Discussion 11 - Recommender Systems

CUNY MSDA DATA 607

Duubar Villalobos Jimenez mydvtech@gmail.com

April 20, 2017

Recommender Systems: Google
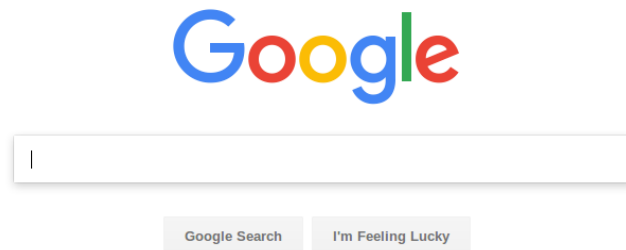

Figure 1: Source: https://cloud.google.com/prediction/docs/

From Wikipedia:

Google is an American multinational technology company specializing in Internet-related services and products. These include online advertising technologies, search, cloud computing, software, and hardware.

Google has an extensive list of products and services and can be viewed here: https://www.google.com/intl/en/about/products/

Google Cloud Prediction API Documentation

As per their description:

"Google Cloud Prediction API provides a RESTful API to build Machine Learning models. Prediction's cloud-based machine learning tools can help analyze your data to add various features to your applications, such as customer sentiment analysis, spam detection, recommendation systems, and more."

Quickstart

Something very interesting is the very well documented steps on how to create and test models; not to mention the disponibility of their resources to increase speed, collaboration and efficiency.

How-to Guides

Developer Guide

This page describes the syntax for setting up and running predictions, as well as how to design a prediction model appropriate for our needs.

A very important recommendation is that they highly recommend that you try the Quickstart to get a better idea of how the Prediction API works.

This developer guide is rich with useful information as follows:

- Using the API
- Examples
- Training Models
- Training Data

Smart Autofill Spreadsheets Add On

Smart Autofill is a Google Spreadsheets Add On that uses the Prediction API for performing Machine Learning directly in a Google Spreadsheet.

This guide is very useful if we want to perform various task.

Examples:

- Estimate car prices
- SmartAutofill with text data

Use Cases

a) Sentiment Analysis

This document explains how to create a basic sentiment analysis model using the Google Prediction API. A sentiment analysis model is used to analyze a text string and classify it with one of the labels that you provide; for example, you could analyze a tweet to determine whether it is positive or negative, or analyze an email to determine whether it is happy, frustrated, or sad. You should read the Hello Prediction page before reading this introduction.

Also it provides a series of steps explained as follows:

Step 1: Collect Data.

Step 2: Label Your Data.

Step 3: Prepare Your Data.

Step 4: Upload Data to Google Cloud Storage.

Step 5: Train a Model with the Google Prediction API.

Step 6: Make Predictions with the Google Prediction API in your application.

Step 7: Update Your Model with New Data.

b) Purchase Prediction

In this case it provides examples on how a site that sells beer, wine, and cheese, and you want to predict whether a visitor will be interested in wine, given their purchase history.

c) Spam Comment Detection

In this case, the documentation provides an example on which someone is trying to detect whether user-submitted comments to a web page are actual comments, or just spam. This is a similar task to spam email detection, but you have a lot fewer signals to go on.

APIs and Reference

API Reference

In this section it provides a tremendous amount of information on how this API reference is organized by resource type. Each resource type has one or more data representations and one or more methods.

Resource types:

- Hostedmodels
- Trainedmodels
- Standard Query Parameters

Samples and Libraries

Something very nice is that The Google Prediction API is built on HTTP and JSON, so any standard HTTP client can send requests to it and parse the responses.

However, the Google APIs client libraries provide better language integration, improved security, and support for making calls that require user authorization. The client libraries are available in a number of programming languages; by using them you can avoid the need to manually set up HTTP requests and parse the responses.

- Go
- Java
- Javascript
- .Net
- Node.JS
- OBJ-C
- PHP
- Python
- Ruby

Prediction Gallery

They also have a set of predictions gallery:

Language Identifier as a Demo only. It analyzes a sentence to determine whether it is English, Spanish, or French.

Tag Categorizer as a demo only. Tags a given comment as pertaining to android, appengine, chrome, or youtube. Training data comes from a collection of social media comments.

Sentiment Predictor as a demo only. Analyzes the sentiment of a short English-language text snippet.

One of the cool things they allow is that we can submit our own model.

PMML Preprocessing

Something very interesting is that The Google Prediction API now supports preprocessing your data against a PMML (Personalized Print Markup Language) transform specified using PMML 4.0 syntax. Google Prediction does not currently support importing a complete PMML model that includes data. PMML preprocessing can be applied to both categorical or regression models.

It provides examples and a detail schema.

Performance Tips

Some of the very important things to consider that they provide in their document is that it covers some techniques you can use to improve the performance of your application. In some cases, examples from other APIs or generic APIs are used to illustrate the ideas presented. However, the same concepts are applicable to the Google Prediction API.

- Using gzip
- Working with partial resources

Batching HTTP Requests

The Google Prediction API supports batching, to allow clients to put several API calls into a single HTTP request.

It provides examples and step by step, simplifying the process of how to.

Building the Google Recommendation Engine

"GCP frees you from the overhead of managing infrastructure, provisioning servers and configuring networks. To let innovators innovate and let coders, well, just code."

One of Google philosophies is that we don't need to get our hands "dirty" if we try to create or innovate, hence they provide a series of services that help achieve that goal.

One of their goals is to enhance collaboration and expertise in many ways; by providing cloud computing services they provide incredible robust tools that help minimize the amount of work to be done.

Their current approach is based in simplicity and centralized expertise; something useful when sometimes we feel trapped in multiple forms the web has to offer.

Their environment is always evolving with products and services; hence sometimes they discontinue the support on some of the under performing services or older products.

A very important part of their commerce is the incredible spread amount of products and services they offer; that provides strength in multiple levels and touches many lives in a split of a second; making them quite important factor of our civilization as we know it.

Google Scenario Design

Analysis

1) Who are your target users?

Google's target users are all (businesses and users); it all depends of the product or service.

2) What are their key goals?

Mission: Organize the world's information and make it universally accessible and useful.

Purpose: Allow users to be able to find information in many different languages; check stock quotes, maps, and news headlines; lookup phonebook listings for every city in the United States; search billions of images and peruse the world's largest archive of Usenet messages. In addition, support thousands of advertisers to use Google's AdWords program to promote their products and services on the web with targeted advertising.

Vision:

- Short-term objectives: are to expand the workforce for anticipated growth, expand further into international markets, and continue developing new products.

- Long-term objective: delivering new advertising technology. Expanding the workforce will help achieve the long-term goals. Google's organization structure is primarily functional but also includes a few geographical organizations.

Philosophy: Google does not document a Vision or Values on the Google website. They do state a philosophy on the Google website, some are listed below:

- Focus on the user and all else will follow

- It's best to do one thing really, really well.

- Fast is better than slow.

- You don't need to be at your desk to need an answer.

- No pop-ups.

3) How can you help them accomplish those goals?

By using, evaluating and promoting their products and services.

Reverse Enginiering

While this is a very difficult task for just a few hours of work and preparation, I have found a very good article in which the Harvard Business Review tried to Reverse Engineering Google's Innovation Machine (https://hbr.org/2008/04/reverse-engineering-googles-innovation-machine).

As you can notice this is no easy task because for this to be performed has to be done as a whole and not necessarily just one part of it since all is interconnected as a "living" organism or entity in this case.

Recommendations

Sometimes their documentation simplicity is confusing, and I think it should be improved in a better way specially the navigation on some of their services and APIS.

Conclusion

The Google Prediction API is a generic machine learning service and can be used to solve almost any regression or classification type problem.

A downside is that some of the models they present are provided temporarily for demonstration purposes only and may be removed without notice at any time. Therefore they advise against building commercial and/or production services using those models.

I hope, that they can can enable a dynamic way to provide some of their services in a more affordable way for some of the specialized services, research or discoveries they do.

References:

https://en.wikipedia.org/wiki/Google

https://www.google.com/intl/en/about/products/

https://en.wikipedia.org/wiki/Recommender_system

https://cloud.google.com/why-google/

https://hbr.org/2008/04/reverse-engineering-googles-innovation-machine

https://github.com/grahamjenson/list_of_recommender_systems

http://www.123helpme.com/strategic-planning-process-google-view.asp?id=167474