Technical Disclosure Commons

Defensive Publications Series

October 2020

On-device Query Caching For Enhancing Zero-Prefix Query Suggestions

Keun Soo Yim

Konhee Cha

On-device Query Caching For Enhancing Zero-Prefix Query Suggestions

ABSTRACT

User interfaces (UI) that provide search functionality, e.g., search boxes, virtual assistants, etc. often include mechanisms that provide users with query suggestions within the UI. Query suggestions presented prior to receiving any input from the user are referred to as zero-prefix query suggestions. Zero-prefix query suggestions are typically derived by a ranking algorithm that is based on recently submitted and/or recurrent queries, accessed from a user-permitted server-side query cache. However, resource and operational constraints of a server cache can result in suboptimal zero-prefix query suggestions. This disclosure describes the implementation of a local on-device cache to overcome these limitations and improve the relevance and effectiveness of zero-prefix query suggestions.

KEYWORDS

- Query suggestion
- Zero prefix
- Zero-prefix suggestion
- Search suggestion
- Query cache
- Cache invalidation
- Recent queries
- Recurring queries
- Query context
- Virtual assistant

BACKGROUND

People routinely engage in search whenever using applications and services, or invoke a virtual assistant to search for desired information. User interfaces (UIs) and user experiences (UX) that involve search functionality often include mechanisms that provide users with query suggestions within the UI. These query suggestions usually fall into two categories: (i) those that are generated, displayed, and dynamically updated based on partial or complete user input, and

(ii) those that are presented prior to receiving any input from the user. The latter type of query suggestions are referred to as zero-prefix query suggestions as they are generated without any user input that can serve as a prefix for the suggestions.

Zero-prefix query suggestions are an effective technique to help users discover product features and capabilities, and to increase their efficiency by proactively providing queries that are likely to be submitted based on predicting likely user needs. Optimizing the relevance and effectiveness of the presented zero-prefix query suggestions can thus significantly enhance metrics related to the user experience of the respective product or service.

DESCRIPTION

Zero-prefix query suggestions can be derived by a ranking algorithm that is typically based on recently submitted queries, accessed with user permission. However, storing and analyzing recent queries to provide real-time zero-prefix query suggestions is technically challenging to scale. These challenges can be addressed by an indexing infrastructure that employs an in-memory cache. Given the storage limitations of the cache, it is limited only to the recent query data of active users. As a result, users who resume their search session after a period of inactivity can experience a latency penalty, creating a cold start with missing or low-quality zero-prefix query suggestions.

Alternatively, or in addition, zero-prefix query suggestions can be based on queries that recur, e.g., at specific times and locations. For example, a user can inquire about the morning news before leaving home for work. With user permission, a user's query history can be used to detect queries that are issued on a repeated basis, and zero-prefix query suggestions can be generated based on recurring queries that match the current contextual conditions, such as time, location, etc.

This disclosure describes techniques for improving the relevance and effectiveness of zero-prefix query suggestions presented by a search or virtual assistant interface. To that end, the techniques involve the use of a local on-device cache, implemented with user permission. The on-device cache is used to mirror recent and/or recurring queries that are cached on the server side. In cases where the server-side cache cannot supply zero-prefix query suggestions, the suggestions are obtained from the local on-device cache.

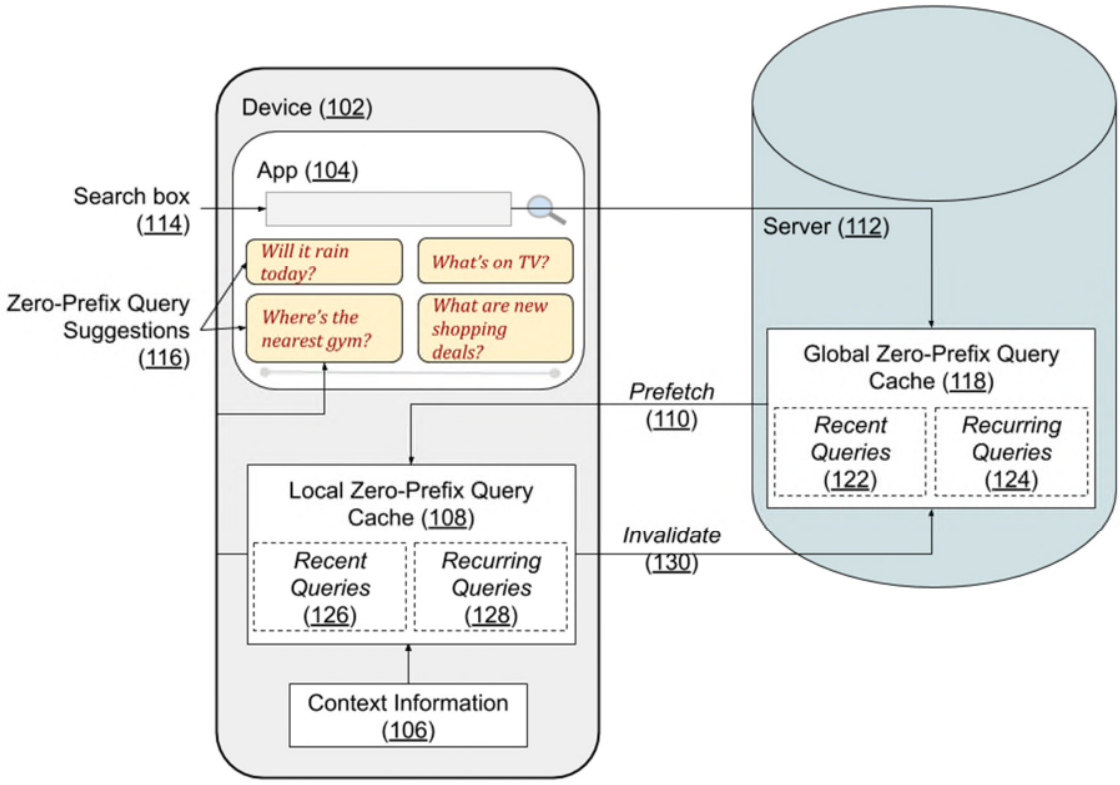

Fig. 1: Employing an on-device query cache for zero-prefix query suggestions

Fig. 1 shows an example of operational implementation of the techniques described in this disclosure. An app (104) (e.g., a virtual assistant or any other application) on a user device (102) provides search capabilities via a search box interface (114) or the like. Prior to providing any search input, the user receives zero-prefix query suggestions (116) that are obtained from an on-device query cache (108) that includes recent (126) as well as recurring (128) queries.

The on-device cache is populated by prefetching (110) corresponding recent (122) and recurring (124) queries from a global zero-prefix query cache (118) maintained on a server (112). If the user permits, relevant context information (106) is used to determine whether the current conditions match those for any of the recurring queries within the cache. Whenever any implicit or explicit user action indicates that any of the cached zero-prefix query suggestions are to be deleted from the local cache, the corresponding query is invalidated (130) and deleted from the global cache as well.

Specifically, if the user permits, the on-device cache is used to prefetch recurring query suggestions in the server cache that are likely to be applicable to triggering conditions expected within the upcoming period marked by a set time of N hours, where N can be any suitable value, such as 24 (representing a day). Similarly, with the user's permission, recurring query suggestions in the server cache that pertain to locations in the vicinity of the user's current location are prefetched to the on-device cache. In general, prefetching processes are used to fill the on-device cache with recurrent queries that are likely to match the user's current and upcoming context, such as time, location, device state, etc.

The entries for any queries that are time-dependent are marked within the local cache to expire after the specified time has elapsed. In contrast, other types of locally cached recurrent queries are not marked for automatic invalidation. For instance, a locally cached query connected to a location other than the user's current location is not marked for invalidation even if the user is not at the location associated with that query.

Whenever a user enters a mode that invokes the search functionality - e.g., launches the virtual assistant or search app - zero-prefix suggestions are obtained from the on-device cache and presented prior to receiving any search input. At the same time, any locally cached time-

specific queries are invalidated if the associated time has already passed. Apart from such automatic invalidation, users have the option to make explicit requests to delete a cached query. In such cases, the query is deleted from the on-device cache and a corresponding invalidate command is relayed to the global cache to delete the cached query from the server side as well. Further, users are provided with options to disable query caching, temporarily or permanently.

The prefetching can be performed in the background using an appropriate mechanism, such as the use of an application programming interface (API). With user permission, prefetching can be triggered by one or more relevant events, such as the user invoking the search UI, within a short time after the user submits a query, at periodic intervals, etc.

The described techniques can be implemented within any application or service that includes a search functionality. Alternatively, or in addition, the techniques can be integrated with the search capabilities offered by a virtual assistant. The zero-prefix query suggestions can be presented using any suitable UI mechanisms, such as visual display, audio output, etc. The various implementation parameters, such as time intervals, context conditions, prefetching frequency, etc., can be set by the developers and/or specified by the users and/or determined dynamically at runtime. Implementation of the described techniques can enhance the relevance and quality of zero-prefix query suggestions, thus enhancing the UX of search functionality within applications and services and augmenting the capabilities of a virtual assistant.

Further to the descriptions above, a user may be provided with controls allowing the user to make an election as to both if and when systems, programs or features described herein may enable collection of user information (e.g., information about a user's current or prior queries, usage of apps, social network, social actions or activities, profession, a user's preferences, or a user's current location), and if the user is sent content or communications (e.g., cached queries)

from a server. In addition, certain data may be treated in one or more ways before it is stored or used, so that personally identifiable information is removed. For example, a user's identity may be treated so that no personally identifiable information can be determined for the user, or a user's geographic location may be generalized where location information is obtained (such as to a city, ZIP code, or state level), so that a particular location of a user cannot be determined. Thus, the user may have control over what information is collected about the user, how that information is used, and what information is provided to the user.

CONCLUSION

This disclosure describes techniques for improving the relevance and effectiveness of zero-prefix query suggestions. A local on-device cache is utilized to prefetch recent and/or recurring queries cached on the server side. Prefetching can be based on likely triggers in the near future. When a user enters a mode that invokes the search functionality, zero-prefix suggestions are retrieved from the on-device cache and presented to the user, prior to receiving any search input. The locally cached queries that are past their applicable time or are subject to the user's cache deletion requests are invalidated and removed from local and server-side caches. The described techniques can be implemented within any application or service that includes a search functionality, e.g., a virtual assistant. Implementation of the described techniques can enhance the relevance and quality of zero-prefix query suggestions thus enhancing the user experience of search.

REFERENCES

1. Kanefsky, Steven T. "Predictive query suggestion caching." U.S. Patent 8,560,562, issued October 15, 2013.