August 2022

Templatized, Attribute-driven Search to Generate Stories

Nithya Kota

Maneesh Dewan

Ganesh Mallya

Jatin Chhugani

Templatized, Attribute-driven Search to Generate Stories

ABSTRACT

This disclosure describes techniques to automatically generate stories for users based on their spending histories. The stories can be formed by aggregating or clustering transactions around common themes or attributes using templatized search. Templatized searches are filled on the fly, and with user permission, are executed over the user's transaction history to obtain matching results. The search results are used to form a story that can be displayed to the user in engaging and insightful ways, e.g., spending locations highlighted on maps; spending histograms; spending versus time; summary statistics (sum, average, max/min); etc.

KEYWORDS

- Spending pattern
- Spending stories
- Payment app
- Financial app
- Transaction history
- Personal finance
- Templatized search
- Attribute-driven search

BACKGROUND

Stories - easy-to-digest summaries of interesting moments and experiences in the user's life - are a popular format for viewing content in various applications such as social media or chat, photo libraries, digital maps, etc. With user permission, such applications can automatically create stories from a user's data, e.g., photos, social media posts, map locations, etc. For example, stories based on people and location can be shown in a photo library app. Such stories can include, e.g., photos taken at the same location over time, photos of the same person(s) taken at different locations, memories from the same time period from prior years, etc. identified based on the user data. The story can then be a short slideshow, video, or other content that include the identified photos or other content.

DESCRIPTION

This disclosure describes techniques, implemented with specific user permission, to automatically create stories for users based on their spending histories, as available on a payment app. The stories can be based upon standalone user transaction history or upon comparison to anonymized demographic peer groups. Spending stories can be formed by aggregating or clustering transactions around a theme (e.g., ‘Friday nights,’ ‘weekend’); transactions that have similar attributes (spending on coffee in June); transactions that are part of specific user journeys (e.g., home renovation, trip to Hawaii); etc.

Given a set of themes or attributes (‘Friday nights,’ ‘Coffee,’ ‘Hawaii’), user transactions are aggregated or clustered using templatized search. Templatized searches are filled on the fly, e.g., by seeding with a templatized story name. The search is executed over the user’s transaction history, e.g., by evaluating the similarity of transactions against the story name. The similarity function can be obtained via several techniques. An example similarity function is the sentence or bag-of-words similarity between the filled template and relevant columns of the transaction table. Similarity evaluation can be aided by embeddings of the sentence or the bag-of-words.

Results of templatized search are used to generate a story that can be displayed to the user. With user permission, the choice of the filled templates can be personalized based on both the user and the transactional context via machine learning (ML) models. For example, the template/filling features can be provided as input to ML models such as multi-armed bandits, etc.

For example, a template ‘{type}{category} spending during {time},’ can result in a single instance of a filled template ‘{Indian}{grocery} spending during {july 2021}.’ In this example, clustering occurs due to the combined similarities between the transaction category, the transaction detail, and the transaction time against the relevant slots in the template. One or more fields in the template can be optional. For example, one instance of '{type}{category} spending during {time}' can be 'spending during {weekend},' which ignores 'type' and 'category.'

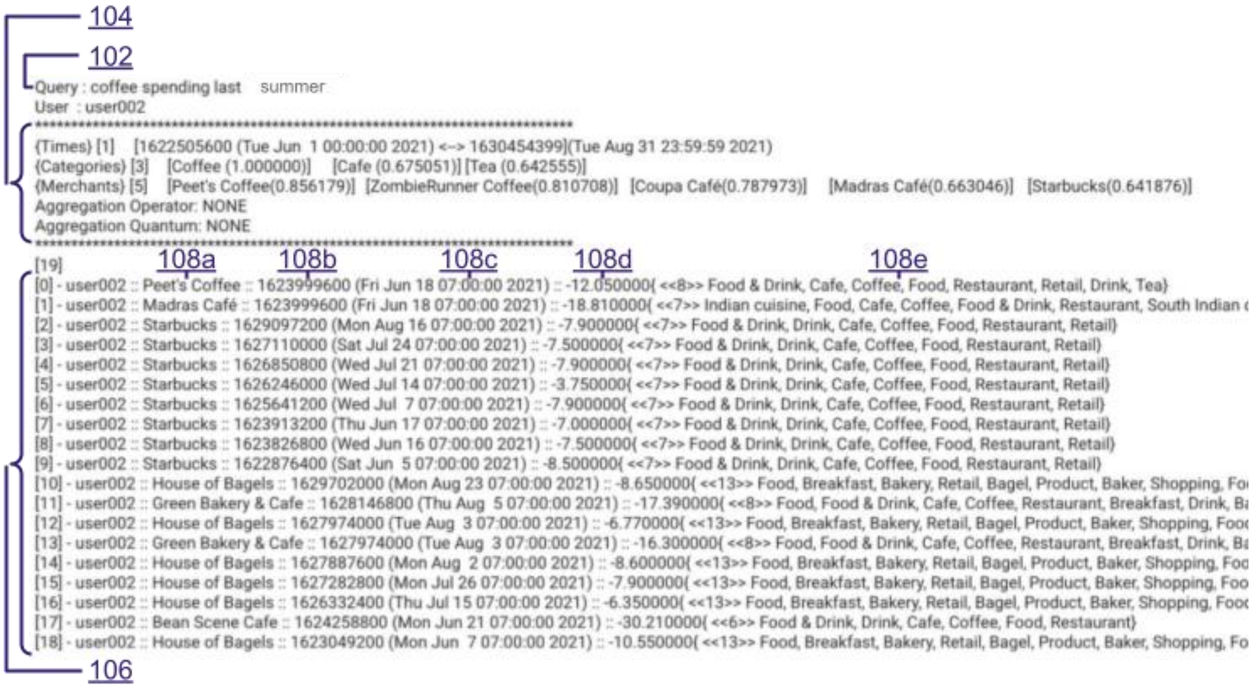

Fig. 1: Generating results from a templatized search over a user's transaction history

Fig. 1 illustrates generating results from a templatized search over a user's transaction history, accessed with user permission. A templatized search query (102) specifies attributes 'coffee' and 'last summer.' The templatized search query is expanded (interpreted) in (104), resulting in the following parameters:

- a 'Times' parameter (Jun. 1 00:00:00 2021 to Aug. 31 23:59:59 2021) that specifies transaction times that fall within 'last summer;'

- a three-vector ‘Categories’ parameter comprising words likely to correspond to ‘coffee,’ along with their confidences (e.g., ‘Coffee,’ with confidence 1.0, ‘Cafe’ with confidence 0.675051, ‘Tea,’ with confidence 0.642555);
- a five-vector ‘Merchants’ parameter comprising names of likely vendors of elements in the ‘Categories’ vector, along with their confidences; etc.

Other parameters, relating to aggregation, are left unfilled since there is no indication in the query that the user wants to aggregate spending across different transactions.

The search, including its parameters ‘Times,’ ‘Categories,’ ‘Merchants,’ is executed over the user’s transaction history, resulting in identification of 19 transactions (106) that correspond to the original query ‘coffee spending last summer.’ The transactions include the merchant name (108a); the transaction ID (108b); the date and time (108c); the dollar amount (108d); the vendor description (108e); the vendor location; etc. The retrieved search results and the data they encapsulate can be used to generate and display a user story around coffee spending last summer.

Additional examples of filled templates for searches through user transactions include ‘bookstore expenses in San Francisco over the last three weeks,’ ‘last year’s Hawaii vacation expenses towards lodging,’ ‘music fees 2020,’ ‘streaming app spending last weekend,’ etc.

Creation of templates for searching

Developers can create and load suitable templates, e.g., of the form ‘{type}{category} expenses during {time},’ into an application (e.g., a payment app or other app that has user-permitted access to payments data) that can generate user stories. Templates can be filled automatically by analyzing user searches. For example, if users typically search through their transactions using queries such as ‘coffee,’ ‘cafe,’ or the name of a coffee shop, it can serve as an indicator for creating a cluster ‘Coffee spending in {time}’ for a story of that form.

With user permission, template and story personalization can be performed based on user feedback. Multiple stories are presented to the user. User interaction with each story is evaluated to determine embeddings of user features, story features, and context that are of value to the user, e.g., improve the user's engagement with the app. For example, based on searches conducted by a user's demographic peer group, the user is presented with stories relating to coffee, automobiles, vacations, etc.

In another example, it may be detected that the user consistently shows interest in automobiles, e.g., by having greater engagement with automobile stories. Such user feedback indicates that automobile stories are of greater value to the user than other categories of stories. Automobile stories are presented to such a user relatively more often (and non-automobile stories somewhat less often). Even as automobile stories are surfaced more often, to enable the emergence of a fuller picture of the user's preferences, non-automobile stories may continue to be surfaced occasionally. Learning techniques such as contextual multi-armed bandits, explore-exploit reinforcement learning, etc., can be used for achieving such personalization based on user feedback.

Display of results

Results from templatized queries can be presented as stories to the users in several ways, such as:

- *Map-based stories:* The results are shown as locations on the map. For example, coffee spending can be geographically visualized through a map with location pins of coffee shops where the user has spent money.
- *Calendar or time-based stories:* The results are shown as transactions that take place over certain dates or times.

- *Aggregation-based stories:* The story includes an aggregation of the search results. An example aggregation includes sum, average, maximum, minimum, etc.

Stories can be arranged in layers, such that upon initial presentation, the user is provided a summary (e.g., a map with location pins in a map-based story). The user can dive deeper into sections of the summary, e.g., by selecting appropriate regions of the summary by touching or clicking on a particular location pin in a map-based story. The user can explore further until the full details of a single transaction become visible.

Displaying stories based on spending histories, as described herein, enables users to obtain interesting and useful insights about their spending and can improve user engagement with the payment app. Spending-related stories can be displayed in engaging ways, e.g., spending locations highlighted on maps; spending histograms; spending versus time; summary statistics (sum, average, max/min); etc. The user is provided with options to disable the stories feature, to restrict access to certain transactions (or transaction types), to limit stories to particular templates, etc. Further, stories are presented only in safe contexts, e.g., upon confirming that the user has authenticated themselves to the device, the user is at a known location, etc. and at suitable times.

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 spends via a payments app, 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 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 to automatically generate stories for users based on their spending histories. The stories can be formed by aggregating or clustering transactions around common themes or attributes using templatized search. Templatized searches are filled on the fly, and with user permission, are executed over the user's transaction history to obtain matching results. The search results are used to form a story that can be displayed to the user in engaging and insightful ways, e.g., spending locations highlighted on maps; spending histograms; spending versus time; summary statistics (sum, average, max/min); etc.