Location-aware Mobile Advertising System

Berk KARAOĞLU Can Çağdaş CENGİZ Enes TAYLAN

Project Description

The system we are about to propose is basically a mobile advertising system. It enables the users to display relevant advertisements based on user query and geo-location. The general structure of the system is shown in Figure 1.



Figure 1 - General Structure of the System

There are two types of users in the system: advertisers, and the people looking for advertisements. When an advertiser adds a new advertisement using the Web Client, the system saves it to the database and updates the index table. When a search query is entered using the mobile client, it automatically obtains the user's location via GPS, and sends both the query and the location to the server. Server searches the index table for the terms occurring in the user query, and generates a final, sorted output considering the user location.

Motivation / Importance

14% of the mobile phones in Turkey are smart phones. Also, most of the Turkish people regularly follow campaigns in their cities. But; the problem is that, how can an advertisement be relevant for a person who lives in the other side of the city. We are going to solve this problem by including the geolocation to the traditional information retrieval methods.

Google Ads is the leading edge in advertising based on user relevance. Google creates indexes based on e-mail contents, and user profile details. In addition to the criteria that Google uses, we are going to benefit from user location, to serve better and more meaningful search results.

Methodology

The system that we are going to build consists of four parts which are a mobile client for Android phones, a Java web service (servlet or jsp), MySQL database and a TomCat Server to host the web services.

The web service will take the user queries, cluster the advertisements according to their geo-locations, maintain the database, regularly generate the index table. Inside, it will also have an indexing algorithm implemented, which explained by Zobel & Moffat [1].

Our contribution in information retrieval will be the following. In addition to common term weighting parameters, we are going to propose a new parameter, based on location clusters to calculate term weights for indexing web service. The system will have role based access for enterprises and regular customers.

Android client is the only way allowing users to interact with the system. There will be a search screen, which consists of a text box for the query, and a toggle to let the user choose whether sending the location or not. If user sets the toggle on, a background service will be started to make a geolocation request from the GPS satellites. After clicking the "Search" button, the device will connect to the web service which is stored in our server, and sends the query and the geolocation data (optional as explained above). The server responds with the relevant search results. We are planning to display the results in a table, and selecting a specific entry will open a new screen showing its details. There will also be an option for users to declare whether are they satisfied with the result. These feedbacks will be used to compare with the traditional approach which does not benefit from the geolocation.

Expected Results

We expect to get meaningful results based on user experiences. The mobile client will show the results with the location-aware model and non-location aware model. Users will rate the query results. According to the ratings of the results, we are expecting to get better scores in location aware model. The other metric will be the query time. We will try to keep the search times closer to the non-location aware model. We will achieve this by implementing the term weight calculation in an efficient way,

which is explained by Salton & Buckley [2].

References

1- J. Zobel, A. Moffat. Inverted files for text search engines. ACM Computing Surveys (CSUR) Surveys, Volume 38 Issue 2, 2006

2- G. Salton and C. Buckley. Term weighting approaches in automatic text retrieval. Information Processing and Management, 24(5):513-523, 1988