Improved Generation of Topic-Based Language Models for an App Search Engine

Inventors:

Natalia Hernandez Gardiol, Catherine Edwards

Section 1: Improved Generation of Topic-Based Language Models for App Search Engine

1. Improved generation of topic-based word probabilities
2. Post-processing of language model

Section 1: Improved Generation of Topic-Based Language Models for App Search Engine

Overview

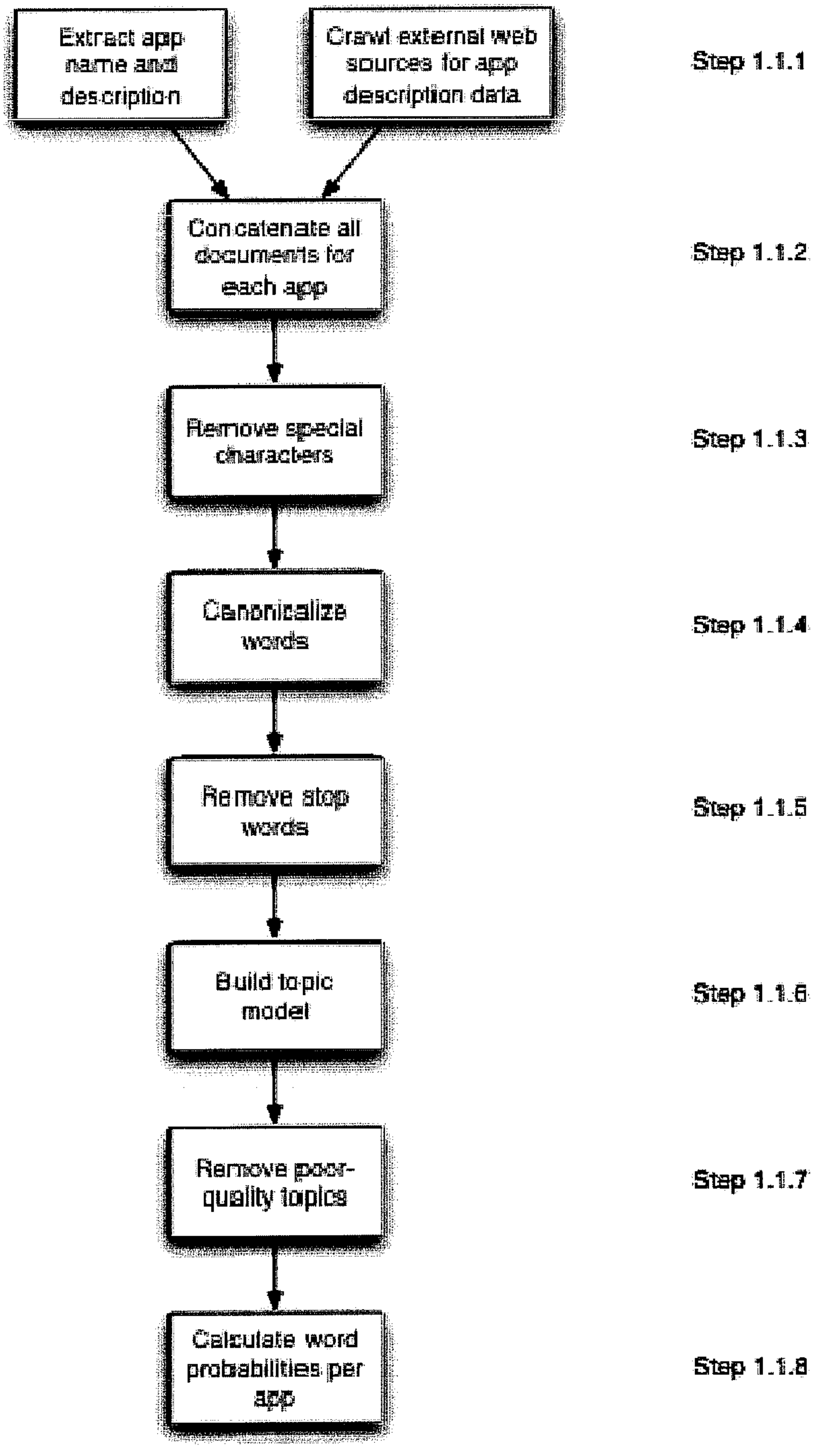
Chomp has developed an implicit search engine that enables searching for apps based on their function rather than their name. This differs substantially from web search in that the contents of the item being searched for are typically not accessible to the search engine. Further, the contents of the app do not correlate with users' query behavior. When querying for apps, users formulate queries that identify the function of that app. However, apps are typically overwhelmingly made up of content that are instances of the function, and do not describe (or even refer to) the actual function itself. For example, a messaging app may contain many message logs that do not refer to the app's function, but users wish to search for terms related to messaging rather than to the contents of the message logs.

In this document, we present an improvement to the language-model generating algorithm described in the provisional patent "App Search Engine." The improvements are centered around, improving the topic-model-based word probabilities (described in Figure 1, below) and the addition of a post-processing step (in Figure 2, below).

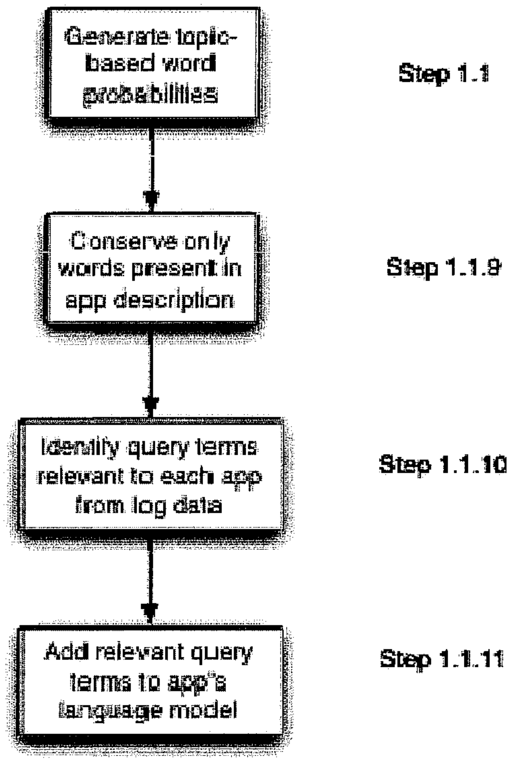
The improved process of generating topic-model-based word probabilities is as follows. First, a corpus of metadata capturing app function is assembled from various data sources for each app. The content of the corpus is normalized to a canonical form, and then the topic model is trained from this corpus. The resulting topic model is then used to learn a language model (i.e., a probability distribution over words) that represents each app's name and function.

In order to ensure relevancy and coverage, post-processing is then carried out. The first step is to eliminate words not pertinent to the app; and, the second step is to associate words deemed relevant from query logs.

1. Improved generation of topic-based world probabilities


Extract app name and description
Crawl external web sources for app description data
Step 1.1.1
Concatenate all documents for each app
Step 1.1.2
Remove special characters
Step 1.1.3
Canonicalize words
Step 1.1.4
Remove stop words
Step 1.1.5
Build topic model
Step 1.1.6
Remove poor-quality topics
Step 1.1.7
Calculate word probabilities per app
Step 1.1.8

2. Post-processing of language model


Generate topic-based word probabilities
Step 1.1
Conserve only words present in app description
Step 1.1.9
Identify query terms relevant to each app from log data
Step 1.1.10
Add relevant query terms to app's language model
Step 1.1.11