain Big Data in decision making. *INFORMATION SCIENCES*, *318*(10), 73–90. https://doi.org/10.1016/j.ins.2014.08.065

Oosterhuis, H. (2020). Policy-Aware Unbiased Learning to Rank for Top- ? Rankings. *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 489–498. https://doi.org/10.1145/3397271.3401102

Papadopoulos, A. N., Tiakas, E., Tzouramanis, T., Georgiadis, N., & Manolopoulos, Y. (2021). Top-k Dominating Queries BT - Skylines and Other Dominance-Based Queries (Eds.) (pp. 63–90). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-031-01876-3_4

Payton, J., Julien, C., Rajamani, V., & Roman, G. (2012). Using snapshot query fidelity to adapt continuous query execution. *Pervasive and Mobile Computing*, *8*(3), 317–330. https://doi.org/10.1016/j.pmcj.2012.02.005

Pei, J., Hua, M., Tao, Y., & Lin, X. (2008). Query answering techniques on uncertain and probabilistic data. Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data. https://doi.org/10.1145/1376616.1376774

Peng, P., & Wong, R. C.-W. (2014). Geometry approach for k-regret query. 2014 *IEEE 30th International Conference on Data Engineering*. https://doi.org/10.1109/icde.2014.6816699

Peters, K. (2020). External Merge Sort for Top-K Queries Eager input filtering guided by histograms. *Distributed and Parallel Processing*, *27*, 2423–2437. https://doi.org/10.1145/3318464.3389729

Popov, S., & Kazanskiy, N. L. (2016). Software And Hardware Infrastructure For Data Stream Processing. *International Conference Information Technology and Nanotechnology 2016*, (January), 782–787. https://doi.org/10.18287/1613-0073-2016-1638-782-787

Qiao, B., Hu, B., Zhu, J., Wu, G., & Giraud-carrier, C. (2020). A top- k spatial join querying processing algorithm based on spark. *Information Systems*, *87*, 101419. https://doi.org/10.1016/j.is.2019.101419

Rahul, S., & Tao, Y. (2019). A Guide to Designing Top-k Indexes. *SIGMOD Record*, *48*(2), 6–17. https://doi.org/10.1145/3377330.3377332

Rai, N., & Lian, X. (2023). Distributed probabilistic top-k dominating queries over uncertain databases. *Knowledge and Information Systems*, *65*(11), 4939–4965. https://doi.org/10.1007/s10115-023-01917-3

Ren, W., Lian, X., & Ghazinour, K. (2021). Effective and efficient top- k query

processing over incomplete data streams. *Information Sciences*, *544*(August), 343–371. https://doi.org/10.1016/j.ins.2020.08.011

Rende, C., & Nancy, L. (2020). Uncertainty and Imprecision in Big Data Management : Models , Issues , Paradigms , and Future Research Directions. *Proceedings of the 2020 4th International Conference on Cloud and Big Data Computing*, *August*, 6–9. https://doi.org/10.1145/3416921.3416943

Sarma, A. Das, Benjelloun, O., Halevy, A., & Widom, J. (2006). Working Models for Uncertain Data ∗. *22nd International Conference on Data Engineering (ICDE'06)*. https://doi.org/10.1109/icde.2006.174

Shen, Z., Cheema, M. A., Lin, X., Zhang, W., & Wang, H. (2014). A Generic Framework for Top- k Pairs and Top- k Objects Queries over Sliding Windows. *IEEE Transactions on Knowledge and Data Engineering*, *26*(6). https://doi.org/10.1109/tkde.2012.181

Shi, T., Cai, Z., & Li, Y. (2022). Query Recombination: To Process a Large Number of Concurrent Top-k Queries towards IoT Data on an Edge Server. In *2022 IEEE 42nd International Conference on Distributed Computing Systems (ICDCS)* (pp. 559–569). https://doi.org/10.1109/ICDCS54860.2022.00060

Singh, A., Kempe, D., & Joachims, T. (2021). Fairness in Ranking under Uncertainty. *Advances in Neural Information Processing Systems*, *34*(2).

Song, C., Li, Z., & Ge, T. (2013). Top-K Oracle : A New Way to Present Top-K Tuples for Uncertain Data. *IEEE 29th International Conference on Data Engineering (ICDE)*, (24 June 2013), 146–157. https://doi.org/10.1109/icde.2013.6544821

Theobald, M., Bast, H., Majumdar, D., & Schenkel, R. (2008). TopX : efficient and versatile top- k query processing for semistructured data. *The VLDB Journal*, *17*, 81–115. https://doi.org/10.1007/s00778-007-0072-z

Vlachou, A., Doulkeridis, C., Nørvåg, K., Vazirgiannis, M., Management, H. D., & Query, S. (2008). On Efficient Top-k Query Processing in Highly Distributed Environments. *Proceedings of the International Conference on Management of Data (ICMD08), Vancouver (Canada)*, 753–764. https://doi.org/10.1145/1376616.1376692

Vouzoukidou, N. (2016). Continuous top-k queries over real-time web streams. Top- k Continues a Large- Echelle Continuous top-k queries over real-time web streams.

Wang, W., Wong, R. C., & Xie, M. (2021). Interactive Search for One of the Top-k. *Proceedings of the International Conference on Management of Data*, *June 2021*, 1920–1932. https://doi.org/10.1145/3448016.3457322

Wang, Yijie, Li, X., Li, X., & Wang, Y. (2013). A survey of queries over uncertain data, 485–530. https://doi.org/10.1007/s10115-013-0638-6

Wang, Yuxiang, Xu, X., Hong, Q., Jin, J., & Wu, T. (2021). Top- k star queries on knowledge graphs through semantic-aware bounding match scores. *Knowledge-Based Systems*, *213*, 106655. https://doi.org/10.1016/j.knosys.2020.106655

Xiao, G. (2015). Reporting l Most Favorite Objects in Uncertain Databases with Probabilistic Reverse Top- k Queries. *2015 IEEE International Conference on Data Mining Workshop (ICDMW)*, 1592–1599. https://doi.org/10.1109/ICDMW.2015.47

Xiao, G., & Li, K. (2015). Efficient top- (k , l) range query processing for uncertain data based on multicore architectures. *Distrib Parallel Databases*, *33*(April 2016), 381–413. https://doi.org/10.1007/s10619-014-7156-8

Xiao, G., Li, K., Zhou, X., & Li, K. (2016). Queueing Analysis of Continuous Queries for Uncertain Data Streams Over Sliding Windows. *International Journal of Pattern Recognition and Artificial Intelligence*, *30*(9), 1–16. https://doi.org/10.1142/S0218001416600016

Xiao, G., Li, K., Zhou, X., & Li, K. (2017). Journal of Computer and System Sciences Efficient monochromatic and bichromatic probabilistic reverse top- k query processing for uncertain big data. *Journal of Computer and System Sciences*, *89*, 92–113. https://doi.org/10.1016/j.jcss.2016.05.010

Xiao, G., Wu, F., Zhou, X., & Li, K. (2016). Probabilistic top-k range query processing for uncertain databases. *Journal of Intelligent & Fuzzy Systems*, *31*(2), 1109–1120. https://doi.org/10.3233/JIFS-169040

Xiao, N., Chen, T., Chen, L., & Tamer, M. O. (2013). Optimizing Multi-Top-k Queries over Uncertain Data Streams. *Transactions On Knowledge And Data Engineering*, *25*(8), 1814–1829. https://doi.org/10.1109/tkde.2012.126

Xie, M., & Wood, P. T. (2013). Efficient Top-k Query Answering using Cached Views. *ACM Journal*, (March 18-22), 489–500. https://doi.org/10.1145/2452376.2452433

Xie, Y., Tech, G., Chen, M., Tech, G., Dai, B., Tech, G., & Pfister, T. (2020). Differentiable Top- k with Optimal Transport. *Advances in Neural Information Processing Systems*, *33*(NeurIPS).

Xu, W., & Li, J. (2019). An improved algorithm for clustering uncertain traffic data streams based on Hadoop platform, *33*(19), 1–19. https://doi.org/10.1142/S0217979219502035

Yang, R., Zhou, Z., Tseng, L., Aloqaily, M., & Boukerche, A. (2020). Efficient

and Robust Top-k Algorithms for Big Data IoT. ICC 2020 - 2020 *IEEE International Conference on Communications (ICC)*, 60, 1–6. https://doi.org/10.1109/icc40277.2020.9148639

Ye, R. (2018). The Simulation of Open One-side Uncertain Probability for Fusion Model of Data Uncertainty and Data Relation Uncertainty. *2018 IEEE 3rd International Conference on Big Data Analysis (ICBDA)*, 97–101. https://doi.org/10.1109/ICBDA.2018.8367658

Zarko, I. P., & Zi, P. (2015). Time- and Space-Efficient Sliding Window Top-k Query Processsing. *ACM Transactions on Database Systems*, *40*(1), 1–44. https://doi.org/10.1145/2736701

Zhang, C. J., Chen, L., Tong, Y., & Liu, Z. (2015). Cleaning uncertain data with a noisy crowd. 2015 IEEE 31st International Conference on Data Engineering. https://doi.org/10.1109/icde.2015.7113268

Zhang, B., & Wang, W. H. (2020). AuthPDB : Authentication of Probabilistic Queries on Outsourced Uncertain Data. *CODASPY '20: Proceedings of the Tenth ACM Conference on Data and Application Security and Privacy*, *March 2020*, 121–132. https://doi.org/10.1145/3374664.3375731

Zhang, J., Tang, B., Yiu, M. L., Yan, X., & Li, K. (2022). T - LevelIndex : Towards Efficient Query Processing in Continuous Preference Space. *Proceedings of the 2022 International Conference on Management of Data*, *June 2022*, 2149–2162. https://doi.org/10.1145/3514221.3526182

Zhang, Z., Xie, X., & Pan, H. (2018). An Efficient Optimization Approach for Top-k Queries on Uncertain Data. *International Journal of Cooperative Information Systems*, *27*(01). https://doi.org/10.1142/S0218843017410027

Zhao, K., Tao, Y., & Zhou, S. (2007). Efficient top- k processing in large-scaled distributed environments. *Data & Knowledge Engineering*, *63*, 315–335. https://doi.org/10.1016/j.datak.2007.03.012

Zheng, Z., Zhang, M., Yu, M., Li, D., & Zhang, X. (2021). User preference-based data partitioning top-k skyline query processing algorithm. In *2021 IEEE International Conference on Industrial Application of Artificial Intelligence (IAAI)* (pp. 436–444). https://doi.org/10.1109/IAAI54625.2021.9699888