
Google™
10

BigQuery and Prediction APIs
Getting more from your data with Google

Amit Agarwal
May 20th, 2010

View live notes and ask questions about this session on Google Wave
http://bit.ly/dr01ws

Overview

- Big Data - Challenging and Important

- Google has tools for deep data analysis

- Now you can use these tools

Overview

- Big Data - Challenging and Important

- Google has tools for deep data analysis

- Now you can use these tools

- Announcing two new APIs to get more from your data:
  1. BigQuery
  2. Prediction API

Benefits

- Built on Google technology
- Scalability
- Security
- Sharing
- Easy integration with Google App Engine, Google Spreadsheets, ...

Using Your Data with BigQuery & Prediction API

1. Upload

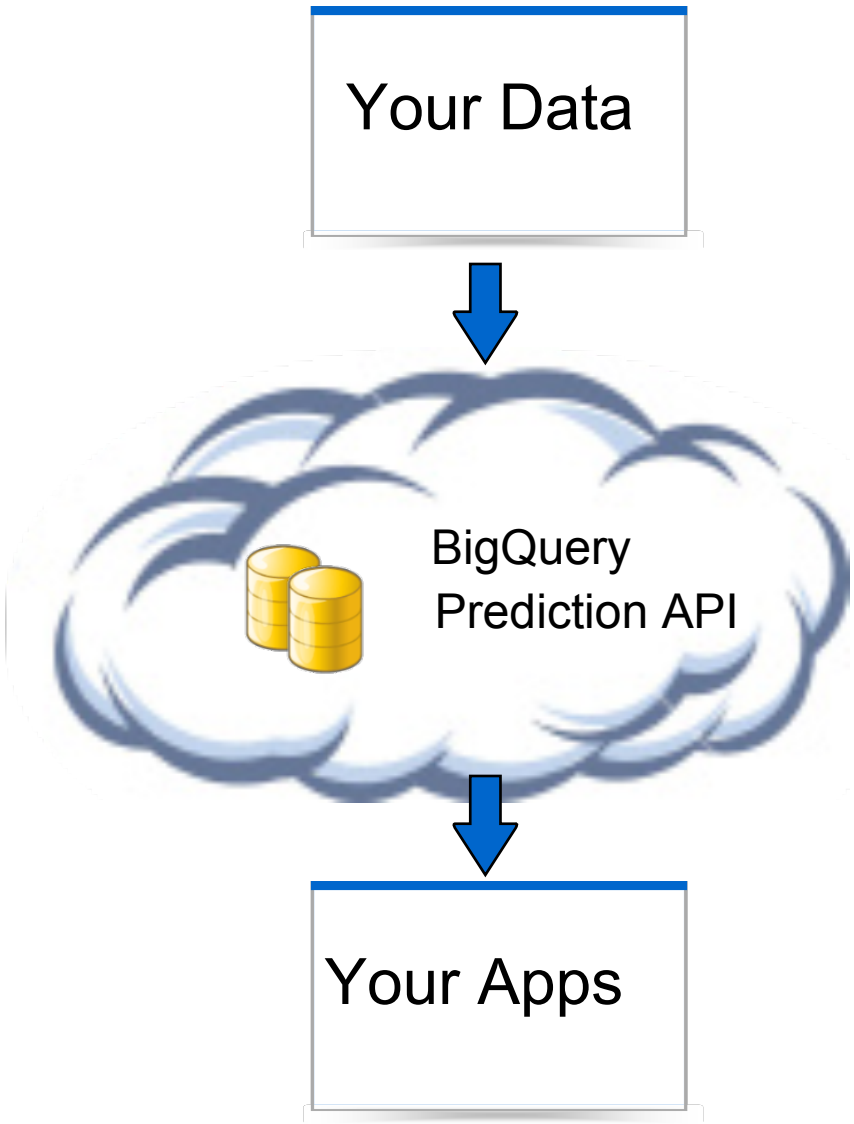
Upload your data
to Google Storage

2. Process

Import to tables
Train a model

3. Act

Run queries
Make predictions

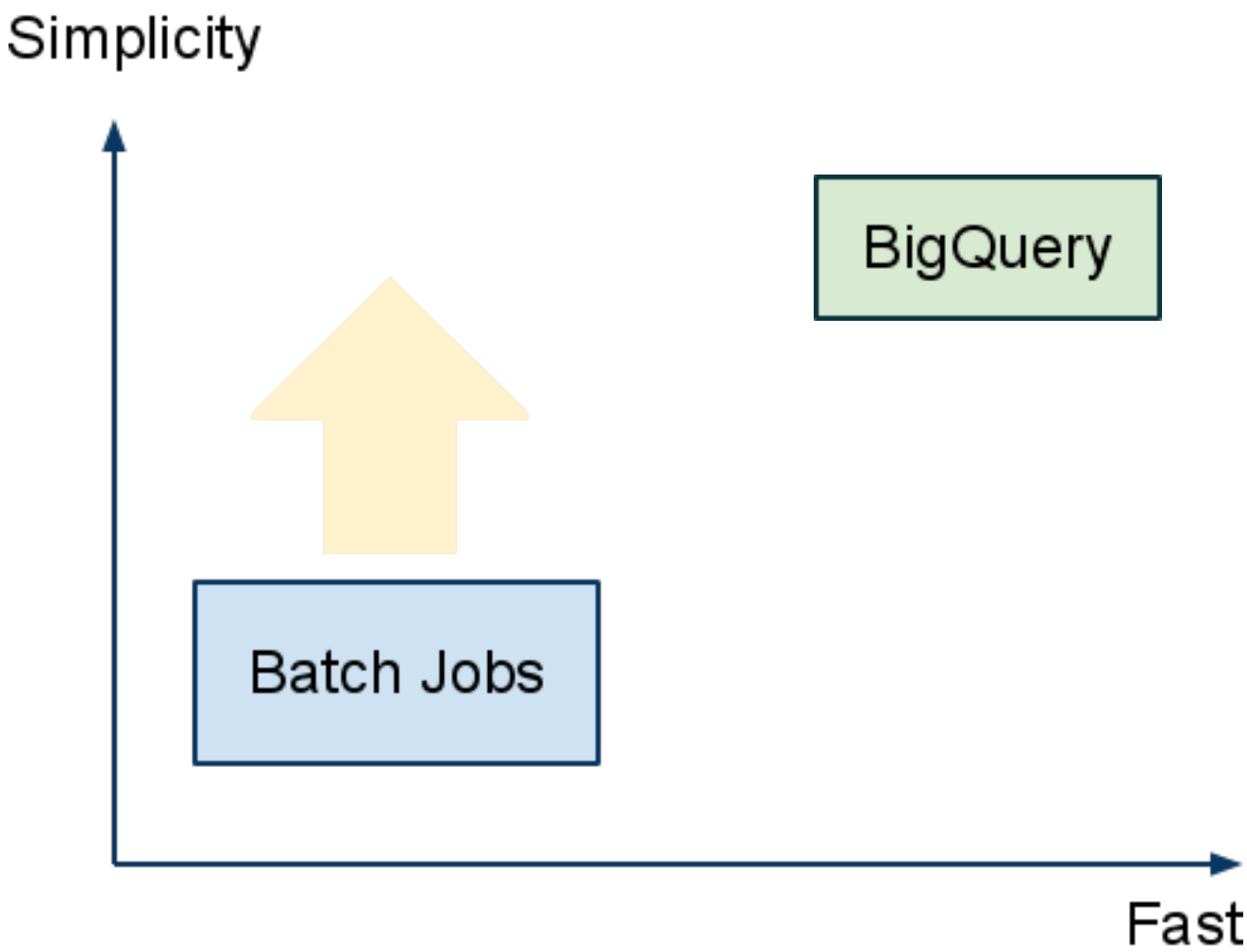


BigQuery
Interactive Analysis of Big Data

Siddartha Naidu
May 20th, 2010

Big Data is Challenging

Starts with **Scale**

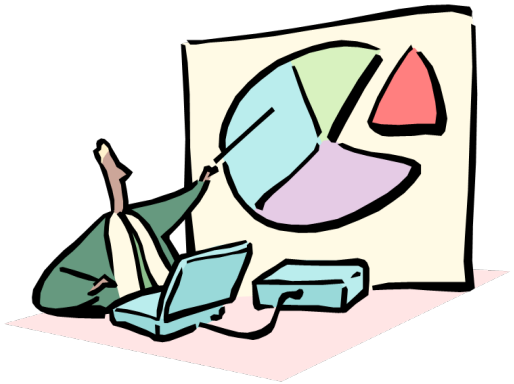


Many Use Cases ...

Interactive Tools

Spam

Trends Detection



Web Dashboards

Network Optimization

Demo: Analyzing M-Lab

An open platform for advanced network research
http://www.measurementlab.net/

BigQuery

Sign out

Enter SQL query

```
SELECT COUNT(*) FROM [mlab.v10];
```

Run

Google I/O 10

Demo: Exploring M-Lab

BigQuery

Sign out

Enter SQL query

```
SELECT TOP(connection_spec.remote_hostname) Host, COUNT(*)
FROM [mlab.v10] WHERE
IS_EXPLICITLY_DEFINED(connection_spec.remote_hostname) AND
connection_spec.remote_hostname CONTAINS 'google';
```

Run

Host	COUNT(*)
216-239-45-4.google.com	52846352
216-239-44-65.google.com	107553
googlebroadband186.excellmedia.net.60.13	50181
googlebroadband187.excellmedia.net.60.13	29927
Host	**COUNT(*)**

Google 10 I/O

Key Capabilities of BigQuery

- Scalable: Billions of rows
- Fast: Response in seconds

- Simple: Queries in SQL

- Web Service
  - REST
  - JSON-RPC
  - Google App Scripts

Using Your Data with BigQuery

Upload to Google Storage

Import data into a BigQuery Table
- No need to define indices, keys, etc..

Execute queries via APIs
- No provisioning machines or resources

Writing Queries

Compact subset of SQL

- **SELECT ... FROM ...**
  WHERE ...
  GROUP BY ... ORDER BY ...
  LIMIT ...;

Common functions

- **Math, String, Time, ...**

Statistical approximations

- **TOP**
- **COUNT DISTINCT**

API in a Minute

GET /bigquery/v1/tables/{table name}

GET /bigquery/v1/query?q={query}

Sample JSON Reply:

```
{
 "results": {
  "fields": { [
   {"id":"COUNT(*)","type":"uint64"}, ... ]
  },
  "rows": [
   {"f":[{"v":"2949"}, ...]},
   {"f":[{"v":"5387"}, ...]}, ... ]
 }
}
```

Also supports JSON-RPC

Security and Privacy

Standard Google Authentication

- Client Login
- OAuth
- AuthSub

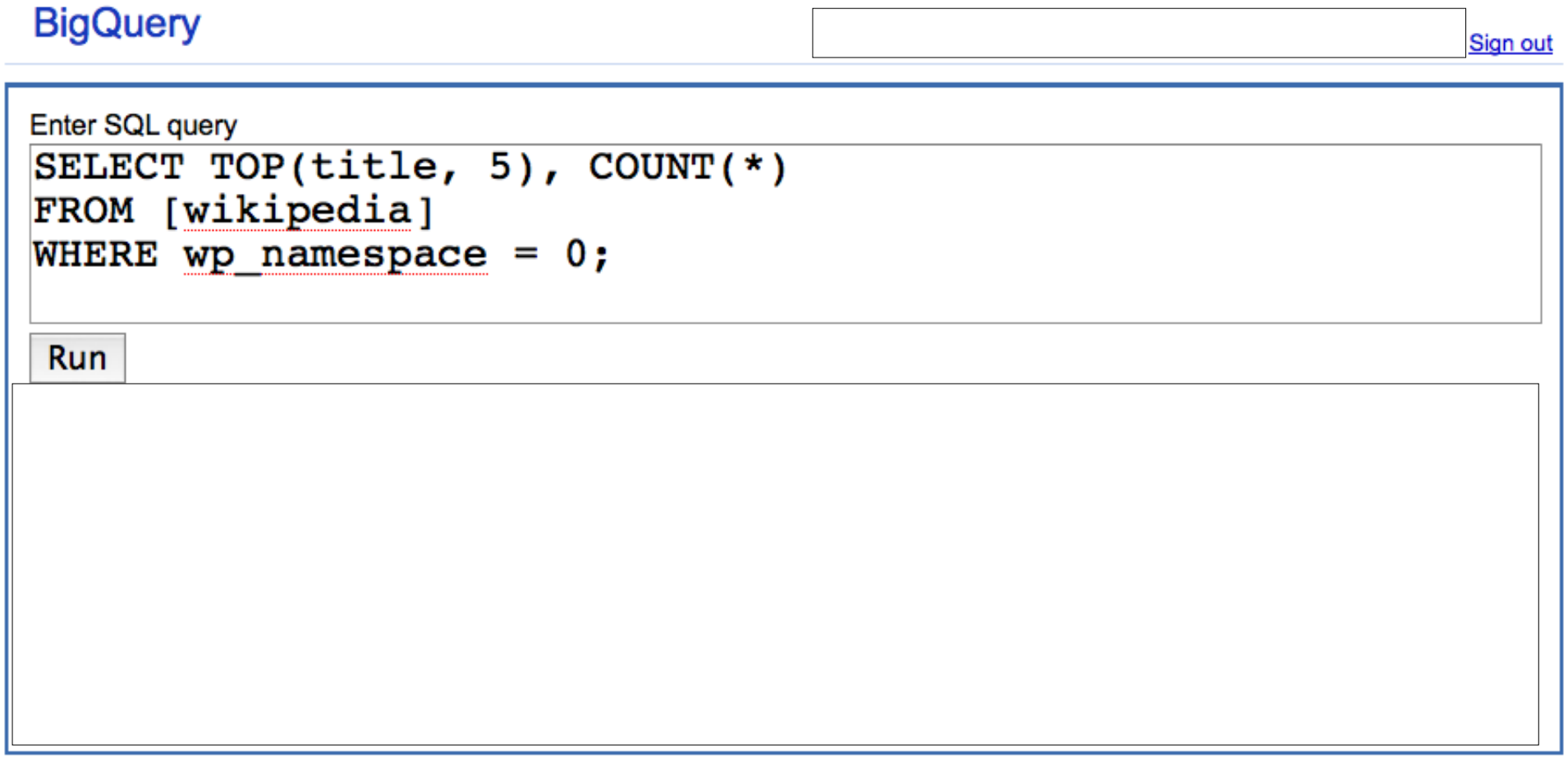
HTTPS support

- protects your credentials
- protects your data

Use Google Storage for Developers to manage access

Large Corpus Analysis

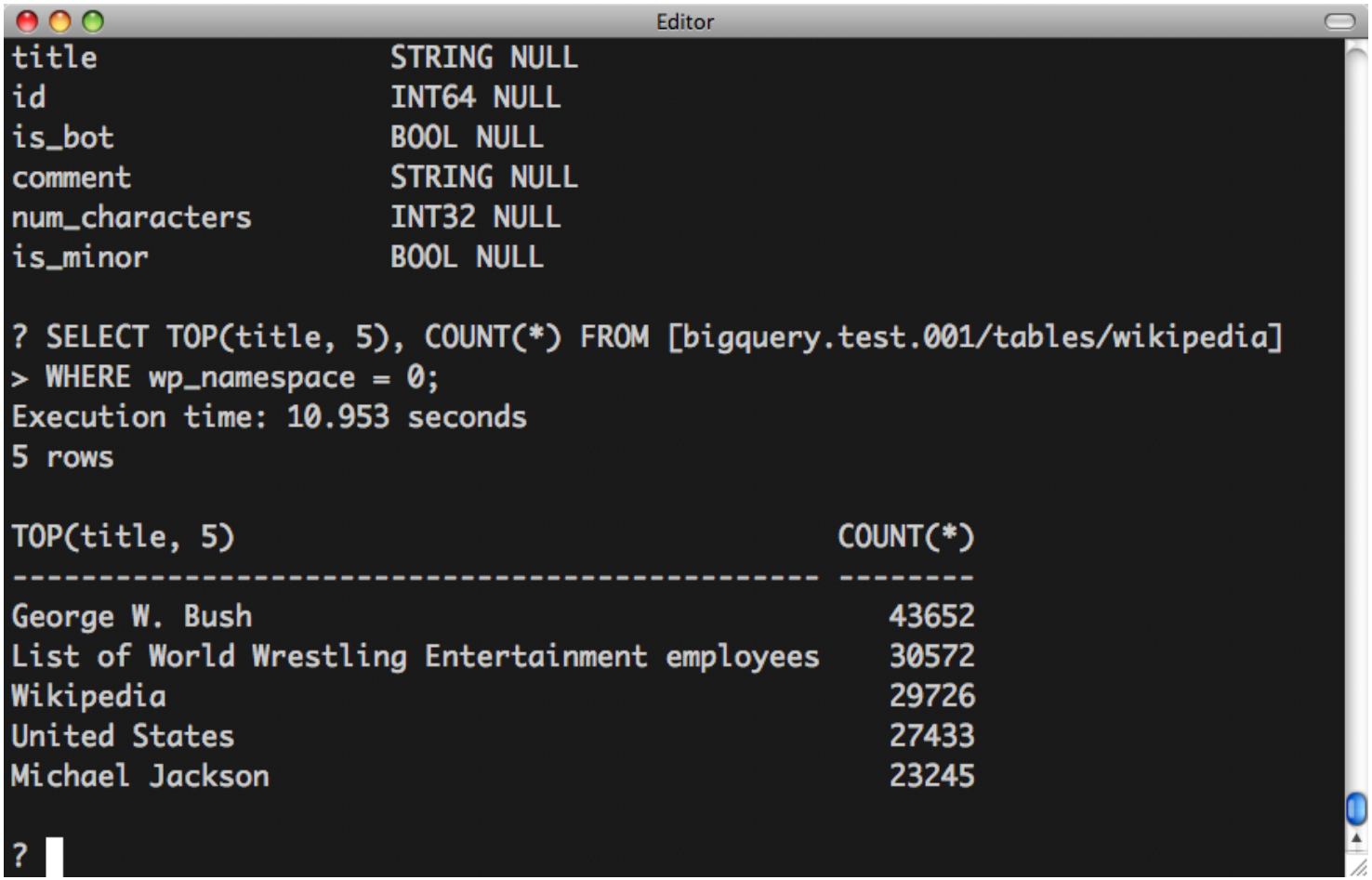
Wikimedia Revision History



Wikimedia Revision history data from: http://download.wikimedia.org/enwiki/latest/enwiki-latest-pages-meta-history.xml.7z

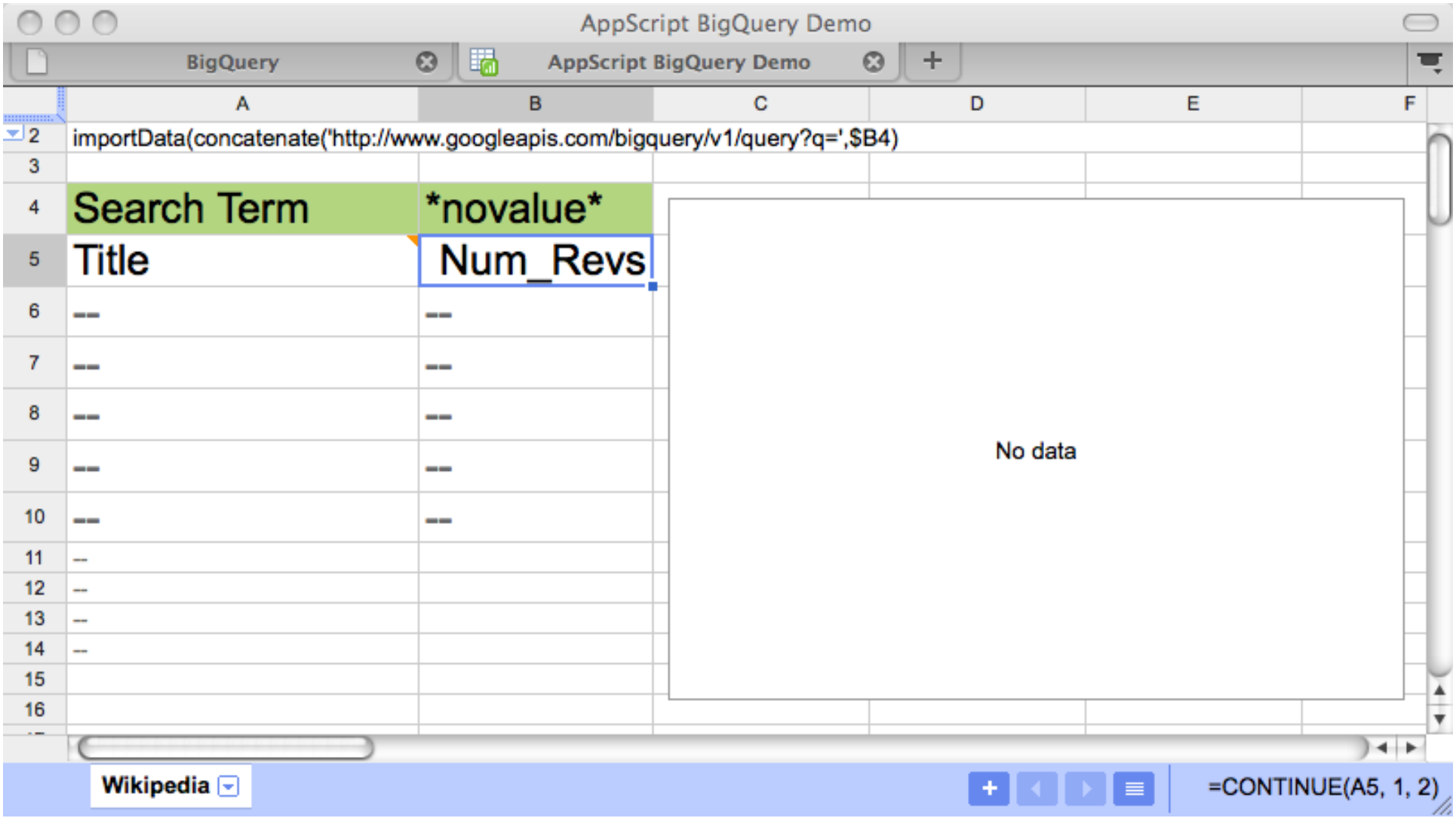
Using BigQuery Shell

Python DB API 2.0 + B. Clapper's **sqlcmd**
http://www.clapper.org/software/python/sqlcmd/



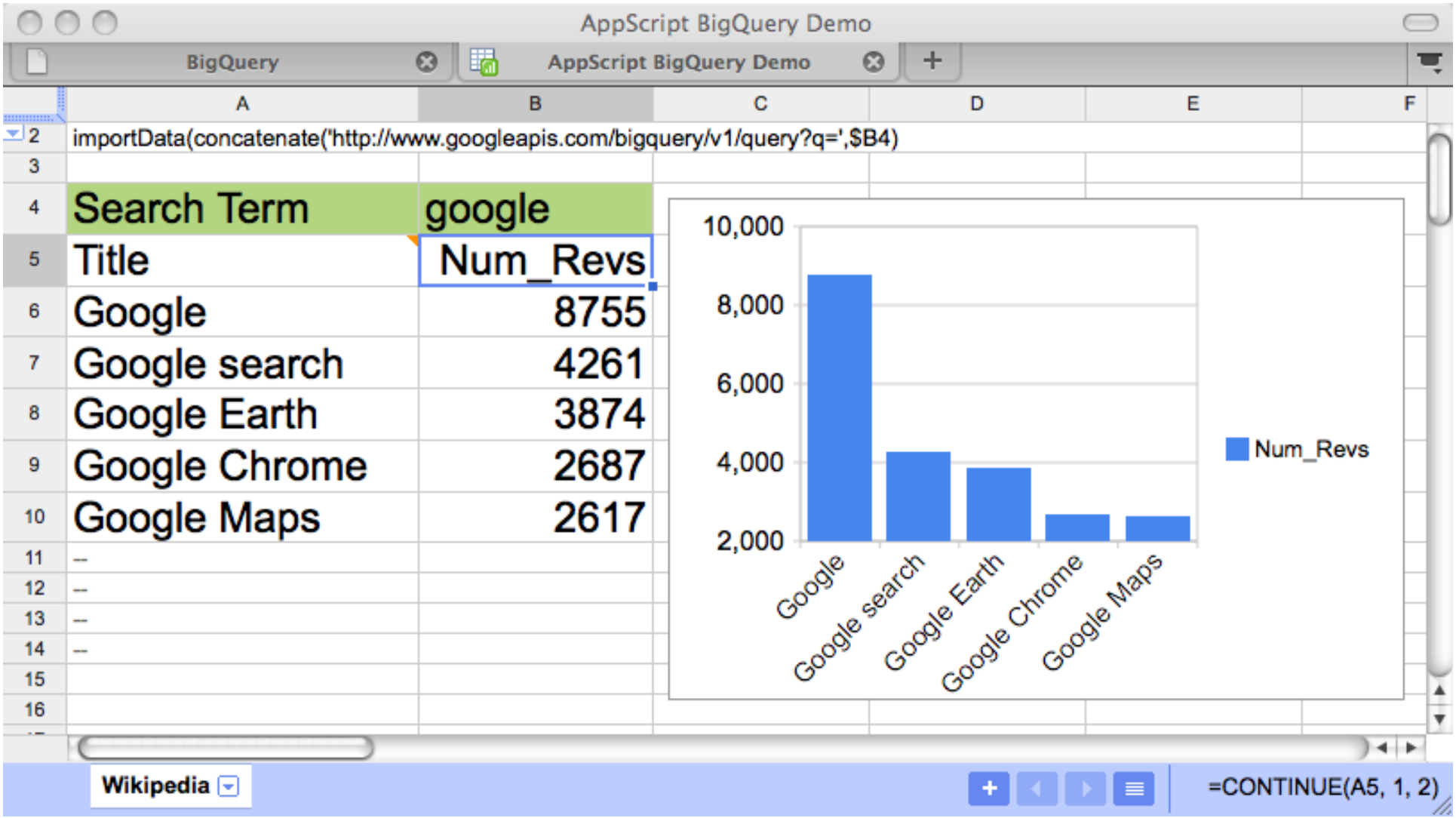
Google I/O 10

BigQuery from a Spreadsheet



Google I/O 10

BigQuery from a Spreadsheet



Google I/O 10

BigQuery Recap

- Interactive analysis of very large data sets
- Simple SQL query language

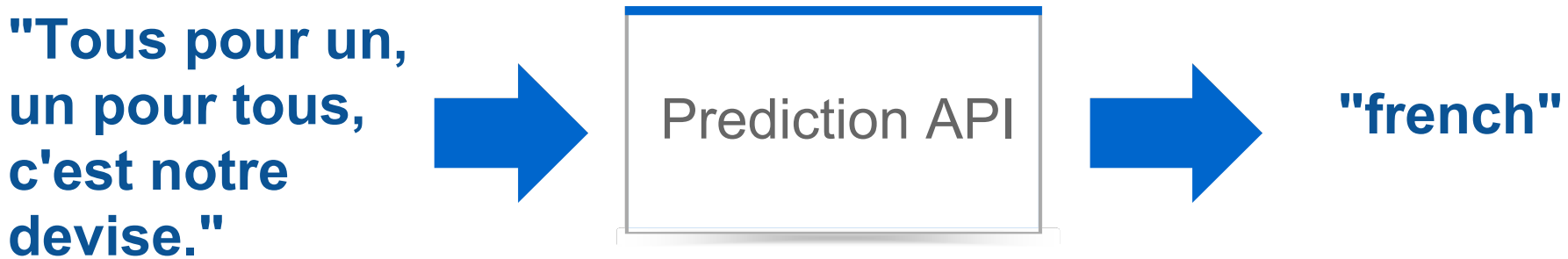
- APIs enable a variety of use cases

Prediction API
Machine learning as a web service

Max Lin, Gideon Mann
May 20th, 2010

Prediction API 101

- Google's sophisticated machine learning algorithms
- Available as an on-demand RESTful HTTP web service
- Train a model offline/asynchronously
- Predict results in real-time



How does it work?

The Prediction API finds relevant ***features*** in the sample data during training.

"english"	**The** quick brown fox jumped over **the** lazy dog.
"english"	**To** err **is** human, but **to** really foul things up you need a computer.
"spanish"	**No** hay mal **que por** bien **no** venga.
"spanish"	**La** tercera **es la** vencida.

The Prediction API later searches for those ***features*** during prediction.

?	**To** be or not **to** be, that **is the** question.
?	**La** fe mueve montañas.

A virtually endless number of applications...

Customer Sentiment

Transaction Risk

Species Identification

Message Routing

Diagnostics

Churn Prediction

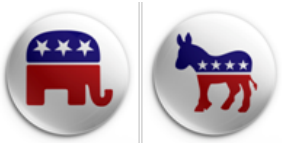
Legal Docket Classification

Suspicious Activity

Work Roster Assignment

Inappropriate Content

Recommend Products



Political Bias

Uplift Marketing

Email Filtering

Career Counselling

... and many more ...

Google I/O 10

Three simple steps to use the Prediction API

1. Upload	Upload your training data to Google Storage	Use the API, gsutil or any compatible utility to upload your data to Google Storage
2. Train	Build a model from your data	prediction/v1/train/{} POST : a training request
3. Predict	Make new predictions	prediction/v1/query/{} GET : model info POST : a prediction request

Prediction API Demo

Automatically categorize and respond to emails by language

- **Customer**: ACME Corp, a multinational organization
- **Goal**: Respond to customer emails in their language
- **Data**: Many emails, tagged with their languages
- **Outcome**: Predict language and respond accordingly

Step 1: Upload

Upload your training data to Google Storage

- Training data: **outputs** and **input features**
- Data format: comma separated value format (CSV)

```
$ head -n 2 ${data}
"english","To err is human, but to really ..."
"spanish","No hay mal que por bien no venga."
```

Upload to Google Storage

```
$ gsutil cp ${data} gs://io10/${data}
```

Step 2: Train

Create a new model by training on data

To train a model:

POST prediction/v1/train/${data}

Training runs asynchronously. To see if it has finished:

GET prediction/v1/query/${data}

```
{"data": {
   "resource": {
    "data": "${data}",
    "modelinfo": "estimated accuracy: ${acc}"}}}
```

Step 3: Predict

Apply the trained model to make predictions on new data

POST prediction/v1/query/${data}

```
{ data : {
  "instance" : {
    "input" : { "text" : [
      "J'aime X! C'est le meilleur" ]}}}}
```

Step 3: Predict

Apply the trained model to make predictions on new data

POST prediction/v1/query/${data}

```
{ data : {
  "instance" : {
    "input" : { "text" : [
      "J'aime X! C'est le meilleur" ]}
    "output" :
    {"output_label" : "french"}}}}
```

Step 3: Predict

Apply the trained model to make predictions on new data

```
import httplib

header = {"Content-Type" : "application/json"}#...put new data in JSON format in params variable
conn = httplib.HTTPConnection("www.googleapis.com")conn.request ("POST",
  "/prediction/v1/query/${data}",  params, header)print conn.getresponse()
```

Prediction API Capabilities

Data

- Input Features: numeric or unstructured text
- Output: up to 100s of discrete categories

Training

- Many machine learning techniques
- Automatically selected
- Performed asynchronously

Access from many platforms:

- Web app from Google App Engine
- Apps Script (e.g. from Google Spreadsheet)
- Desktop app

Get the BigQuery & Prediction APIs

- Preview, opened to a limited number of developers
- To request access and get more information, go to:
  - http://code.google.com/apis/bigquery
  - http://code.google.com/apis/prediction

View live notes and ask questions about this session on Google Wave
http://bit.ly/dr01ws


Google™
10