Noname manuscript No.
(will be inserted by the editor)

Approximate Spatio-Temporal Top-k Publish/Subscribe

Lisi Chen · Shuo Shang

Received: / Accepted:

Abstract Location-based publish/subscribe plays a significant role in mobile information disseminations. In this light, we propose and study a novel problem of processing location-based top-k subscriptions over spatio-temporal data streams. We define a new type of approximate location-based top-k subscription, Approximate Temporal Spatial-Keyword Top-k (ATSK) Subscription, that continuously feeds users with relevant spatio-temporal messages by considering textual similarity, spatial proximity, and information freshness. Different from existing location-based top-k subscriptions, Approximate Temporal Spatial-Keyword Top-k (ATSK) Subscription can automatically adjust the triggering condition by taking the triggering score of other subscriptions into account. The group filtering efficacy can be substantially improved by sacrificing the publishing result quality with a bounded guarantee. We conduct extensive experiments on two real datasets to demonstrate the performance of the developed solutions.

Keywords Publish/Subscribe · Subscription · Location · Stream

1 Introduction

With the development of Location-based services (LBS), user generated content on the Web has been increasingly associated with geo-locations. Massive amount of data that contain both text information and geographical location information are being generated at an unprecedented scale on the Web. For

Lisi Chen
University of Wollongong, Australia
E-mail: lisi@uow.edu.au

Shuo Shang
King Abdullah University of Science and Technology, Saudi Arabia
E-mail: jedi.shang@gmail.com

Fig. 1 Tweet with Location

example, Tweets, each containing no more than characters, can be associated with locations as illustrated in Figure 1, and some social photo sharing websites (e.g., Instagram) contain photos with both textual description tags and location.

We refer to such data with text, location, and timestamp as spatio-temporal messages. These spatio-temporal messages cover a wide range of topics. For example, Tweets often offer the quickest first-hand reports of news events [49], comments and reviews representing public idea, bursty events, etc. Users are interested in receiving up-to-date tweets such that their locations are close to a user specified location and their texts are interesting to users [4, 49]. For example, a user may want to be updated with tweets near her home on the topic "dengue spread," or a user may be interested in tweets whose locations are close to her office and are related to "ice-cream sales" As another application example, a user may be interested in recent tweets whose locations are close to a POI and whose text is relevant to the description of the POI; furthermore, a POI service provider (e.g., Yelp) may want to annotate each POI with its up-to-date relevant tweets for providing users a better service. In these applications, users may not want to be overwhelmed by a large number of tweets. Instead, users would prefer to being updated with the top-k most relevant tweets in terms of distance, text relevance and recency, which is also one of the key indicators of relevance for tweets.

To solve this problem, some existing studies [4, 51] develop location-based top-k publish/subscribe systems that focus on processing a large number of location-based top-k subscriptions over geo-textual data streams. Specifically, such location-based top-k subscription is treated as a continuous query and tweets are treated as published spatio-temporal messages. The subscription takes into account the following three aspects in evaluating the relevance with a spatio-temporal message: (1) text similarity; (2) spatial proximity; and (3) freshness of spatio-temporal message. The location-based top-k subscription continuously maintains its up-to-date top-k results over a stream of spatio-temporal messages. A subscription is triggered by a new published message if and only if the new message scores higher than the current k-th top result message. As a result, when a new message arrives we need to evaluate each subscription and compute their relevance score to check if their relevance score is higher than the k-th result maintained by the subscription. Such one-by-one evaluation is very time-consuming especially when the number of subscriptions is very large (i.e., more than 1M). To enable the group evaluation, existing studies develop a grouping filtering technique [4] or hybrid indexing

scheme [51] to accelerate the subscription matching process. Nevertheless, the improvement of performance of processing a large number of location-based top-k subscriptions is still limited. Consequently, we need to develop a more efficient way to process such large set of subscriptions with excellent scalability.

To address the challenge, we define a new type of approximate location-based top-k subscription, Approximate Temporal Spatial-Keyword Top-k (ATSK) Subscription. Instead of having a static triggering condition as the original location-based top-k subscription (i.e., the new message scores higher than the current k-th top result message), the triggering condition for ATSK is dynamic. In particular, it can automatically adjust the score of triggering condition by taking the score of other subscriptions into account. As a result, the group filtering efficacy can be substantially improved by sacrificing the publishing result quality. Note that the approximate location-based top-k subscription enables users to pre-define an approximation ratio to guarantee the publishing result quality.

For efficiently maintaining the up-to-date results of a large number of ATSK subscriptions over a stream of spatio-temporal messages, we define Approximate Conditional Influence Ring (ACIR) to represent each ATSK subscription. Based on the concept of ACIR, we develop hybrid indexing structure to group similar subscriptions and generate effective group filtering conditions to help filter out a group of subscriptions when a new message arrives. The proposed technique is featured by the following key technical components. 1) We propose a new concept to describe each ATSK subscription, namely the Approximate Conditional Influence Ring, based on which we develop an approach to determining whether a new spatio-temporal message is a result of a ATSK subscription. 2) We propose the Approximate Conditional Influence Ring based Quad-tree indexing structure for organizing the ATSK subscriptions based on the ACIRs of subscriptions. 3) We develop a spatial-aware inverted file technique to organize the ATSK subscriptions associated with a spatial cell in the CIQ-tree.

The rest of this paper is organized as follows. Section 1 introduces preliminaries and framework of processing ATSK subscriptions. Section 2 defines the ATSK subscription. Section 3 details our proposed solution. Section 4 presents the experimental studies. Section 5 reviews the related work, and Section 6 concludes the paper.

2 Problem Statement

We introduce the spatio-temporal message, Exact Temporal Spatial-Keyword Top-k Subscription (ETSK), and Approximate Temporal Spatial-Keyword Top-k (ATSK) Subscription. In particular, spatio-temporal messages and ETSK subscriptions are defined based on existing work [4].

Definition 1 Spatio-temporal Message. A spatio-temporal message is represented with a triple $m = \langle \psi, \rho, t_c \rangle$, where $m.\psi$ is a set of terms, $m.\rho$ is a

geographical point location with latitude and longitude, and t_c is the creation time of message m.

In this paper, we consider a stream of spatio-temporal messages. For instance, it can be geo-tagged tweets in Twitter, geo-tagged photos with tags in Instagram, check-ins with text descriptions in Foursquare, geo-tagged webpages, etc.

Intuitively, given a stream of spatio-temporal messages, an Exact Temporal Spatial-Keyword Top-k Subscription (ETSK) is to continuously retrieve k messages over time such that these messages are temporally most relevant to the subscription (i.e., their textual information is highly relevant to the subscription keywords, their locations are close to the subscription location, and they are fresh). Notice that apart from text similarity and spatial proximity, freshness is important for spatio-temporal data streams. For example, tweets are inclined to refer to some specific trending event and their relevance to a subscription declines as time elapses. Next, we define Approximate Temporal Spatial-Keyword Top-k Subscription (ATSK), which maintains an approximate result set of k most relevant messages over spatio-temporal data streams.

2.1 Exact Temporal Spatial-Keyword Top-k Subscription

We present the concept of *triggering condition*, which is regarded as a preliminary of publish/subscribe systems, and the definition of ETSK subscription.

Definition 2 Triggering Condition. Let s be a subscription, m be a message from data streams, and $B(m, s)$ be a Boolean expression. If m will be delivered to s if and only if $B(m, s)$ is true, $B(m, s)$ is considered to be the triggering condition of s.

Next, based on existing work [4] we define Exact Temporal Spatial-Keyword Top-k Subscription (ETSK) and its triggering condition.

Definition 3 Exact Temporal Spatial-Keyword Top-k Subscription (ETSK). An ETSK subscription $s = \langle \psi, \rho, k \rangle$, contains a set of subscription keywords/terms, $s.\psi$, a location $s.\rho$, and the number of messages to be maintained as results, $s.k$. The messages returned at time t_e are ranked according to the temporal spatial-keyword similarity score, which is defined by

$$S_{tsk}(m, s, t_e) = S_{sk}(m, s) \cdot S_t(m.t_c, t_e) \tag{1}$$

where $S_{sk}(m, s)$ computes the spatial-keyword similarity between subscription s and object m and $S_t(m.t_c, t_e)$ computes the message freshness.

When a new message m_n arrives, an ETSK s will be triggered by m_n if:

$$S_{tsk}(m_n, s, t_{cur}) \geq S_{tsk}(m_e, s, t_{cur}), \tag{2}$$

where m_e denotes the message with the k-th highest spatial-keyword similarity score in the result set of s.

Following previous work (e.g., [5]) we compute the spatial-keyword similarity $S_{sk}(m, s)$ between s and m as follows.

$$S_{sk}(m, s) = \alpha \cdot SD(m.\rho, s.\rho) + (1 - \alpha) \cdot TR(m.\psi, s.\psi), \tag{3}$$

where $SD(m.\rho, s.\rho)$ is the *proximity score* between subscription s and message m, $TR(m.\psi, s.\psi)$ indicates the *text similarity* between m and s, and $\alpha \in (0, 1)$ denotes a preference parameter that balances the weight between spatial proximity and text similarity. The proximity score is computed by the normalized Euclidian distance: $SD(m.\rho, s.\rho) = 1\text{-}\frac{dist(m.\rho, s.\rho)}{dist_{max}}$,where $dist(m.\rho, s.\rho)$ is the Euclidian distance between m and s, and $dist_{max}$ can be the maximal possible distance in the spatial area. The text similarity is calculated by using an information retrieval model, such as language models [11], cosine similarity [33], or BM25 [10], and is normalized to a scale between 0 and 1. Here we use language models because it is originally defined for modeling the relevance between a keyword-based query and a textual document.

The freshness of message m is calculated by exponential decay function, which is defined as Equation 4:

$$S_t(m.t_c, t_e) = D^{-(t_e - m.t_c)}, \tag{4}$$

where D is base number that determines the rate of the freshness decay. The function is monotonically decrease with $t_e - m.t_c$. It is introduced in [20] and is applied [1, 5, 22] as the measurement of recency for streaming data. Based on the experimental studies [12], the exponential decay function has been shown to be effective in blending the freshness and text similarity of textual documents.

2.2 Approximate Temporal Spatial-Keyword Top-k Subscription

We proceed to define Approximate Temporal Spatial-Keyword Top-k Subscription (ATSK) and its triggering condition.

Definition 4 Approximate Temporal Spatial-Keyword Top-k Subscription (ATSK). An ATSK subscription $s = \langle \psi, \rho, k \rangle$, contains a set of terms, $s.\psi$, a location $s.\rho$, and the number of messages to be maintained as results, $s.k$. The messages returned at time t_e are ranked according to the temporal spatial-keyword similarity score, which is defined by

$$S_{tsk}(m, s, t_e) = S_{sk}(m, s) \cdot S_t(m.t_c, t_e) \tag{5}$$

where $S_{sk}(m, s)$ computes the spatial-keyword similarity between subscription s and message m and $S_t(m.t_c, t_e)$ computes the message freshness.

Let $AS_{tsk}(m, s, t_{cur})$ be the approximate spatial-keyword similarity score computed by our approximate subscription matching algorithm. When a new message m_n arrives, whether s will be triggered by m_n is dependent by the spatial-keyword similarity score between s and m_n, which is classified by the following scenarios:

(1) If $AS_{tsk}(m, s, t_{cur}) \geq (1 + \epsilon) \times S_{tsk}(m, s, t_{cur})$, m will be delivered to s;

(2) If $(1-\epsilon) \times S_{tsk}(m, s, t_{cur}) \leq AS_{tsk}(m, s, t_{cur}) < (1+\epsilon) \times S_{tsk}(m, s, t_{cur})$, m will either be delivered to s or be filtered out by s;

(3) If $AS_{tsk}(m, s, t_{cur}) \leq (1 - \epsilon) \times S_{tsk}(m, s, t_{cur})$, m will be filtered out by s.

In our applications, the typical arrival rate of spatio-temporal messages (e.g., tweets) is in the scale of millions a day, while new ETSK or ATSK subscriptions are added at the rate of tens of thousands a day, and we may serve millions of ETSK or ATSK subscriptions at one time. We thus aim to develop a scalable solution to maintain the up-to-date results for a large number of ETSK or ATSK subscriptions over a data stream of spatio-temporal messages. It is possible for millions of subscriptions to be fitted into the available memory of modern servers. Hence, our solution is developed for in-memory setting. In the rare case that the ETSK or ATSK subscriptions cannot fit into memory, we can employ our proposed solution on multiple servers, each processing a subset of ETSK or ATSK subscriptions independently.

We would like to minimize the following costs: (1) CPU cost for subscription matching, which can be regarded as the runtime for finding the set of subscriptions that can include an incoming message as their top-k results; (2) CPU cost for result update, which is the time spent for updating the result of each subscription when a message arrives.

3 Framework for ATSK Subscription Processing

3.1 Preliminaries

We aim at the problem of continuous delivery of up-to-date top-k spatio-temporal messages for a large number of ATSK subscriptions over a stream of spatio-temporal messages. Before presenting our method to process ATSK subscriptions, we first introduce the baseline method and an advanced method to process ETSK subscriptions developed by Chen et al. [4].

Notation	Description
$SD(m.\rho, s.\rho)$	spatial proximity score between m and s
$TR(m.\psi, s.\psi)$	text similarity score between m and s
$S_{tk}(m, s)$	spatial-keyword similarity between m and s
$S_{tsk}(m, s, t)$	temporal spatial-keyword similarity between m and s at time t
$AS_{tsk}(m, s, t)$	approx. temporal spatial-keyword similarity between m and s at time t
t_{cur}	current timestamp

Table 1 Table of frequently used notations

3.1.1 Baseline of ETSK *processing*

A straightforward approach for processing ETSK works as follows: For each new spatio-temporal message m (i.e., arriving at timestamp t_{cur}) compute its temporal spatial-keyword similarity score with each subscription s, $S_{tsk}(m, s, t_{cur})$; If the score is greater than the score with the temporal spatial-keyword similarity score of the k-th result of subscription s (at timestamp t_{cur}), message m will be used to update the top-k results for s. Note that the temporal spatial-keyword similarity score of the k-th result of subscription s (at timestamp t_{cur}) declines as time passes. As a consequence, for each new message we need to compute its score with respect to each subscription, and for each subscription we need to re-compute the temporal spatial-keyword score of its k-th result. Therefore, the straightforward approach is computationally prohibitive especially when the number of subscription is huge and the spatio-temporal messages arrive at a high rate.

3.1.2 Advanced method of ETSK *processing*

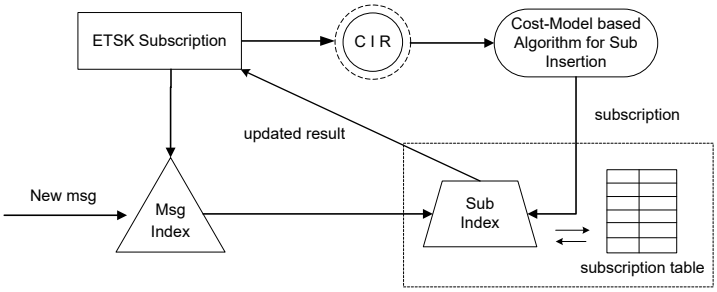


Fig. 2 Framework for Processing ETSK subscriptions

Chen et al. [4] develop a more efficient mechanism to process ETSK subscriptions over spatio-temporal message streams, where both new subscriptions and new messages arrive in a streaming manner. Figure 2 shows our proposed architecture for processing ETSK subscriptions. To index the spatial aspect of ETSK subscriptions, each subscription is represented by a set of Conditional Influence Regions (CIRs) based on the Quad-tree cells. The CIRs of all ETSK subscriptions are indexed by a CIQ-tree. Each ETSK subscription maintains the current top-k results over the stream of spatio-temporal messages. It is challenging to develop an index that can be used to efficiently maintain the up-to-date results for ETSK subscriptions over a stream of spatio-temporal messages because the indexing scheme is required to take spatial, textual, and temporal aspects into account while computing the similarity score of a spatio-temporal message for a subscription.

The cost model for subscription insertion is used to decide how to index each subscription into the CIQ-tree. The cost model for subscription update

is used to decide how to maintain the CIQ-tree in response to the update of the top-k results for a ETSK subscription. The CIQ-tree and the corresponding cost models are the main components developed by Chen et al. [4]. The subscription table maintains the basic information of all subscriptions and their current results, including the subscription location, subscription keywords, subscription preference parameter α, and the minimum temporal spatial-keyword similarity score in the results of the subscription. Additionally, a complementary index for spatio-temporal messages is maintained for initializing the top-k result when a new subscription arrives. The problem of efficiently indexing spatio-temporal messages is beyond the scope of this work.

3.2 Overview of ATSK Subscription Processing

The difference between ETSK and ATSK subscriptions is that an ATSK subscription will be triggered by a new message whose temporal spatial-keyword similarity score is ϵ-approximately greater than the k-th result of the subscription. While an ETSK subscription will be triggered by a new message whose temporal spatial-keyword similarity score is *exactly* greater than the k-th result of the subscription. The ϵ-approximation property of the ATSK subscription allows us to develop a more effective group filtering technique by taking advantage of an approximate filtering condition generated by the ϵ-approximation property. The major challenge here is to develop an effective way to represent each ATSK subscription and to propose corresponding group filtering condition with much more filtering power in comparison to that of an ETSK subscription.

Figure 3 illustrates the framework of processing ATSK subscriptions. We can see that the procedure of processing ATSK subscriptions is similar to the processing of ETSK subscriptions [4] but with the following differences:

(1) *Subscription representation.* Because the triggering conditions of ATSK subscription and ETSK subscription are different, the conditional influence region defined by Chen et al. [4] is inapplicable. Consequently, we need to develop a new way to represent the spatial, textual, and temporal aspects of each ATSK subscription. Here we propose the *approximate conditional influence ring* to define each ATSK;

(2) *Group filtering technique.* Because of the differences of triggering conditions, a new grouping mechanism is required to be developed for enabling the group filtering of a batch of ATSK subscriptions when a new message arrives. In addition, we need to generate ATSK based group filtering conditions accordingly.

4 Representing ATSK Subscriptions

In this section, we first present the concept of *approximate conditional influence ring*. Next we introduce how to represent an ATSK subscription by using a set of approximate conditional influence rings.

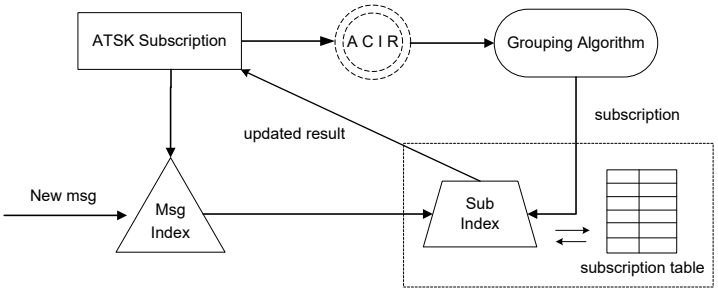


Fig. 3 Framework for Processing ETSK subscriptions

4.1 Approximate conditional influence rings

We propose the concept of Approximate Conditional Influence Ring (ACIR) to answer the ATSK subscription.

Our idea is inspired by the concept of *influence region* that is widely employed for processing continuous k nearest neighbor (CkNN) queries. Each CkNN is associated with a circular influence region that is centered at the query location, and whose radius is the distance from the query location to its kth nearest neighbor.

Based on the influence regions, they build different types of spatial index (e.g. grid, quad-tree, and etc.) to store those regions for matching an geographical object with the queries. For example, if we use the grid index to store the influence regions, when a message m arrives, we just need to search in the gird cell c where m is located and check the queries of which influence regions have overlapping areas with c. If a spatial object falls in the influence region of a subscription, the object becomes a result of the subscription; otherwise, it cannot be a result. The influence regions of CkNN queries can be organized by a spatial indexing technique (e.g., grid cell), and we can easily find the queries for which a new object is a result.

Such method can avoid a great proportion of unnecessary subscription evaluations and is used by most of the techniques for processing CkNN queries. For the textual part, inverted file is the most efficient indexing scheme for text information retrieval, which is widely used and optimized to solve various types of text-relevant problems.

However, the influence region cannot be directly used for representing ETSK and ATSK subscriptions. To represent an ETSK subscription, Chen et al. [4] use *conditional influence region (CIR)* for the propose. Specifically, each ETSK subscription is represented by a set of CIRs according a simple spatial-partitioning based indexing structure (i.e.,quad-tree). Each CIR denotes an influence region regarding a particular score of textual similarity and freshness. With the help of CIRs, irrelevant incoming messages can be filtered out in advance.

Recall that the triggering condition of ETSK is based on an exact temporal spatial-keyword similarity score (i.e., the score of the k-th message in the result

set), while the triggering condition of ATSK is based on a temporal spatial-keyword similarity score range. Consequently, it is impossible to generate a set of CIRs for each ATSK subscription. To solve this problem, we propose *approximate conditional influence ring (ACIR)* to represent each ATSK subscription. The major difference between a CIR and an ACIR is that an ACIR has two boundary circles (i.e., an inner circle and an outer circle) while CIR just has one boundary circle.

Next, we formally define the approximate conditional influence ring, and then explain how it may help process the ATSK subscription.

Definition 5 Approximate Conditional Influence Ring. Let R_q be the list of up-to-date top-k spatio-temporal messages for an ATSK subscription s. The approximate conditional influence ring of s at time t and textual similarity score tr is denoted by a ring centered at the location of s. In particular, the radius of the inner circle of the ring is $\mathsf{r}_s^{in}(tr,t)$ and the radius of the outer circle of the ring is $\mathsf{r}_s^{out}(tr,t)$.

$$\mathsf{r}_s^{in}(tr,t) = (1-\epsilon)\cdot(\frac{S_{sk}(q,R_q[k])\cdot b^{-(t-R_q[k].c_t)}}{\alpha} - \frac{1-\alpha}{\alpha}\cdot tr)dist_{max}, \quad (6)$$

$$\mathsf{r}_s^{out}(tr,t) = (1+\epsilon)\cdot(\frac{S_{sk}(q,R_q[k])\cdot b^{-(t-R_q[k].c_t)}}{\alpha} - \frac{1-\alpha}{\alpha}\cdot tr)dist_{max}, \quad (7)$$

where $R_q[k]$ is the kth result of subscription s. The approximate conditional influence ring is denoted by $\mathsf{ACI}_q(tr,t_{cur})$.

Like the influence region of CkNN query, the approximate conditional influence ring is expected for narrowing the scope of subscription evaluation when organized by proposed indexing scheme.

Property Consider a new spatio-temporal message m arriving at time t and suppose that its the text relevance to a ATSK subscription s is tr. Message m becomes a top-k result of s iff m falls in the approximate conditional influence ring $\mathsf{ACI}_q(tr,t_{cur})$.

From Equations 6 and 7, we can see that $\mathsf{r}_s^{in}(tr,t)$ and $\mathsf{r}_s^{out}(tr,t)$ increase as the time elapses, and a greater value of tr results in the greater values of $\mathsf{r}_s^{in}(tr,t)$ and $\mathsf{r}_s^{out}(tr,t)$. Example 1 demonstrates the approximate conditional influence ring.

Example 1 Let m_0, m_1 and m_2 be three spatio-temporal messages such that $tr_0 = TR(m_0.\psi, s.\psi)$, $tr_1 = TR(m_1.\psi, s.\psi) = TR(m_2.\psi, s.\psi)$, and $tr_0 < tr_1$. Assume that t_0 and t_1 are two timestamps and $t_0 < t_1$. Table 1 and Figure 4 illustrate the influence regions of subscription s under different timestamps and different values of text similarity. We observe that region radius becomes larger as time elapses and a greater value of tr corresponds to a greater radius.

From Figure 4 we can see that the approximate conditional influence rings are based on the text relevance of tr_0 at time t and t' respectively, and the corresponding radii of inner/outer radii of approximate conditional influence

Outer Radius	Inner Radius	Timestamp	Text Similarity
r_0^{out}	r_0^{in}	t	tr_0
$r_0^{out\prime}$	$r_0^{in\prime}$	t'	tr_0
r_1^{out}	r_1^{in}	t	tr_1
$r_1^{out\prime}$	$r_1^{in\prime}$	t'	tr_1

Table 2 Approximate Conditional Influence Rings of Subscription s

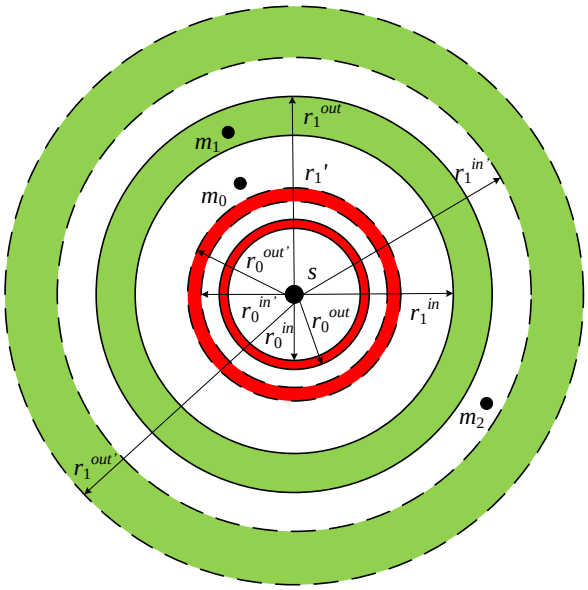


Fig. 4 Approximate Conditional Influence Rings of Subscription s

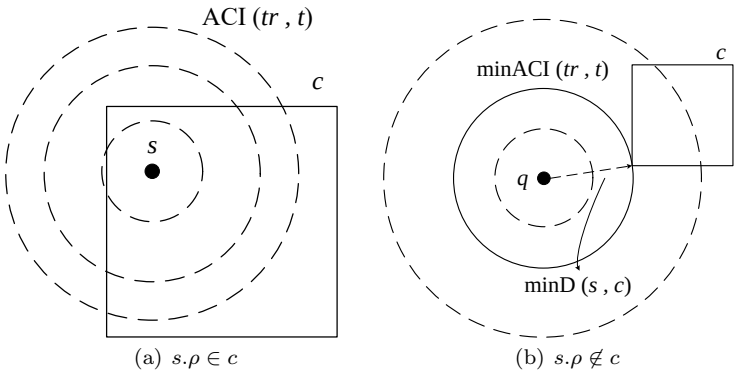
rings are shown in Table 1. We find that m_0 falls outside the outer circle of $\mathsf{ACI}(tr_0, t_0)$ and $\mathsf{ACI}(tr_0, t_1)$. As a result, m_0 cannot be a result of s at timestamps t_0 or t_1. We also find that m_1 falls between in inner and outer circles of $\mathsf{ACI}(tr_1, t_0)$. Consequently, m_1 may or may not be a result of s at timestamps t_0. While if m_1 arrives at timestamp t_1, it will definitely be a result of s because m_1 located inside the inner circle of $\mathsf{ACI}(tr_1, t_1)$.

4.2 Utilizing Approximate Conditional Influence Rings

According to Definition 5, the radii of ACIRs of a subscription s depend on not only the spatial-keyword similarity score between s and its k-th result (in R_s), but also the text similarity between s and an incoming spatio-temporal message m and the arrival time of m. This makes it inapplicable to generate an ACIR and then use the region to decide whether an incoming spatio-temporal message is a new result of s (in the similar way as the influence region is used for the CkNN subscription).

To this end, we propose a novel way of generating conditional influence regions and utilizing them for processing ATSK subscriptions. Our high-level idea comprises three steps: 1) for an ATSK subscription s, and a spatial cell c, we generate two circle regions, whose radiuses are the minimum distance between s and cell c, and the maximum distance between s and c, respectively; 2) for each of the two circles, we generate a conditional influence region, and compute its corresponding text relevance score tr with respect to a past timestamp, e.g., the creation time ($R_s[k].c_t$) of the kth result message in R_s; 3) for a new spatio-temporal message m that arrives at time t and falls in the cell c, we design an approach that is able to "map" the spatial-textual score between m and subscription s at time t to a backdated equivalent score at time $R_s[k].c_t$; this enables us to compare the backdated equivalent score with the scores computed at step 2) to decide whether m is a result of s. We proceed to detail the three steps.

Step 1 Given a spatial cell c and an ATSK subscription s, we generate the following two circle regions: (1) minimum circle: its radius is the the minimum distance between s and cell c, and the radius is denoted by $\mathsf{minD}(s, c)$. When subscription s is located in cell c, $\mathsf{minD}(s, c) = 0$; (2) maximum circle: its radius is the maximum distance between s and any location in cell c, and the radius is denoted by $\mathsf{maxD}(s, c)$. Figure 5(a) illustrates the two circles when s is located within c and Figure 5(b) illustrates the case when s is not located within c.



(a) $s.\rho \in c$ (b) $s.\rho \notin c$

Fig. 5 Two inner circles for given s and c

Step 2 Based on the two circles generated in Step 1, we generate two conditional influence regions. Recall that the text similarity tr and the time t are given and we compute the radius of the corresponding inner circle. In contrast, here we know the radius ($\mathsf{minD}(s, c)$ or $\mathsf{maxD}(s, c)$) and the time is the creation time ($R_s[k].c_t$) of the k-th result message in R_s, and we want to compute the corresponding text relevance score.

Definition 6 Minimum Approximate Conditional Text Similarity and Maximum Approximate Conditional Text Similarity. Given subscription s, cell c, time $R_s[k].c_t$, and their minimum circle $\mathsf{minD}(s,c)$, based on Equation 12, the corresponding text relevance score is computed by

$$\mathsf{minAT}(s,c) = \frac{1}{1-\epsilon} \cdot (\frac{\mathsf{S}_{sk}(s,t_b)}{1+\alpha} + \frac{\alpha \cdot \mathsf{minD}(s,c)}{1+\alpha}). \tag{8}$$

The text similarity score is called *minimum approximate conditional text similarity*. Given subscription s, cell c, time $R_s[k].c_t$, and their maximum inner circle $\mathsf{maxD}(s,c)$, based on Equation 12, the corresponding text similarity score is computed by

$$\mathsf{maxAT}(s,c) = \frac{1}{1-\epsilon} \cdot (\frac{\mathsf{S}_{sk}(s,t_b)}{1+\alpha} + \frac{\alpha \cdot \mathsf{maxD}(s,c)}{1+\alpha}), \tag{9}$$

The text similarity score is called *maximum approximate conditional text similarity*.

Step 3 This step is to utilize the *minimum approximate conditional text similarity* and *maximum approximate conditional text similarity* to check whether a new spatio-temporal message m is a result for subscription s if m falls in the spatial area of c. Recall that the two inner circles of approximate conditional influence rings correspond to the timestamp $R_s[k].c_t$. However, the arrival time of m, denoted by t_{cur}, is different from $R_s[k].c_t$, and thus the spatial-keyword similarity score between m and s is not comparable with the score between m and its k-th results at time $R_s[k].c_t$.

To make the two scores comparable, our idea is to design a method that can be used to "map" the spatial-textual score between m and subscription s at time t_{cur} to a backdated equivalent score at time $R_s[k].c_t$; and then we are able to compare the backdated equivalent score with the scores computed at step 2) to decide whether m is a result of s. The mapping is defined as follows.

Definition 7 Backdated Equivalent Text Relevance Score. Given an ATSK subscription s at time t_{cur}, the backdated equivalent text relevance score at time t_b is defined by

$$\mathsf{TR}^{t_b}(m.\psi, s.\psi) = \mathsf{TR}(m.\psi, s.\psi) \cdot D^{t_{cur}-t_b} \tag{10}$$

With the mapped score, we have the following lemma to determine whether m is a result.

Lemma 1 *Let m be an incoming spatio-temporal message located in the spatial cell c. We have (1) when $\mathsf{TR}^{t_b}(m.\psi, s.\psi) < \mathsf{minAT}(s,c)$, m is not be a result of s; (2) when $\mathsf{TR}^{t_b}(m.\psi, s.\psi) \geq \mathsf{maxAT}(s,c)$, m must be a new result of s; (3) when $\mathsf{minAT}(s,c) \leq \mathsf{TR}^{t_b}(m.\psi, s.\psi) < \mathsf{maxT}(s,c)$, m may be a result of s.*

Proof We can easily proof this lemma based on Definitions 5 and 8.

4.2.1 Mapping to backdated equivalent scores at a uniform time

Our three step idea and the mapping technique enable us to utilize the ACIR to check whether a message is a result of subscription s. Recall for each subscription we need to map its spatial-keyword similarity with a new message m back to the similarity value at the creation time of its kth result $R_s[k].c_t$. However, we need to handle a large number of subscriptions, rather than a single subscription, and the k-th results of these subscriptions may have different timestamps. Hence, it is expensive to map the spatial-keyword similarity to each of different timestamps. To address the problem, our idea is to map Approximate Minimum Conditional Text Similarity and Approximate Maximum Conditional Text Similarity of all the subscriptions to their backdated equivalent scores at a single backdated time, denoted by t_b. Different subscriptions may have different values of $t_u(s)$, which make it difficult to compute the text similarity required for updating each subscription (since we have to compute them one by one based on unique $t_u(s)$ for each subscription). So we convert $minACIR(s, c, t_u)$ and $maxACIR(s, c, t_u)$ of different subscriptions into $minACIR(s, c, t_b)$ and $maxACIR(s, c, t_b)$ respectively by using the backdated equivalent subscription update threshold, which is defined in Definition 8, instead of the subscription update threshold, where t_b is a subscription-independent backdated time.

Definition 8 Backdated Equivalent Score. Let s be an ATSK subscription, the backdated equivalent subscription update threshold at time t_b is defined as Equation 11

$$BUT(s, t_b) = UT(s) \cdot D^{t_{cur}-t_b} \tag{11}$$

Based on Equation 11, we map the update thresholds of all subscriptions to a previous predefined timestamp t_b and use $BUT(s, t_b)$ to replace $UT'(s)$, that is:

$$\frac{R_{ACIR}(s, tr, t)}{dist_{max}} = \frac{BUT(s, t_b) \cdot D^{t_b-t}}{\alpha} - \frac{1-\alpha}{\alpha} \cdot tr. \tag{12}$$

Since we have $minACIR(s, c, t_b)$ and $maxACIR(s, c, t_b)$ with a unified timestamp t_b, we can derive their corresponding values of text relevance based on Equation 12. Consequently, we have the following definition.

Definition 9 Minimum and Maximum Border Score. The minimum border score of subscription s under cell c at time t_b, denoted by $BS_{min}(s, c, t_b)$, is the corresponding text relevance w.r.t. $minACIR(s, c, t_b)$. And denoted by $BS_{max}(s, c, t_b)$ the maximum border score of s under c is the corresponding text relevance w.r.t. $maxACIR(s, c, t_b)$. Equation 13 and 14 compute $BS_{min}(s, c, t_b)$ and $BS_{max}(s, c, t_b)$ respectively.

$$BS_{min}(s, c, t_b) = \frac{BUT(s, t_b) \cdot D^{t_b-t_{cur}}}{1+\alpha} + \frac{\alpha \cdot R_{minACIR}(s, c, t_b)}{1+\alpha}, \tag{13}$$

$$BS_{max}(s, c, t_b) = \frac{BUT(s, t_b) \cdot D^{t_b-t_{cur}}}{1+\alpha} + \frac{\alpha \cdot R_{maxACIR}(s, c, t_b)}{1+\alpha}, \tag{14}$$

where $R_{minACIR}(s, c, t_b)$ and $R_{maxACIR}(s, c, t_b)$ denote the radius of $minACIR(q, c, t_b)$ and $maxCIR(s, c, t_b)$ respectively.

For each subscription s, we select a set of cells, which may from different layers of the quad-tree, to generate corresponding $BS_{min}(s, c, t_b)$ and $BS_{max}(s, c, t_b)$ respectively on each selected cell c. Now we proceed to discuss how to store a subscription on a selected cell.

Based on Equation 13 (resp. 14), $BS_{min}(q, c, t_b)$ (resp.$BS_{max}(q, c, t_b)$) can be regarded as the sum of the following two parts, namely $BUT(q, t_b) \cdot D^{t_b - t_{cur}}/(1+\alpha)$ and $\alpha \cdot R_{minCIR}(s, c, t_b)/(1 + \alpha)$, where $BUT(s, t_b) \cdot D^{t_b - t_{cur}}/(1 + \alpha)$ is dependent on the current time t_{cur} and $\alpha \cdot R_{minCIR}(s, c, t_b)/(1 + \alpha)$ (resp. $\alpha \cdot R_{maxCIR}(s, c, t_b)/(1 + \alpha)$) is time-independent.

We separately store the following values for each subscription s on each relevant cell c and then organize them together with hierarchical based inverted file: (1) $BUT(s, t_b)/(1 + \alpha)$; (2) $\alpha \cdot R_{minCIR}(s, c, t_b)/(1 + \alpha)$; (3) $\alpha \cdot R_{maxCIR}(s, c, t_b)/(1 + \alpha)$. Notice that we do not maintain the real-time values of the border scores, instead, we store the border scores at a pre-defined backdated time t_b, just as (1), and calculate the current border scores as soon as a new message arrives. The reason is that it is impractical to maintain the real-time values of $BS_{min}(s, c, t_b)$ and $BS_{max}(s, c, t_b)$.

4.3 IQ*-Tree

We proceed to describe the structure of the IQ*-tree, which is a hybrid spatial keyword indexing structure and basically consists of the quad-tree and hierarchical inverted file. It is used for indexing conditional influence regions of the ATSK subscription.

After having $BS_{min}(s, c, t_b)$ and $BS_{max}(s, c, t_b)$ for the subscription s under cell c, we notice that incoming spatio-temporal messages may fall in any cells in the spatial index, and each subscription s may have different values of $BS_{min}(s, c, t_b)$ and $BS_{max}(s, c, t_b)$ for different c. If we use the grid index to store each subscription, then all the cells in the grid will be associated by the subscription. It is hard to determine an optimal granularity of grid index because different subscriptions may have their unique values of optimal granularity based on their locations, keywords, and preference parameters. Consequently, we use the quad-tree to index the subscriptions, which is a space partitioning based hierarchical index structure.

The reasons for using quad-tree to organize the spatial information of the ATSK subscription are summarized as follows. First, quad-tree has a space based partitioning scheme that allows ATSK subscription to be indexed in mutually-exclusive cells. In contrast, the bounding rectangles of the R-tree are dependent on the location distribution of messages, and they may overlap with each other. As a result, it is difficult to use the R-tree to index the conditional influence regions. Second, we can use diverse indexing granularity for different ATSK subscription by considering the spatial distribution, text

distribution, and different values of the subscription performance parameter α. This is important to the performance of both stream message assignment and subscription result update. Note that different from many applications that only keep the leaf level quad-tree cells, we need to keep all the cells including both leaf cells and internal cells.

The next problem is how to organize the subscriptions in each cell c. According to our problem definition, the messages that do not contain any subscription keyword will never be considered as subscription result. So we just need to evaluate the subscriptions containing at least one common keyword with the incoming message. In order to filter the subscriptions without common keyword with the message in advance, we organize the subscriptions in hierarchical inverted lists for each cell.

4.4 Multi-level Inverted File

Each node of the IQ*-tree corresponds to a multi-level inverted file for indexing the text information of the ATSK subscriptions associated at the node. Each inverted file consists of the hierarchical inverted lists for each subscription keyword.

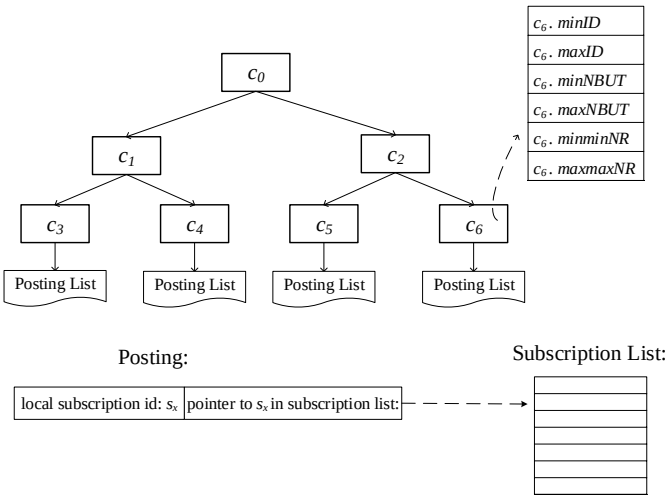


Fig. 6 Structure of the Hierarchical Inverted List

Figure 6 illustrates the structure of the hierarchical inverted lists and the postings. We can see that the hierarchical inverted list is basically a binary tree where each leaf node points to a posting list, which contains at most p_{max} postings. Each posting stores the subscription id and the pointers to corresponding subscriptions in the subscription list. Note that each subscription may have different subscription ids under different quad-tree cells. This is because the

subscription id is assigned based on the value of $\alpha \cdot R_{minACIR}(s, c, t_b)/(1+\alpha)$ and each subscription bears unique value under different cells.

In order to help prune the queries in the subtree rooted at the nodes of the inverted list that cannot make the incoming spatio-temporal message be the subscription results, each node c is augmented with the following values: (1) $minID$ and $maxID$, which respectively indicate the minimum and maximum ids of the subscriptions in the subtree rooted at node c; (2) $minNBUT$ and $maxNBUT$, which are the minimum and maximum values of $BUT(s, t_b)/(1 - s.\alpha)$ where $s \in S_c$ and S_c represents the subscriptions that belong to the subtree rooted at node n; (3) $minminNR$, which is the minimum value of $\alpha \cdot R_{minACIR}(s, c, t_b)/(1 + \alpha)$ for all $s \in S_c$; (4) $maxmaxNR$, which is the maximum value of $\alpha \cdot R_{maxACIR}(s, c, t_b)/(1 + \alpha)$ for all $q \in S_c$.

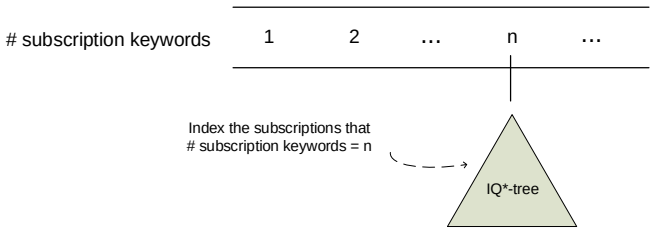


Fig. 7 Subscription Classification based on # subscription keywords

Since the text relevance between a subscription and an spatio-temporal message is related to both the number of subscription keywords and the number of message keywords, storing the subscriptions according to the number of subscription keywords in the first place can help estimate the bound of text relevance between a subscription and a message. Consequently, we make ATSK subscriptions firstly classified according to their numbers of subscription keywords. As Figure 7, we build separate IQ*-trees for the subscriptions with a particular number of subscription keywords.

5 EXPERIMENTAL STUDY

5.1 Baselines

We discuss how to exploit existing techniques for maintaining the real-time results for ATSK subscriptions over a stream of spatio-temporal messages. No structure and algorithm exist for solving the problem. Hence we develop some baselines utilizing existing structures for spatial and text retrieval.

A straightforward baseline is to use the inverted file to index the ATSK subscriptions. The posting of each subscription contains the corresponding subscription id. We also maintain a global subscription table. For each subscription s, the table stores the subscription id, $S_{bsk}(s, R_s[k], t_b)$, $s.\rho$, $s.\alpha$, and

the current subscription results of s. When a spatio-temporal message m arrives, we traverse the postings lists associated with the terms in the message in the Document-at-a-Time (DAAT) manner. For each posting, we get the subscription information from the subscription table based on its subscription id for computing the temporal spatial-keyword similarity score between m and the subscription ($S_{tsk}(m, s, t_{cur})$). Then we compare $S_{tsk}(m, s, t_{cur})$ with $S_{tsk}(s, R_s[k], t_{cur})$, which is computed from $S_{bsk}(s, R_s[k], t_b)$, and we determine whether m is a result of s. If so, we update the result set of s with m in the subscription table.

5.2 Experimental Settings

Our experiments are conducted on two real datasets: FSQ and TWE. FSQ is a real-life dataset collected from Foursquare, which contains 4 million worldwide check-ins with both location and text description. The dataset TWE is a larger real-life dataset that comprises 40 million tweets with geo-locations.

The ATSK subscriptions are generated as follows. For each check-in (resp. tweet with geo-location), we randomly select a number of keywords from the check-in (resp. tweet) keywords, and the subscription location is the same as the corresponding check-in (resp. tweet with geo-location) location. Note that both the check-ins description and the tweets posted by the user may reveal the interests of the user and their locations are likely to represent the active spots of the user, and thus the subscription generated in this way would be close to real subscriptions. In addition, each check-in description together with its location is considered to be a spatio-temporal message on FSQ, and each tweet with geo-location is regarded as a spatio-temporal message on TWE.

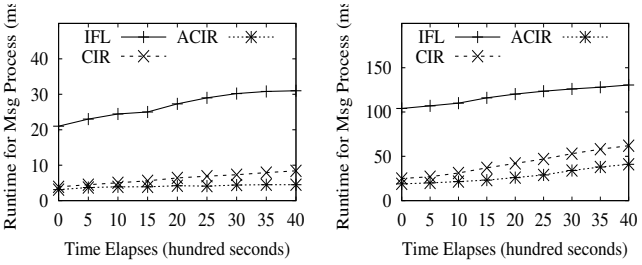
Default value for each parameter involved in the experiments are presented in Table 5.2.

Parameter	Setting	Default
number of subscription keywords	1 – 5	random from 1 to 5
preference parameter α	0.1 – 0.9	random from 0.1 to 0.9
number of maintained results	10 – 30	random from 10 to 30
number of postings in each block	N.A.	FSQ:128 TWE:1024
approximation ratio ϵ	0.05 – 0.2	0.1
number of indexed subscriptions	N.A.	FSQ:2M TWE 10M

Table 3 Experimental Parameter Settings

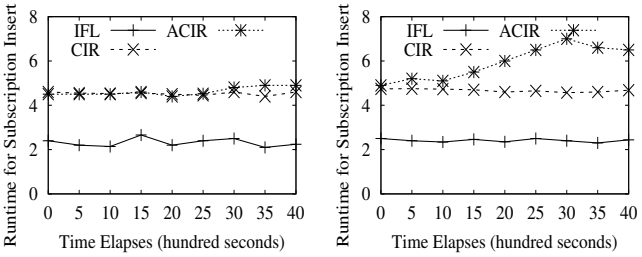
5.3 Experimental Result

A1 the Time Effect: In this set of experiments, each index runs for 4,000 seconds on both FSQ and TWE. At the beginning, each algorithm is initialized with 2,000,000 and 10,000,000 subscriptions respectively for FSQ and TWE.



(a) FSQ (b) TWE

Fig. 8 Effect of Time for Message Processing



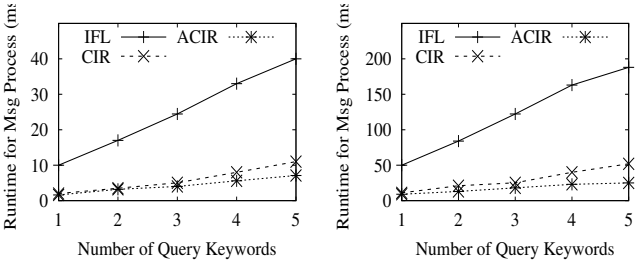
(a) FSQ (b) TWE

Fig. 9 Effect of Time for Subscription Insert

During each second 5 spatio-temporal messages are issued and 5 subscriptions are issued and expired respectively. We report the average runtime for processing a message and the average runtime for inserting a subscription during each period of 500 seconds. Note that the decaying scale is set to 0.5. The value of decaying scale is related to the exponential decay function, which has been presented in Equation 4. If we set the decaying scale to d_s, then $D^{-\Delta t_{sim}} = d_s$, where Δt_{sim} is the duration of the simulation. In our experimental setting, $\Delta t_{sim} = 4000$.

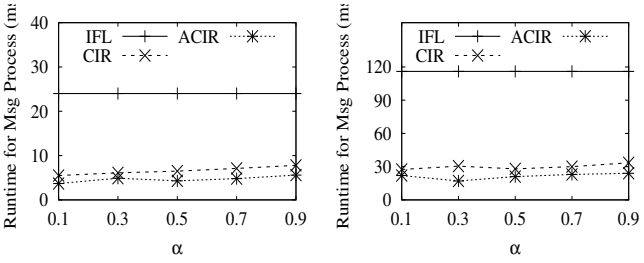
From Figure 8, we notice that both CIR and ACIR outperform the two baselines significantly in message processing, and ACIR exhibits the best performance. The reasons could be explained as follows.

For IFL, we need to check each postings in the postings lists the contain the keyword of the incoming spatio-temporal message. Its pruning technique is not able to consider the spatial proximity between the subscriptions and incoming messages. So the performance is the worst. In addition, ACIR performs moderately better than CIR. Such performance improvement is contributed by the grouping technique that takes advantage of the Approximate Conditional Influence Rings. Furthermore, we notice that the runtime of message processing for all structures exhibits an increasing trend as the time elapses. This is because that based on the exponential decaying function the values of $R_s[k]$ for the subscriptions whose results are not updated will decrease as the



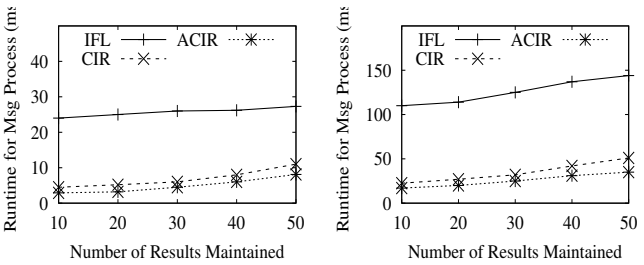
(a) FSQ (b) TWE

Fig. 10 Varying the Number of Subscription Keywords



(a) FSQ (b) TWE

Fig. 11 Effect of α



(a) FSQ (b) TWE

Fig. 12 Varying the Number of Results for each Subscription

time goes by, which may increase the number of subscriptions that can regard an incoming message as their results.

Figure 9 demonstrates the time cost of subscription insertion for each method. Because CIR and ACIR store subscriptions in the postings list maintained by each associating cell, the time costs of subscription insertion for CIR and ACIR are higher than IFL. However, according to the real-world publish/subscribe scenario, the frequency of subscription insertion is much lower than that of new messages. Consequently, the overall performance for CIR and ACIR still substantially outperforms the baseline.

A2 the Number of Subscription Keywords: This set of experiments investigates the effect of the number of keywords on each method. Figure 10 demonstrates that all of the methods exhibit an increasing trend for the runtime of message processing as we increase the number of subscription keywords. The reason is that increasing the number of subscription keywords results in the increasing of the number of postings for indexing the subscription, which will extend posting lists.

A3 Effect of α: This set of experiments evaluates the effect of the subscription preference parameter α. A greater value of α denotes the greater weight on the spatial proximity. While a smaller value of α indicates more emphasis on the text similarity. We observe the similar trends on both datasets. More weight put on the spatial aspect will lower the textual pruning effectiveness and vice-versa.

A4 Effect of k: This experiment evaluates the performance w.r.t. the number of results maintained by each subscription. From Figure 12 we can find that the runtime for message processing slightly increases as we increase k. It can be explained by the fact that higher value of k is likely to induce the lower temporal spatial-keyword similarity score of the k-th message.

6 Related work

Top-k Spatial Keyword Search. Top-k spatial keyword query returns k most spatial-textual relevant geo-textual objects that are ranked by both spatial proximity and text relevancy between them and the query. A number of geo-textual indices have been developed to efficiently answer top-k spatial keyword queries [11, 18, 33, 62, 63]. There exits no sensible way to adopt these indexes to handle a stream of ATSK subscriptions. Moreover, Wu et al. [55] consider the problem of continuously answering moving top-k spatial keyword queries over a set of static geo-textual objects. The solution is to compute the safe zone based on a variant of Voronoi cells. The problem and solution are very different from ours.

Content based top-k publish/subscribe Existing work on top-k publish/subscribe systems [2, 6, 14, 15, 32, 50] make published items trigger a subscription if it ranks among the top-k published items. The top-k publish/subscribe system [50] is similar to our setting, where the published items are tweets and the subscriptions are news. The published items do not have a fixed expiration time. Instead, time is a part of the relevance score, which decays as time passes. Older items retire from the top-k only when new items that score higher arrive and take their places. The inverted files are used as the indexes and the classic information retrieval methods are adapted for the filtering.

Some proposals on publish/subscribe [3,19] in the literature consider spatio-temporal documents. The subscription contains a spatial circle and a set of keywords, and the published items are spatio-temporal documents. Published items trigger a subscription when they match both the keywords and spatial

region of the subscription. In the setting, each subscription has a region and a set of keywords, and the problem is to maintain an index on the subscriptions such that the subscriptions matching an incoming spatio-temporal document can be efficiently retrieved. To do this, Chen et al. [3] present a hybrid index based on Quad-tree and Inverted files, and Li et al. [19] present a hybrid index based on R-tree and Inverted files. The problem is much easier than that for the top-k publish/subscribe, and the indexing methods and subscription matching algorithms cannot be employed to handle our subscriptions.

A different type of ranked publish/subscribe systems is introduced [31], which produces a ranked list of subscriptions for a published item, while we produce a ranked set of published items for each subscription. The subscriptions [31] are defined as interval ranges, and published items are points that match the intervals. The setting is very different from that in our work.

In the future, it is of interest to integrate text data with multiple spatial queries. First, it is of interest to integrate spatial trajectory data [24, 26–30, 34, 37–39, 64, 65] with text data and to conduct novel spatio-textual trajectory search, recommendation, and analysis studies. Second, it is of interest to use POI data and geo-tagged social media data to discover significant locations/regions [13,41,45–48,52,60]. Third, it is of interest to study integrate text data to routing problem [35,36,40,42–44,56,59,66] and to study spatio-textual routing problem. Fourth, it is of interest to consider streaming data sampling and analysis, and other query types [7–9, 16, 17, 21, 23, 25, 53, 54, 57, 58, 61].

7 Conclusions

We consider the problem of maintaining up-to-date most relevant spatio-temporal messages for a large number of location-based top-k subscriptions with a guaranteed approximation ratio of temporal spatial-keyword similarity sores between subscription and new messages. We define a brand new type of location-based top-k subscription, the ATSK subscription, that can automatically adjust the score of triggering condition by taking the scores of other subscriptions into account. As a result, the group filtering efficacy can be substantially improved at the cost of sacrificing the publishing result quality in a controlled range. We propose an efficient approach to processing a large number of ATSK subscriptions. The experimental results on two real-world datasets show that our solution is able to achieve a reduction of the processing time by 20%–70% compared with two baselines.

References

1. G. Amati, G. Amodeo, and C. Gaibisso. Survival analysis for freshness in microblogging search. In *CIKM*, pages 2483–2486. ACM, 2012.
2. L. Chen and G. Cong. Diversity-aware top-k publish/subscribe for text stream. In *SIGMOD*, pages 347–362, 2015.
3. L. Chen, G. Cong, and X. Cao. An efficient query indexing mechanism for filtering geo-textual data. In *SIGMOD*, pages 749–760, 2013.

4. L. Chen, G. Cong, X. Cao, and K. Tan. Temporal spatial-keyword top-k publish/subscribe. In *ICDE*, pages 255–266, 2015.
5. L. Chen, G. Cong, C. S. Jensen, and D. Wu. Spatial keyword query processing: an experimental evaluation. In *PVLDB*, pages 217–228, 2013.
6. L. Chen, Y. Cui, G. Cong, and X. Cao. SOPS: A system for efficient processing of spatial-keyword publish/subscribe. *PVLDB*, 7(13):1601–1604, 2014.
7. Z. Chen and M. J. Cafarella. Integrating spreadsheet data via accurate and low-effort extraction. In *KDD*, pages 1126–1135, 2014.
8. Z. Chen, M. J. Cafarella, and H. V. Jagadish. Long-tail vocabulary dictionary extraction from the web. In *WSDM*, pages 625–634, 2016.
9. Z. Chen, S. Dadiomov, R. Wesley, G. Xiao, D. Cory, M. J. Cafarella, and J. Mackinlay. Spreadsheet property detection with rule-assisted active learning. In *CIKM*, pages 999–1008, 2017.
10. M. Christoforaki, J. He, C. Dimopoulos, A. Markowetz, and T. Suel. Text vs. space: efficient geo-search query processing. In *CIKM*, pages 423–432, 2011.
11. G. Cong, C. S.Jensen, and D. Wu. Efficient retrieval of the top-k most relevant spatial web objects. In *PVLDB*, pages 337–348, 2009.
12. M. Efron and G. Golovchinsky. Estimation methods for ranking recent information. In *SIGIR*, pages 495–504. ACM, 2011.
13. D. Guo, Y. Zhu, W. Xu, S. Shang, and Z. Ding. How to find appropriate automobile exhibition halls: Towards a personalized recommendation service for auto show. *Neurocomputing*, 213:95–101, 2016.
14. P. Haghani, S. Michel, and K. Aberer. Evaluating top-k queries over incomplete data streams. In *CIKM*, pages 877–886, 2009.
15. P. Haghani, S. Michel, and K. Aberer. The gist of everything new: Personalized top-k processing over web 2.0 streams. In *CIKM*, pages 489–498, 2010.
16. J. Han, K. Zheng, A. Sun, S. Shang, and J. Wen. Discovering neighborhood pattern queries by sample answers in knowledge base. In *ICDE*, pages 1014–1025, 2016.
17. S. Hu, J. Wen, Z. Dou, and S. Shang. Following the dynamic block on the web. *World Wide Web*, 19(6):1077–1101, 2016.
18. I.D.Felipe, V.Hristidis, and N.Rishe. Keyword search on spatial databases. In *ICDE*, pages 656–665, 2008.
19. G. Li, Y. Wang, T. Wang, and J. Feng. Location-aware publish/subscribe. In *KDD*, pages 802–810, 2013.
20. X. Li and W. B. Croft. Time-based language models. In *CIKM*, pages 469–475. ACM, 2003.
21. Z. Li, S. Shang, Q. Xie, and X. Zhang. Cost reduction for web-based data imputation. In *DASFAA*, pages 438–452, 2014.
22. H. Liang, Y. Xu, D. Tjondronegoro, and P. Christen. Time-aware topic recommendation based on micro-blogs. In *CIKM*, pages 1657–1661, 2012.
23. A. Liu, X. Shen, Z. Li, J. Xu, L. Zhao, K. Zheng, and S. Shang. Differential private collaborative web services qos prediction. *World Wide Web*, online first:1–25, 2018.
24. A. Liu, W. Wang, S. Shang, Q. Li, and X. Zhang. Efficient task assignment in spatial crowdsourcing with worker and task privacy protection. *GeoInformatica*, online first:1–28, 2017.
25. J. Liu, S. Shang, K. Zheng, and J. Wen. Multi-view ensemble learning for dementia diagnosis from neuroimaging: An artificial neural network approach. *Neurocomputing*, 195:112–116, 2016.
26. J. Liu, K. Zhao, P. Sommer, S. Shang, B. Kusy, and R. Jurdak. Bounded quadrant system: Error-bounded trajectory compression on the go. In *ICDE*, pages 987–998, 2015.
27. J. Liu, K. Zhao, P. Sommer, S. Shang, B. Kusy, J. Lee, and R. Jurdak. A novel framework for online amnesic trajectory compression in resource-constrained environments. *IEEE Trans. Knowl. Data Eng.*, 28(11):2827–2841, 2016.
28. K. Liu, Y. Li, J. Dai, S. Shang, and K. Zheng. Compressing large scale urban trajectory data. In *CloudDP@EuroSys*, pages 3:1–3:6, 2014.
29. K. Liu, Y. Li, Z. Ding, S. Shang, and K. Zheng. Benchmarking big data for trip recommendation. In *ICCCN*, pages 1–6, 2014.

30. K. Liu, B. Yang, S. Shang, Y. Li, and Z. Ding. MOIR/UOTS: trip recommendation with user oriented trajectory search. In *MDM*, pages 335–337, 2013.
31. A. Machanavajjhala, E. Vee, M. Garofalakis, and J. Shanmugasundaram. Scalable ranked publish/subscribe. *PVLDB*, 1(1):451–462, Aug. 2008.
32. K. Pripužić, I. P. Žarko, and K. Aberer. Top-k/w publish/subscribe: Finding k most relevant publications in sliding time window w. In *DEBS*, pages 127–138, 2008.
33. J. B. Rocha-Junior, O. Gkorgkas, S. Jonassen, and K. Nørvåg. Efficient processing of top-k spatial keyword queries. In *SSTD*, pages 205–222, 2011.
34. S. Shang, L. Chen, C. S. Jensen, J. Wen, and P. Kalnis. Searching trajectories by regions of interest. *IEEE Trans. Knowl. Data Eng.*, 29(7):1549–1562, 2017.
35. S. Shang, L. Chen, Z. Wei, D. Guo, and J. Wen. Dynamic shortest path monitoring in spatial networks. *J. Comput. Sci. Technol.*, 31(4):637–648, 2016.
36. S. Shang, L. Chen, Z. Wei, C. S. Jensen, J. Wen, and P. Kalnis. Collective travel planning in spatial networks. *IEEE Trans. Knowl. Data Eng.*, 28(5):1132–1146, 2016.
37. S. Shang, L. Chen, Z. Wei, C. S. Jensen, K. Zheng, and P. Kalnis. Trajectory similarity join in spatial networks. *PVLDB*, 10(11):1178–1189, 2017.
38. S. Shang, L. Chen, Z. Wei, C. S. Jensen, K. Zheng, and P. Kalnis. Parallel trajectory similarity joins in spatial networks. *VLDB J.*, online first:1–25, 2018.
39. S. Shang, R. Ding, K. Zheng, C. S. Jensen, P. Kalnis, and X. Zhou. Personalized trajectory matching in spatial networks. *VLDB J.*, 23(3):449–468, 2014.
40. S. Shang, D. Guo, J. Liu, and J. Wen. Prediction-based unobstructed route planning. *Neurocomputing*, 213:147–154, 2016.
41. S. Shang, D. Guo, J. Liu, K. Zheng, and J. Wen. Finding regions of interest using location based social media. *Neurocomputing*, 173:118–123, 2016.
42. S. Shang, J. Liu, K. Zheng, H. Lu, T. B. Pedersen, and J. Wen. Planning unobstructed paths in traffic-aware spatial networks. *GeoInformatica*, 19(4):723–746, 2015.
43. S. Shang, H. Lu, T. B. Pedersen, and X. Xie. Finding traffic-aware fastest paths in spatial networks. In *SSTD*, pages 128–145, 2013.
44. S. Shang, H. Lu, T. B. Pedersen, and X. Xie. Modeling of traffic-aware travel time in spatial networks. In *MDM*, pages 247–250, 2013.
45. S. Shang, B. Yuan, K. Deng, K. Xie, K. Zheng, and X. Zhou. PNN query processing on compressed trajectories. *GeoInformatica*, 16(3):467–496, 2012.
46. S. Shang, B. Yuan, K. Deng, K. Xie, and X. Zhou. Finding the most accessible locations: reverse path nearest neighbor query in road networks. In *ACM SIGSPATIAL*, pages 181–190, 2011.
47. S. Shang, K. Zheng, C. S. Jensen, B. Yang, P. Kalnis, G. Li, and J. Wen. Discovery of path nearby clusters in spatial networks. *IEEE Trans. Knowl. Data Eng.*, 27(6):1505–1518, 2015.
48. S. Shang, S. Zhu, D. Guo, and M. Lu. Discovery of probabilistic nearest neighbors in traffic-aware spatial networks. *World Wide Web*, 20(5):1135–1151, 2017.
49. A. Shraer, M. Gurevich, M. Fontoura, and V. Josifovski. Top-k publish-subscribe for social annotation of news. *PVLDB*, 6(6):385–396, 2013.
50. A. Shraer, M. Gurevich, M. Fontoura, and V. Josifovski. Top-k publish-subscribe for social annotation of news. *PVLDB*, 6(6):385–396, 2013.
51. X. Wang, Y. Zhang, W. Zhang, X. Lin, and W. Wang. Ap-tree: Efficiently support continuous spatial-keyword queries over stream. In *ICDE*, pages 1107–1118, 2015.
52. Y. Wang, J. Li, Y. Zhong, S. Zhu, D. Guo, and S. Shang. Discovery of accessible locations using region-based geo-social data. *WWW Journal*, online first:1–18, 2018.
53. Z. Wei, X. He, X. Xiao, S. Wang, S. Shang, and J. Wen. Topppr: Top-k personalized pagerank queries with precision guarantees on large graphs. In *SIGMOD*, pages 1–16, 2018.
54. Z. Wei, X. Liu, F. Li, S. Shang, X. Du, and J. Wen. Matrix sketching over sliding windows. In *SIGMOD*, pages 1465–1480, 2016.
55. D. Wu, M. L. Yiu, C. S. Jensen, and G. Cong. Efficient continuously moving top-k spatial keyword query processing. In *ICDE*, pages 541–552, 2011.
56. K. Xie, K. Deng, S. Shang, X. Zhou, and K. Zheng. Finding alternative shortest paths in spatial networks. *ACM Trans. Database Syst.*, 37(4):29:1–29:31, 2012.
57. Q. Xie, S. Shang, B. Yuan, C. Pang, and X. Zhang. Local correlation detection with linearity enhancement in streaming data. In *CIKM*, pages 309–318, 2013.

58. X. Xie, H. Lu, J. Chen, and S. Shang. Top-k neighborhood dominating query. In *DASFAA*, pages 131–145, 2013.
59. B. Yang, C. Guo, C. S. Jensen, M. Kaul, and S. Shang. Stochastic skyline route planning under time-varying uncertainty. In *ICDE*, pages 136–147, 2014.
60. B. Yao, Z. Chen, X. Gao, S. Shang, S. Ma, and M. Guo. Flexible aggregate nearest neighbor queries in road networks. In *ICDE*, pages 1–12, 2018.
61. B. Yao, W. Zheng, Z. Wang, Z. Chen, S. Shang, K. Zheng, and M. Guo. Distributed in-memory analytics for big temporal data. In *DASFAA*, pages 1–16, 2018.
62. C. Zhang, Y. Zhang, W. Zhang, and X. Lin. Inverted linear quadtree: Efficient top k spatial keyword search. In *ICDE*, pages 901–912, 2013.
63. D. Zhang, K.-L. Tan, and A. K. H. Tung. Scalable top-k spatial keyword search. In *EDBT*, pages 359–370, 2013.
64. B. Zheng, H. Wang, K. Zheng, H. Su, K. Liu, and S. Shang. Sharkdb: an in-memory column-oriented storage for trajectory analysis. *World Wide Web*, 21(2):455–485, 2018.
65. K. Zheng, Y. Zheng, N. J. Yuan, and S. Shang. On discovery of gathering patterns from trajectories. In *ICDE*, pages 242–253, 2013.
66. S. Zhu, Y. Wang, S. Shang, G. Zhao, and J. Wang. Probabilistic routing using multi-modal data. *Neurocomputing*, 253:49–55, 2017.