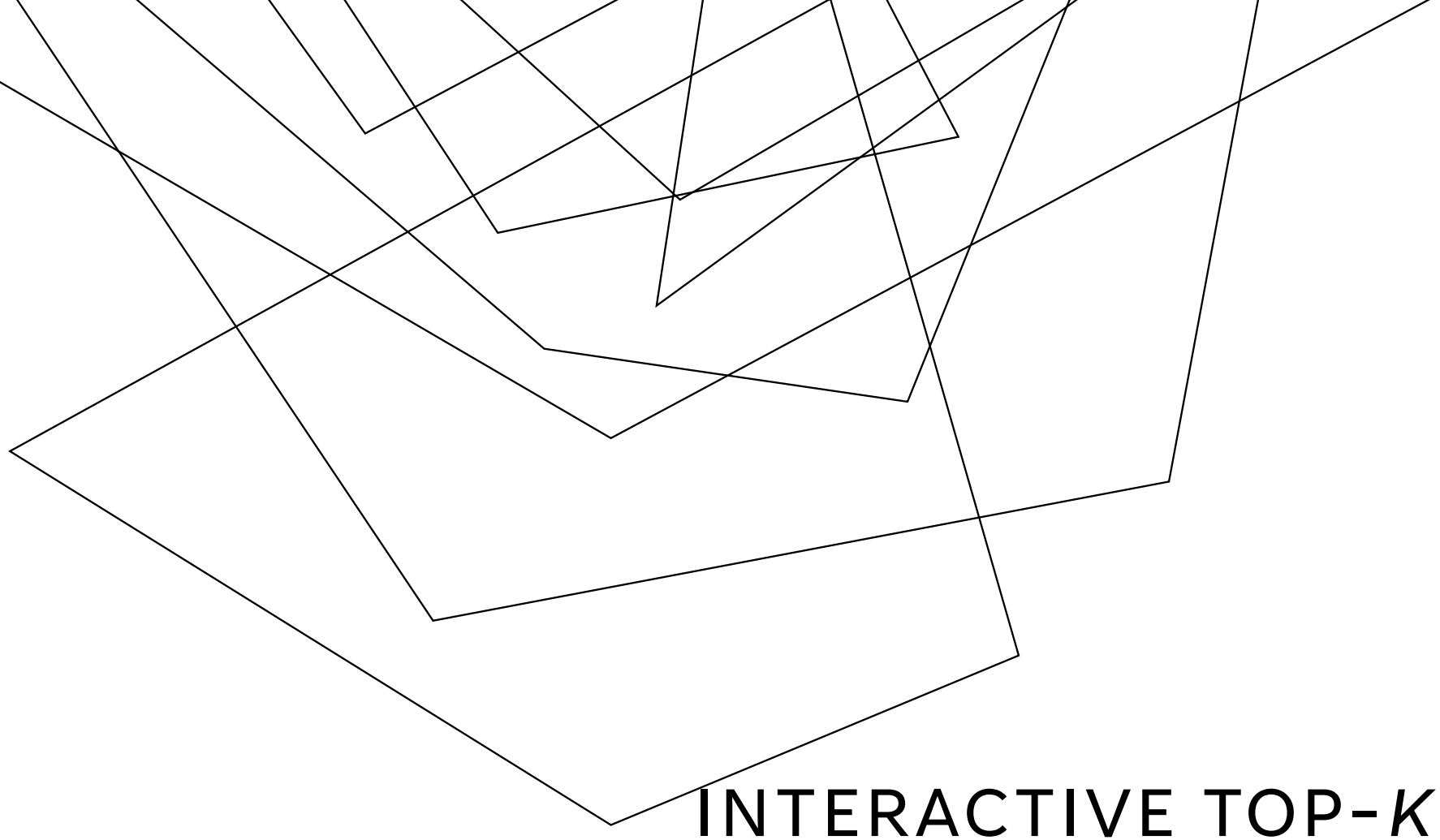


INTERACTIVE TOP-*K* SPATIAL KEYWORD QUERIES

Yu-Pin Liang

Background

- With the **rapid** transformation of web clients from desktop computers to mobile devices, increasing volumes of geo-textual objects are becoming available on the web that represent Points of Interest (**PoIs**).

- Specifically, a geotextual object contains the geo-location of its PoI and textual descriptions of the PoI (e.g., features, facilities, reviews).

Background

- **Boolean Range Queries:** retrieve all objects whose text description contains all the query keywords and whose location is within the query region.

- **Boolean kNN Queries:** retrieve the k objects nearest to the query location and each object's text description contains all the query keywords.

- **Top-*k* Range Queries:** retrieve up to k objects whose location is within the query region and has the highest textual relevance to the query keywords.

- **Top-k kNN Queries:** retrieve the k objects with the highest ranking scores, measured by a weighted combination of their distances to the query location and the textual similarity between their textual descriptions and query keywords.

TABLE I: CHARACTERISTICS OF DIFFERENT TYPES OF SPATIAL KEYWORD QUERIES

Query type	**Matching all query keywords required**	**Controlled result size**	**Results are ranked**	**Preferences on spatial and textual dimensions are considered**	**Preferences on different query keywords are considered**
Boolean Range Query	yes	no	no	no	no
Boolean kNN Query	yes	yes	yes	no	no
Top-k Range Query	no	yes	yes	no	no
Top-k kNN Query	no	yes	yes	yes, but users need to specify their preferences explicitly	no
Interactive Top-k Spatial Keyword Query	no	yes	yes	yes and the preferences are learnt from user feedback	yes and the preferences are learnt from user feedback

Background – limitation of different type of queries

- Result set of Boolean Range Queries has uncontrolled size and is unranked, leading to too many/few results. Boolean kNN Queries address problem by applying a rank over the results according to their distances to the query location and returning the k closest objects only.

- However, both types of queries **require each result to contain all the query keywords**, which may find too few results and/or the results are far away from the query location.

Background – Top-*k* queries

- Top-k Range Queries and Top-k kNN Queries relax this requirement by introducing textual relevance function as the **similarity measure** between query keywords and text description of PoIs.

- Top-k Range Queries rank the result set by textual similarity to the query only, while Top-k kNN Queries combine the similarities over **spatial** and **textual** dimensions together into a unified utility function and return the top-k results based on this utility function.

- Top-k kNN Queries are the most novel and advanced type of spatial keyword queries in literature, which are often referred to as **Top-k Spatial Keyword Queries (TkSK)** when the context is clear.

Background – limitation of T*k*SK (1)

It is impractical to ask users to specify their preferences

- TkSK queries combine spatial similarity and textual similarity into one utility function in the form of $\beta S_{spatial}+(1-\beta)S_{text}$, in which $S_{spatial}$, S_{text} are spatial and textual similarity between query and object respectively and $\beta \in [0, 1]$ is a weighting parameter indicating user's preference over spatial and textual dimensions.

- A high β favours the objects that are geographically closer to the query location while small β tends to return the objects whose text description more relevant to query keywords.

- Nevertheless value of β needs to be specified by the user a priori, which can be quite impractical in real applications. In fact, user preferences are often latent and thus hard to be quantified exactly and explicitly.

Background – limitation of T*k*SK (2)

The results of TkSK queries with respect to the textual similarity may not be as intuitive as the boolean keyword queries.

- It calculates the weight for each common keyword of query's and object's text using TF-IDF measure and computes the normalized dot product between vectors of query keywords and object keywords.

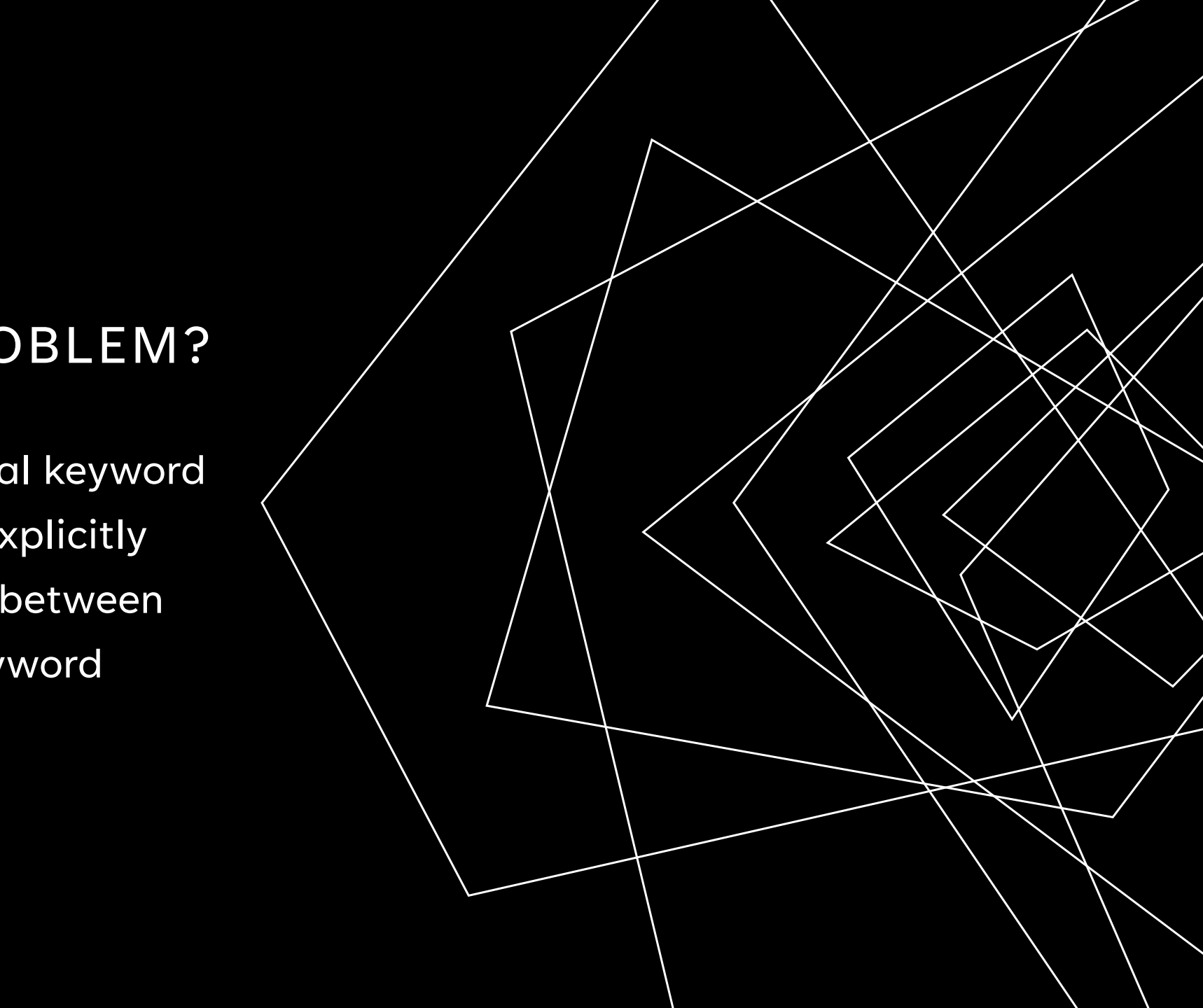
- results derived from this model may not be desired by the users, since a user would like to assign a larger weight to some keyword simply because she feels it is more important

Example of spatial keyword query

1. A user looking for a Cafe nearby, which must serve fish&chips (more important) and ideally plays music (less important).
2. She issues a spatial keyword query q with her current location and two keywords fish&ships, music.
3. o1 to o6 are restaurants/Cafes nearby q with the keywords and number indicates the normalized distance to q.
4. user may have no idea to specify weight β in the TkSK query, and accepts default value $\beta = 0.5$.
5. Since the weight for each keyword is assigned based on TF-IDF model in TkSK query, music has much higher weight (= 0.5) than fish&chips (=0.25).
6. o6 turns out to be the best object.
7. However obviously not a satisfactory result for the user: o1 is equally close to q with o6, but contains a more important keyword fish&chips, so o6 is at least worse than o1.

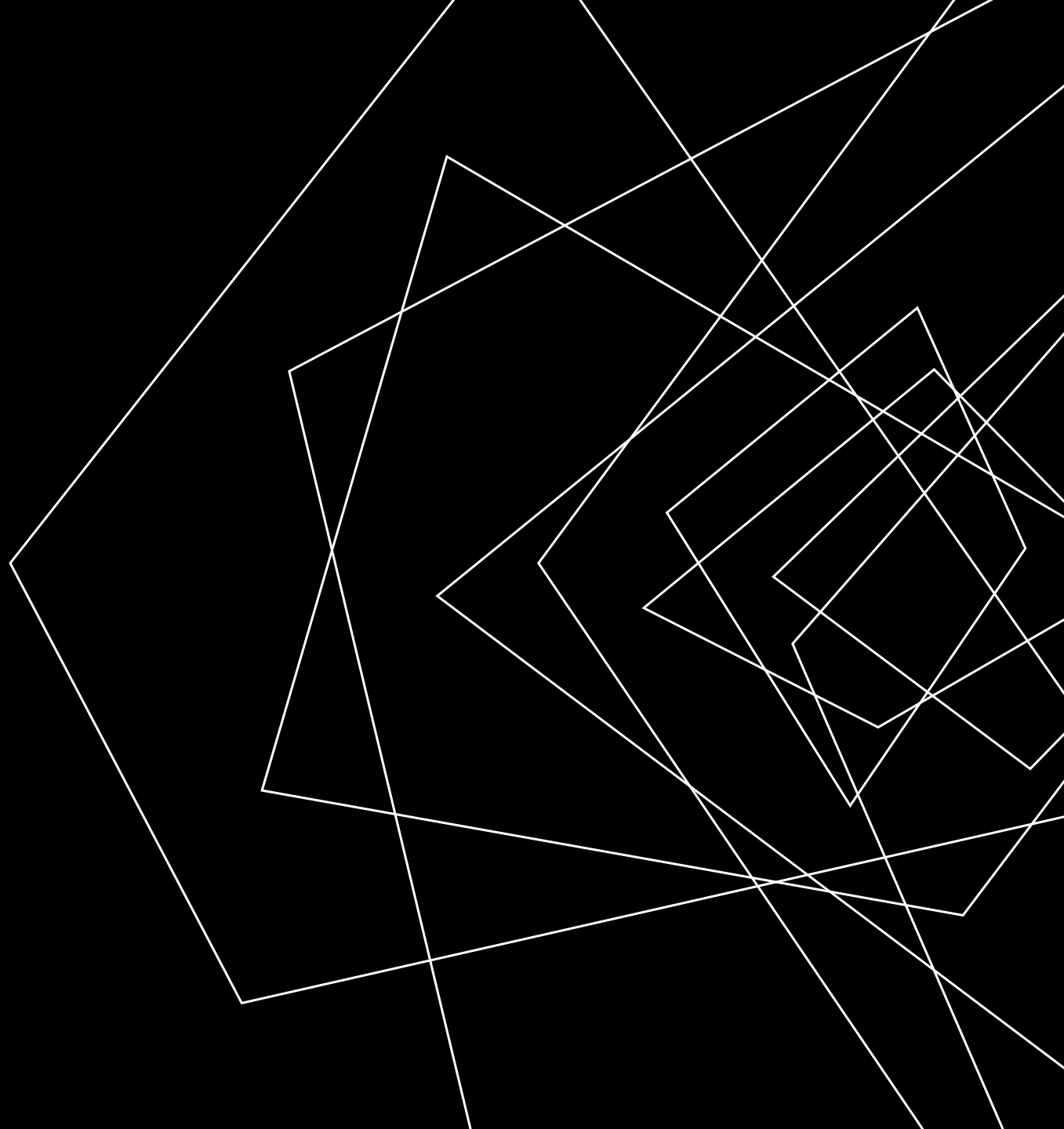
WHAT IS THE PROBLEM?

Conventional top-k spatial keyword queries require users to explicitly specify their preferences between spatial proximity and keyword relevance.



WHY IS IT IMPORTANT?

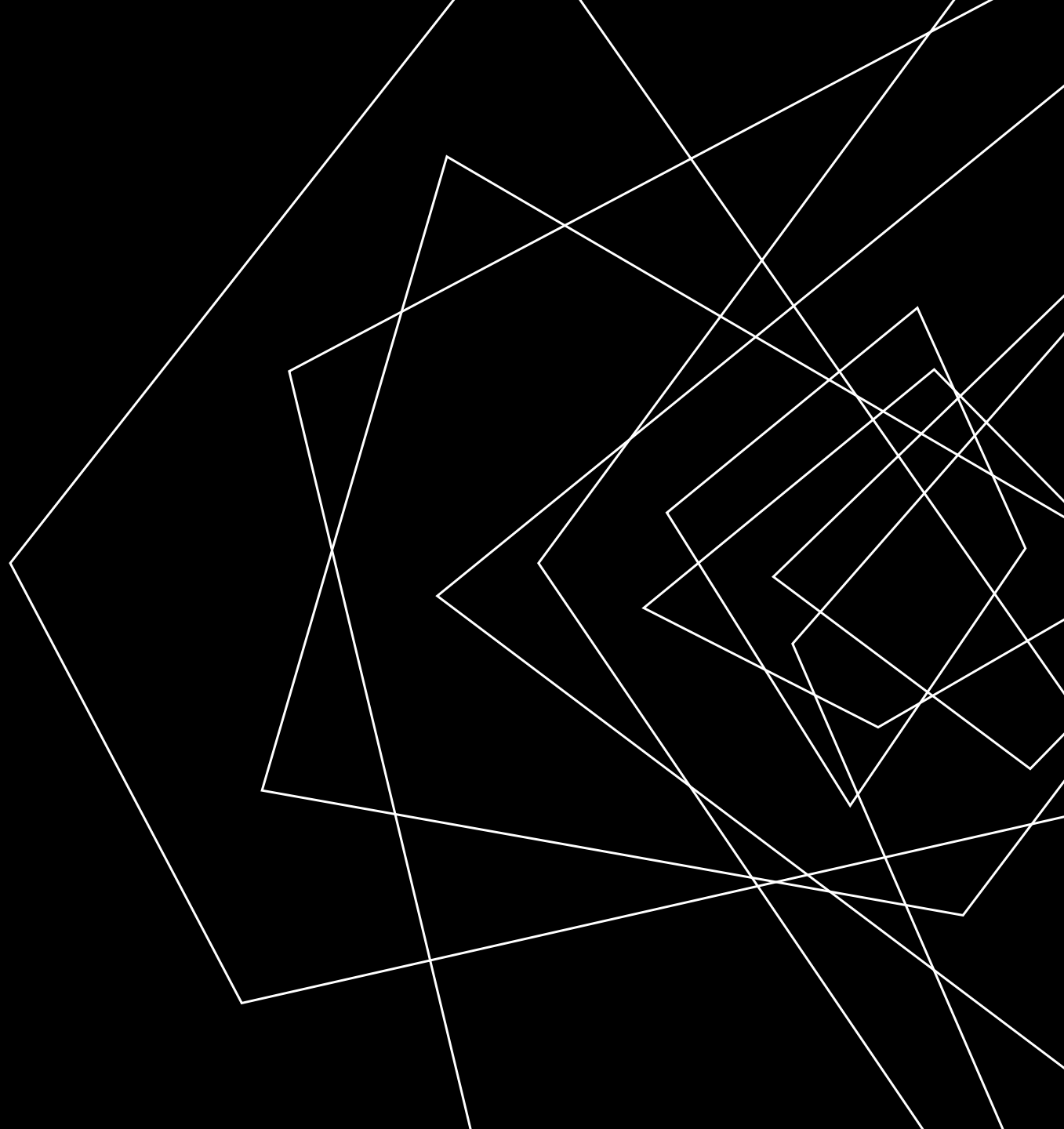
With the rapid transformation of web clients from desktop computers to mobile devices increasing volumes of geo-textual objects are becoming available on the web that represent Points of Interest (PoIs).



WHAT ARE THE EXISTING TECHNIQUES AND THEIR LIMITATIONS?

1. It is impractical to ask users to specify their preferences

2. The results of TkSK queries with respect to the textual similarity may not be as intuitive as the boolean keyword queries.



WHAT IS THE CURRENT/NAÏVE SOLUTION?

- Ask client specify their preferences prior to searching

- Use traditional method and return a lots of results sets

WHAT MAKES IT POSSIBLE FOR THE AUTHORS TO ADDRESS THESE LIMITATIONS?

Enhancing spatial keyword queries with user interactions.

HOW IS PROBLEM ACTUALLY SOLVED IN THIS PAPER?

Efficiently search for a subset of tuples in the database at each round that can effectively learn the user's preference based on her choice.

How is problem actually solved in this paper? -1

We identify limitations of existing spatial keyword queries. Based on this, we define **a novel Interactive Top-k Spatial Keyword (ITkSK) query**, which not just allows users to control the preference weights on distance and individual keyword in an intuition consistent way, but also eliminates the hassles to specify all these parameters explicitly by involving interactions with the users.

How is problem actually solved in this paper? -2

We propose **a three-phase solution** to process ITkSK query.

- The **first** phase (candidate generation phase) quickly narrows the search space from the entire database down to a set of candidates by retrieving Geo-textual k-skyband set from the database with respect to the query.
- In the **second** phase (interaction phase) we develop several strategies to select a subset of candidates and present them to users at each round, with the aim of maximizing the benefit of learning from the user's feedback.
- In the **last** phase (finalization phase) we discuss how to terminate the interaction automatically and estimate the final preference vector based on a set of linear constraints.

How is problem actually solved in this paper? -3

We conduct empirical study based on real PoI datasets. The favourable results verifies our expectation that ITkSK queries indeed return more satisfactory results by learning a more accurate user preference. Moreover, our interaction strategies are shown to be quite effective in terms of convergence speed.

PROBLEM STATEMENT - A. Preliminaries

Definition 1 (Geo-textual object). *Let D be a geo-textual dataset. Each geo-textual object $o \in D$ is defined as a pair $(o.\rho, o.\psi)$, where $o.\rho$ is a 2-dimensional geographical location with longitude and latitude and $o.\psi$ is a text document represented by a set of keywords or terms.*

Definition 2 (Utility function). *Let $(q.\rho, q.\psi = \{t_1, t_2, ..., t_m\})$ be a spatial keyword query specified by a user, $\mathbf{w} = \{w_0, w_1, w_2, ..., w_m\}$ be a vector representing user preference, in which $\forall i \in [0, m], 0 \leq w_i \leq 1$. For each geo-textual object $o \in D$, the user's utility obtained from o can be evaluated by the following utility function,*

$$u_{q,w}(o) = w_0(1 - d(q.\rho, o.\rho)) + \sum_{i=1}^{m} w_i h_o(q.t_i) \qquad (1)$$

where $d(q.\rho, o.\rho)$ is a function that normalizes the Euclidean distance between o and q into the range $[0, 1]$ and h is a function indicating the existence of t_i in o, i.e.,

$$h_o(q.t_i) = \begin{cases} 1, \text{if } q.t_i \in o.\psi \\ 0, \text{otherwise} \end{cases}$$

When context about q and $\mathbf{w}$ is clear, we simply use $u(o)$ to represent $u_{q,w}(o)$.

PROBLEM STATEMENT - A. Preliminaries

Definition 3 (Top-k Spatial Keyword Query)**.** *Given a geo-textual dataset D, a query $q : (q.\rho, q.\psi)$, a preference vector* $\mathbf{w}$ *and the number of results k, a top-k spatial keyword (TkSK) query returns a set S of up to k objects from D such that they have the highest utilities w.r.t. $u(o)$, i.e.,*

$$S = \{S \subseteq D, |S| = k | \forall o \in S, o' \in D \setminus S, u(o) \geq u(o')\}$$

Definition 4 (Interactive Top-k Spatial Keyword Query)**.** *Given a geo-textual dataset D, a query $q : (q.\rho, q.\psi)$, an integer k and an unknown preference vector* $\mathbf{w}$*, the interactive top-k spatial keyword (ITkSK) query will be processed in rounds. In each round, the system displays at most κ tuples to the user and asks her to pick the favourite one according to* $\mathbf{w}$*. After interaction, the system will estimate the user's preference as $\mathbf{w}'$ based on her feedbacks and return a final set of k tuples based on $\mathbf{w}'$.*

PROBLEM STATEMENT - B. Solution Overview

The proposed solution consists of three phases:

1) **Candidate** generation: This phase will find a set of geo-textual k-skyband tuples from the entire database as the initial candidates.

2) **Interaction**: This phase proceeds in the fashion of rounds. At each round, the system strategically selects a subset of candidates and presents them to the user, who will pick her favourite tuple according to her latent preference. The system will refine the candidate set based on her feedback and continue the process by selecting another subset of candidates. Meanwhile, a termination condition is checked automatically and once satisfied the system exits this phase.

3) **Finalisation**: This phase estimates the user's preference w' based on her feedbacks during the interaction

CANDIDATE GENERATION- A. Geo-textual dominance

- Since the user preferences on spatial distance and keywords are **unknown** at this stage, the desired candidate set should include all the objects that could possibly become a final result given a certain preference vector.

- Naturally we can adopt the notion of skyline, which is a set of tuples in a database that are not **dominated** by any other tuple. Here a tuple a is said to dominate tuple b if a has better or equal values in all attributes and a better value in at least one attribute than b. In the sequel, we first define the dominance relationship between two geo-textual objects.

CANDIDATE GENERATION- B. Search Algorithm

- Following the branch-and-bound paradigm, next we propose the GSB (Geo-textual SkyBand) algorithm to find the candidate set efficiently.

- **IR2-Tree Index**

- **GSB Algorithm:**
  - If e is a non-leaf node, GSB performs a dominance check for each of its child entries to see if it is dominated by more than k – 1 skyband tuples found so far.
  - If e is a leaf node, then it contains geo-textual objects only.

$$MINGTD_q(e) = MINDIST(q.\rho, M_e) + \sum_{t \in q.\psi} (s_t \wedge s_e) \oplus s_t$$

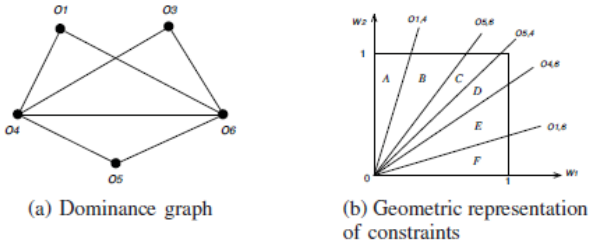
INTERACTION PROCESS

- In this phase, system will interact with end users in rounds. More specifically, the system at each round will choose a subset of objects from the candidate set generated in the previous phase and then present them to the users who will browse and pick one favourite object from them.

- The system keeps refining the user's preference that has been learnt based on the user's selections in current and all previous rounds.

- The interaction will continue until the user stops it explicitly or the system automatically decides to exit when it believes no more benefit of doing so.

- approximation of user preference by involving user interaction.



(a) Dominance graph

(b) Geometric representation of constraints

Fig. 2: Selection strategies

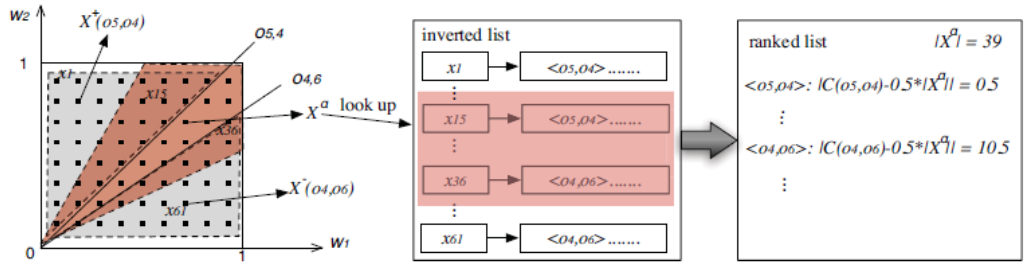


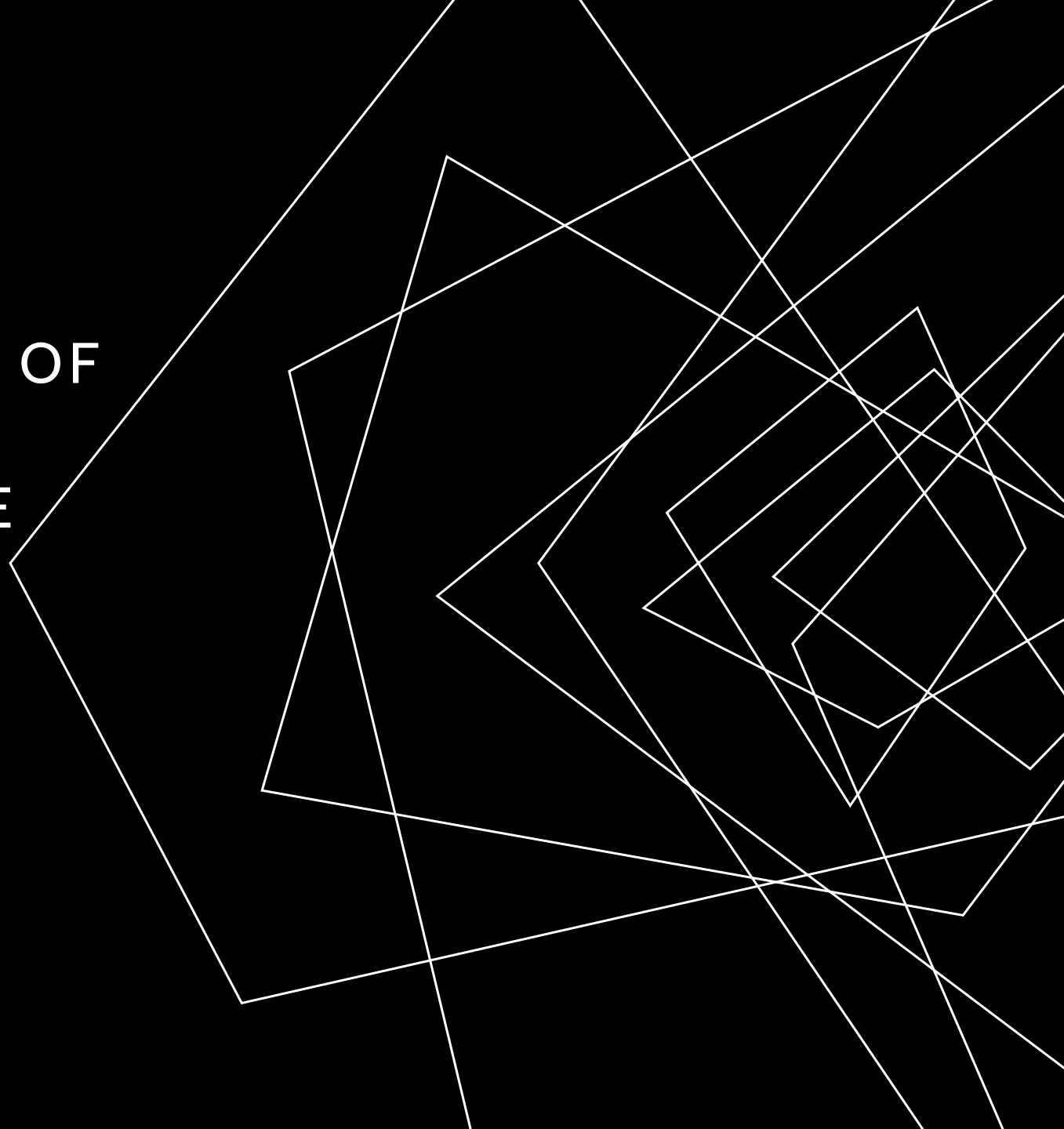
Fig. 3: Efficient approximation of UR strategy

FINALISATION

A. Termination condition

B. Estimation of w

HOW IS THE PERFORMANCE OF THE PROPOSED TECHNIQUE COMPARED TO THAT OF THE EXISTING ONES?



EXPERIMENTAL EVALUATION

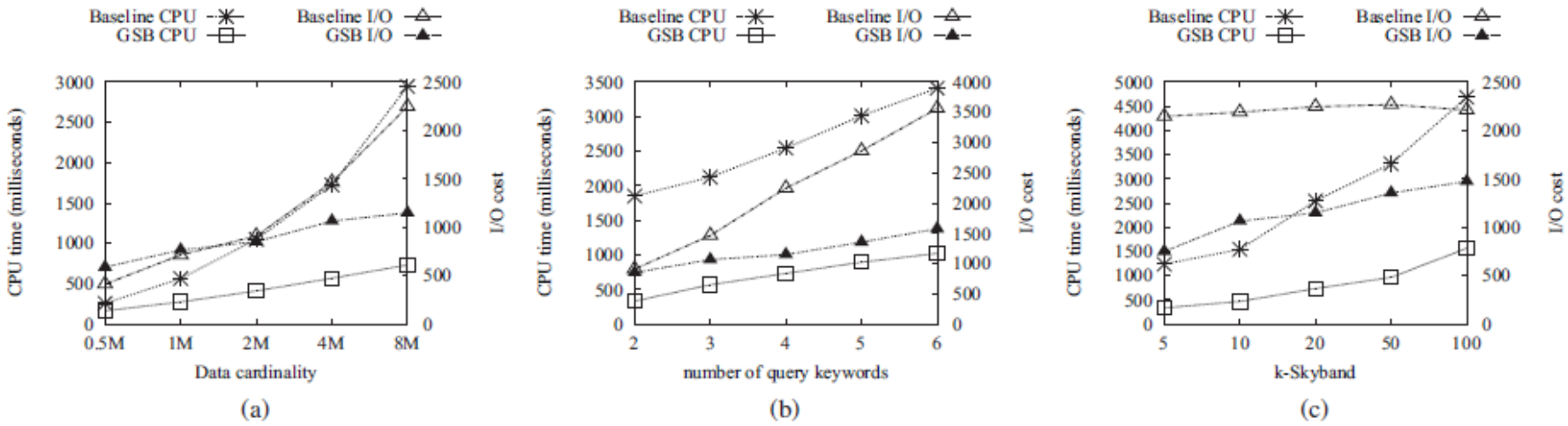


Fig. 4: Performance evaluation of GSB algorithm (CH)

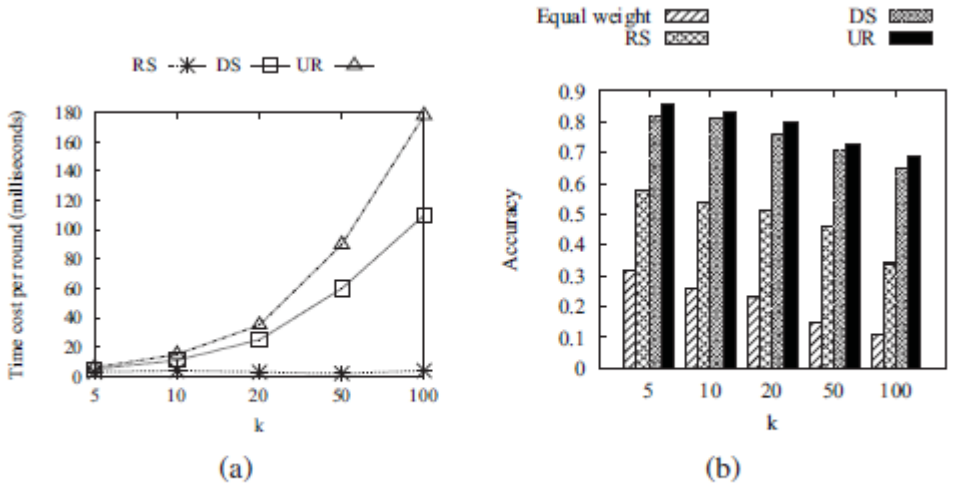


Fig. 5: Interaction performance with varying k (CH)

EXPERIMENTAL EVALUATION

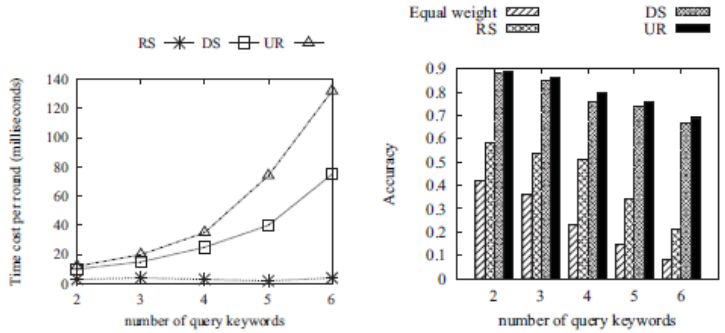


Fig. 6: Interaction performance with varying query keywords (CH)

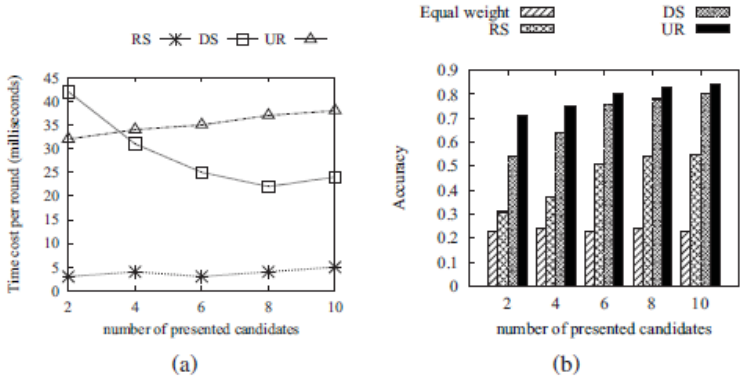


Fig. 7: Interaction performance with varying κ (CH)

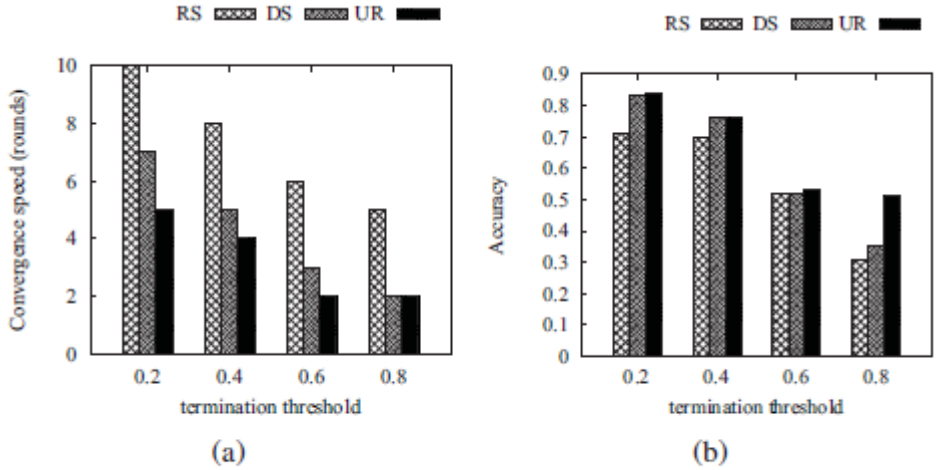
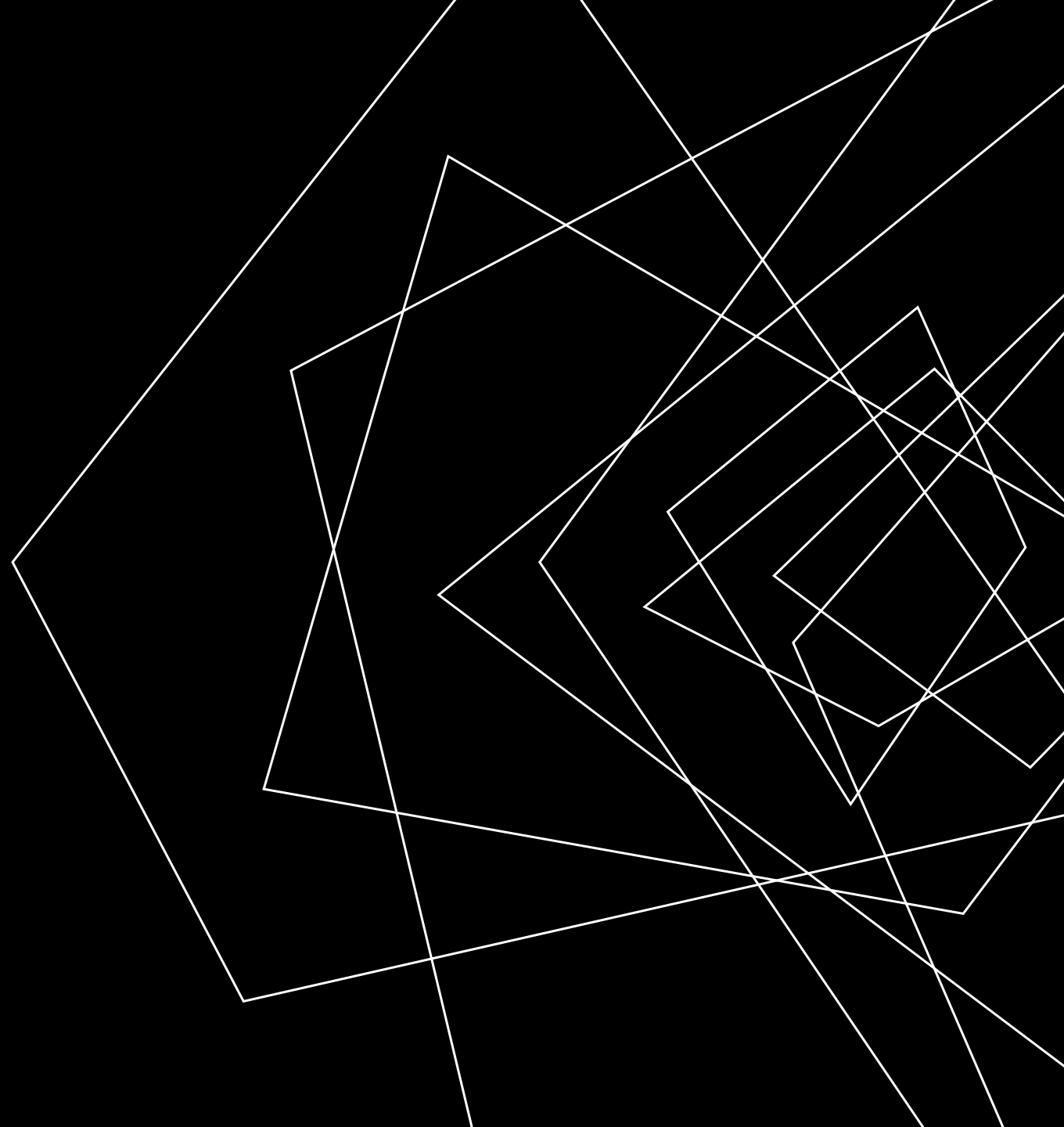


Fig. 8: Effect of τ (CH)

EVALUATE AND COMMENT



1) the importance of the problems addressed
   - With the rapid transformation of web clients from desktop computers to mobile devices increasing volumes of geo-textual objects are becoming available on the web that represent Points of Interest (PoIs).
2) the novelty of the proposed solutions
   - Interactive Top-k Spatial Keyword Queries is the most state-of-art approch with adding user's feedback.
3) the technical depth
   - Authors also presented a sophisticated approach to examinate ITkSK performance, with solid derivation of theorems.
4) potential impact
   - Empirical study based on real PoI datasets verifies our theoretical observation that the quality of top-k results in spatial keyword queries can be greatly improved through only a few rounds of interactions.
5) the quality of the presentation
   - This article have shown us what a well-written paper should be and how to deliver the idea to readers clearly.

THANK YOU

Any question?

Yu-Pin Liang (Jasmine)

yupinl@iastate.edu