Search Engine for Local and Global Businesses

Anonymous Author(s)
Affiliation
Address
`email`

Abstract

Many users now-a-days look for local businesses or places to dine at on the web. With access to a lot of data online, one can go online and search for businesses and read about not only the service provided by the business but also experiences of people who have interacted with them. We have built a search engine that makes it easier for one to find businesses locally based on reviews and ratings provided by their respective customers. We not only provide businesses that a user is looking for, but also recommend other such businesses that the user might like. Many other features have been discussed in this report.

1 Motivation

Searching for businesses online has become easier now-a-days. Search engines such as Yelp's has provided users with information about businesses and also opinions of people who have used services provided by the business. Ratings and reviews go a long way in determining the success of a business and people invest considerable amount of time providing their experiences and suggestions as to how could one leverage the business' services to get a positive experience. Our main aim was to make use of this user-generated information to further enhance the information retrieved by our search engine.

We leveraged information about businesses and users to build a search engine that not only searches businesses queried by our users but also recommends similar businesses in and around the vicinity of their searches. This similarity is derived from the reviews provided for each business. We also provide recommendations based on user similarity. We used the Map-Reduce framework to get the user-based and the content based similarities. The subsequent sections provide detailed discussions on how we incorporated recommendations into our search engines. The search engine also performs location based personalization such that the retrieved businesses pertain to the region our search engine is being queried from.

We have built our search engine such that one could not only use it for searching for a specific business but also use free text to look for businesses that provide services which match the user's queries. For example, one could look for places that serve pizzas by querying *pizzas* on our search engine.

2 Design and Architecture

2.1 data

We used Yelp's academic dataset to build our search engine. The dataset provides business and review information for nearly 56,000 businesses belonging to Canada UK and the United States. There were a total of 1.9 million reviews and around 100,000 tips. The information was provided

1

in 3 different json files which were merged and written into one text file for each business. The following table shows ....:

1	Business Id
2	Name
3	Latitude and Longitude
4	Ratings
5	Address
6	Categories
7	Reviews
8	Tips

2.2 Document Processing

The business information for all 56,000 businesses were stored in different text files for all businesses. A business document consisted was based off a format that was followed for all businesses. Meta-data such as business id, business name, address, ratings, latitude and longitude were stored, one per line. This was followed by categories and attributes of the businesses which form the main part of the document. Every business has tags associated with it which provides vital information about the type of service it provides. For example, tags such as Mexican Restaurant, Mexican cuisine, Food, Restaurant would be provided for a Restaurant. Tags such as Medical Services would be associated with a Medical Clinic.

Apart from this, other attributes that provide additional information about the business such as Wheelchair access or Accepts credit cards or Wi-Fi enabled or Good for Kids and Family, are also included as categories. These tags help the user to get an idea about the business that they would like to know that businesses generally don't provide. This was followed by tips and then reviews. Since the number of tips and reviews would differ for each document. The total number of tips and reviews are also added as it would not only be useful for ease of processing, but would also be used during ranking.

While reading in the documents, some amount of pre-processing needed to be done, namely removal of words with unusually long length with repetitions of vowels. Words such as *good* were written with more vowels like *goooood* to portray the extent to which the service was good. Many such words with repetition of alphabets were trimmed down to not more than two. This processing was only done in the reviews text and the name and other attributes were unchanged. Stemming of words is also done to remove frequently occurring words or words that do not actually contribute to the ranking. Ease of matching of query terms with the document terms was also one of the reasons for stemming the words in the documents.

2.3 Indexing

Since, the users issue keyword based search queries, we build an index that acts a lookup for words that occur We read in 56,000 text files containing information about businesses along with their reviews and other meta-data and stored it in the form of an inverted index using byte encoding to compress it. We built an inverted index of words occurring in the review text and score them based on the occurrences in the text. Building an index based on names of businesses wouldn't have helped the engine because there could be businesses with the same names and also, it would allow for free text to be used for querying.

2.4 Ranking

2.4.1 Ranking based on Title, Categories and Reviews

Ranking based on the terms contained in the review text is more likely to retrieve relevant businesses. The first thing we do while retrieving is to try and match the queries with the title of the businesses, since retrieval of a business matching the query issued should be fast and thus the title is weighed heavily. We also look for matches in the categories the business belongs to.

2.4.2 Ranking based on number of Reviews and Ratings

Including number of reviews of a business in our ranking definitely improves the retrieval for a search query. The idea behind including number of reviews is that if a business has more reviews, then certainly a lot of people have used it for its services and this would further enhance the business' credibility. Just because a business has more reviews does not make it a good business. Including a measure of how good the business is for it's services in terms of ratings would retrieve top businesses. Addition of ratings with the number of reviews adds to the improvement and now the retrieval is not just better in terms of whether the query is matched to the business but also the credibility of the business and how good the business is for it's services. Number of reviews can be equivalent to the number of views of a document on a web page.

3   Implementation

3.1   Data

3.2   Recommendation using Map-Reduce framework

3.2.1   Algorithm

3.3   Ranking

In the previous section, we discussed the different features being used for ranking the businesses being retrieved by the engine. Intuitively one would think about using a linear ranking that generates scores based on different features, weighing them and add the scores linearly. But this doesn't improve the retrieval as will be discussed in the next section. Taking a product of the scores based on different features improves the retrieval to a certain extent.

The reason ranking is done multiplicatively is because if score from one of the features is high while the score from another is low, the overall score does not increase to a great extent. This way of aggressively ranking the businesses, results in retrieval of businesses that truly satisfy almost all the requirements.

$$\text{total score} = \frac{(\text{cosine score}) \text{x} (\text{score number of reviews}) \text{x} (\text{score ratings})}{\text{distance}}$$

3.4   Querying

One can search businesses directly by simply querying the name of the businesses. The engine returns the corresponding business the user was looking for along with similar businesses based on the similarity computed from term matches and similar businesses looked by other users.

The other way one could issue queries would be to find businesses that meet certain requirements like *Mexican Restaurants* or *Coffee houses*. The engine retrieves businesses that match the term not only in the Categories and the titles, but also in the review text. The resulting retrieval is ranked based on Cosine similarity, location, number of reviews and ratings.

Geographic location based retrieval is the other part of the project, we focussed on. The engine not only retrieves businesses that match the query but also makes sure the businesses are situated in the region from where the query is issued.

4   Evaluation

Since we built a search engine that retrieves businesses matched by the query being issued

Table 1: Data Description

PART	DESCRIPTION
Dendrite	Input terminal
Axon	Output terminal
Soma	Cell body (contains cell nucleus)

Acknowledgments

References

References follow the acknowledgments. Use unnumbered third level heading for the references. Any choice of citation style is acceptable as long as you are consistent. It is permissible to reduce the font size to 'small' (9-point) when listing the references. **Remember that this year you can use a ninth page as long as it contains *only* cited references.**

[1] Alexander, J.A. & Mozer, M.C. (1995) Template-based algorithms for connectionist rule extraction. In G. Tesauro, D. S. Touretzky and T.K. Leen (eds.), *Advances in Neural Information Processing Systems 7*, pp. 609-616. Cambridge, MA: MIT Press.

[2] Bower, J.M. & Beeman, D. (1995) *The Book of GENESIS: Exploring Realistic Neural Models with the GEneral NEural SImulation System.* New York: TELOS/Springer-Verlag.

[3] Hasselmo, M.E., Schnell, E. & Barkai, E. (1995) Dynamics of learning and recall at excitatory recurrent synapses and cholinergic modulation in rat hippocampal region CA3. *Journal of Neuroscience* **15**(7):5249-5262.