



Resume Tracking System

By
Rajan Patel

About project

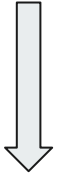


I have developed resume tracking system for recruiters, which keep track of large amount of applicants information and provided better, faster and efficient way to store and analyze informations using advance technology.

System Design



Web
scraping



Data Collection



Data Streaming



Data Storage



Data Visualization

Language used :- Python

Data Collection - Web Scraping

Reason to use web scraping.

- To get updated information.
- Open opportunity to get access large amount of data.
- Advantage to collect data from multiple platforms.

We choose indeed platform to collect resume information.

Technology used for web scraping:

- **Language:** Python
- **Packages:** beautifulsoup, selenium and threading



Software Developer
Software Developer

Guaynabo, PR
Software Developer, Web Developer, Mobile Developer

Work Experience

Software Developer
University of Puerto Rico
August 2015 to May 2016
Software and Database Development for the University Center for Psychological Services and Studies.

Software Developer
University of Puerto Rico
October 2015 to October 2015
Software Development Competition with universities from Puerto Rico and other countries.

Education

Bachelor's in Computer Sciences
University of Puerto Rico at Bayamón
May 2017

JSON

```
{  
  "id": "8ee28afc135d310b1",  
  "jobs": [  
    {"company": "University of Puerto Rico",  
      "hire_date": "August 2015",  
      "location": "",  
      "title": "Software Developer"},  
    {"company": "University of Puerto Rico",  
      "hire_date": "October 2015",  
      "location": "",  
      "title": "Software Developer"},  
  ],  
  "schools": [  
    {"degree": "Bachelor's in Computer  
Science",  
      "school_name": "University of Puerto  
Rico at bayamon",  
      "grad_date": "May 2017"},  
  ]  
}
```

Issue - data is not uniform.



California State University Long Beach

CSULB

CSU - Long Beach.

Cal State Long Beach

California State University, LB

Master in Computer Science

MS in Computer Science

Master's in CS.

Master's in Computer Science

Master of science in Computer Science.

MSCS

Method to make data uniform.

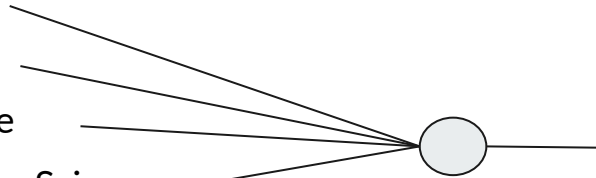
1. Replace mnemonic with actual words (using mnemonic data set).
2. Used “best string matching” algorithm to map all data to its respective category

MS in Computer Science

Master's in CS.

Master's in Computer Science

Master of science in Computer Science.



Master in Computer Science

- **Degree**

- **College name**

- **Company name**

- **Job Title**

```
[{ "id": "8ee28afc135d310b1",  
  "jobs": [{"company": "University of Puerto Rico",  
            "hire_date": "August 2015", "location": "",  
            "title": "Software Developer"}, {"company":  
            "University of Puerto Rico", "hire_date": "October  
            2015", "location": "", "title": "Software  
            Developer"},  
  ], "schools": [  
    {"degree": "Bachelor's in Computer  
    Science",  
     "school_name": "University of Puerto Rico  
    at bayamon",  
     "grad_date": "May 2017", }  
  ]  
}] ..... 1000+
```



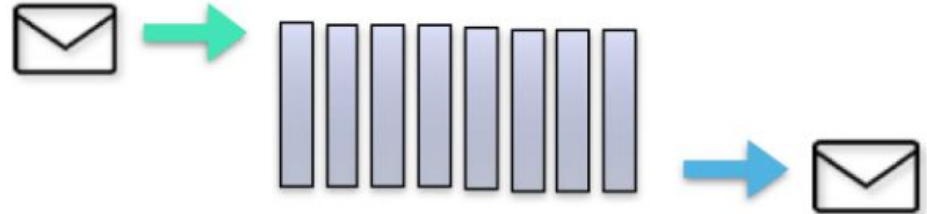
Data Streaming

Data Streaming



Apache Kafka is a distributed publish-subscribe messaging system that receives data from different source systems and makes the data available to target systems in real time.

- Acts as Safety buffer.
- Highly Scalable.





Kafka Topic

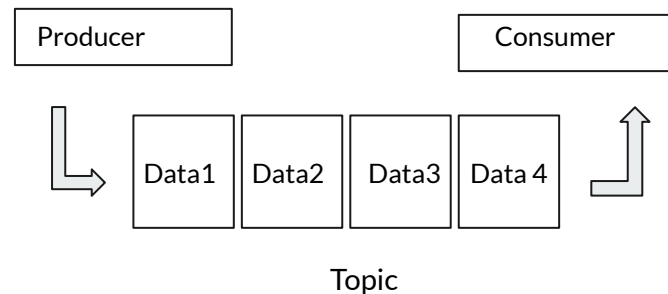
A list of data where producers add data to one end (back in this case) and consumers read from the other end

Kafka Producers

Producers are processes that publish data (push messages) into Kafka topics within the broker.

Kafka Consumers

A consumer of topics pulls messages off a Kafka topic.

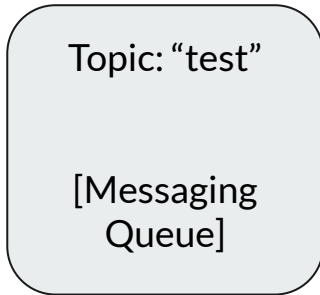


How did we use Kafka?



Producer

Uniform data stored in JSON format is pushed into the messaging queue by the Producer.



Consumer

Data is pulled from the messaging queue by the Consumer and sent to Data Storage.

Producer code.



```
from kafka import KafkaProducer
import json

producer = KafkaProducer(value_serializer=lambda v:
    json.dumps(v).encode('utf-8'),bootstrap_servers=['localhost:9092'])

producer.send('test', #data)
```

Consumer code.



```
from kafka import KafkaConsumer
import json

consumer = KafkaConsumer('test', group_id='my-group', bootstrap_servers=['localhost:9092'])
KafkaConsumer(auto_offset_reset='latest', value_deserializer=lambda m:
json.loads(m.decode('ascii'))))

for message in consumer:
    try:
        Message #(single resume data)
    except:
        print("Error")
        break
```

Single resume

```
{ "id": "8ee28afc135d310b1",  
  "jobs": [{"company": "University of Puerto Rico",  
            "hire_date": "August 2015", "location": "",  
            "title": "Software Developer"}, {"company":  
            "University of Puerto Rico", "hire_date": "October  
            2015", "location": "", "title": "Software  
            Developer"},  
  ], "schools": [  
    { "degree": "Bachelor's in Computer  
      Science",  
      "school_name": "University of Puerto Rico  
at bayamon",  
      "grad_date": "May 2017", }  
  ]  
}
```



Data Storage

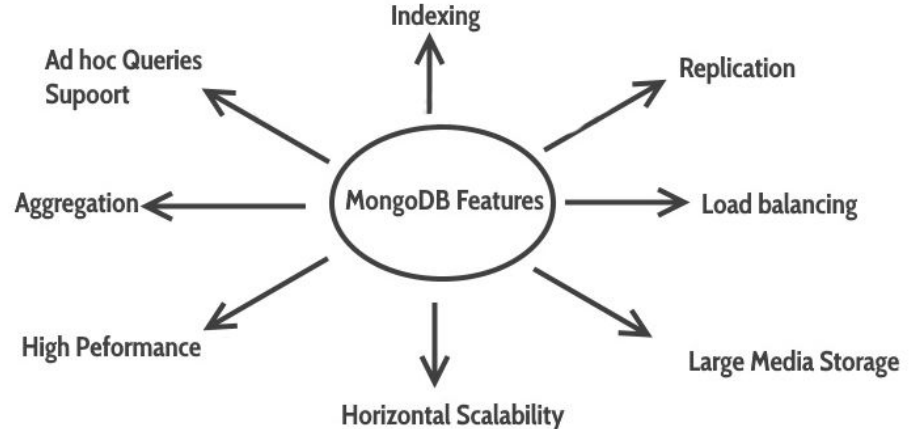
Data Storage



MongoDB is an open-source document-based database management tool that stores data in JSON-like formats.

Main Features :

- Highly scalable
- Faster Performance
- Flexible and distributed NoSQL database.



MongoDB - Implementation

```
from kafka import KafkaConsumer
import json
import pymongo
from pymongo import MongoClient

client = MongoClient()
db = client.project
collection = db.myCollection

consumer = KafkaConsumer('test', group_id='my-group', bootstrap_servers=['localhost:9092'])
KafkaConsumer(auto_offset_reset='latest', value_deserializer=lambda m: json.loads(m.decode('ascii')))

for message in consumer:
    try:
        post_id = collection.insert_one(json.loads(message.value)).inserted_id
    except pymongo.errors.DuplicateKeyError:
        pass
    except:
        print("Error")
        break
```


All data..

```
[{ "id": "8ee28afc135d310b1", .....}  
[{"id": "2ee28afc135d310d1", .....}  
[{"id": "3ee28afc135d310g1", .....}  
[{"id": "4ee28afc135d310b2", .....}  
[{"id": "6ee28afc135d310b3", .....}  
[{"id": "6ee28afc135d310b4", .....}  
[{"id": "3ee28afc135d310b5", .....}  
[{"id": "4ee28afc135d310b6", .....}  
[{"id": "6ee28afc135d310b6", .....}  
[{"id": "6ee28afc135d310b7", .....}
```

.....
.
.
.

Live update



Data Visualization

Data visualization



We used Tableau.

- Powerful.
- Faster growing data visualization tool.
- Used widely in business intelligence industry.

We connected tableau to MongoDB using BI connector.

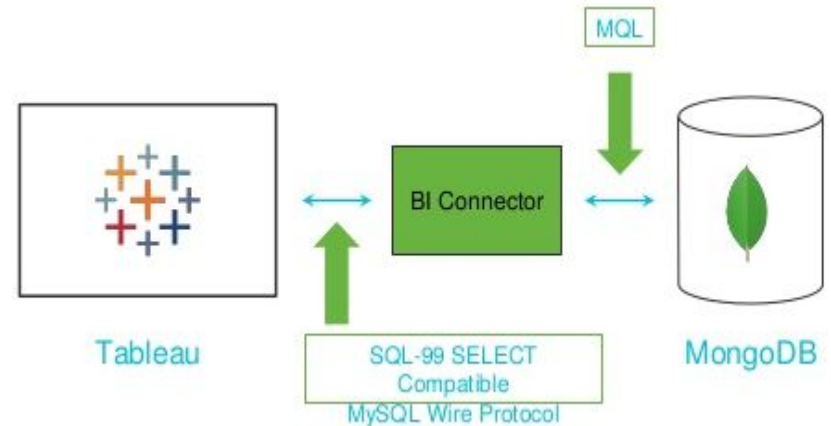


Tableau data import interface

← → 📄 ↻

Connections [Add](#)

localhost
MongoDB BI Connector

Database

project

Table 🔍

- myCollection
- myCollection_jobs
- myCollection_schools
- New Custom SQL
- New Union

myCollection_jobs (project)

Connection: Live Extract

Filters: 0 | [Add](#)

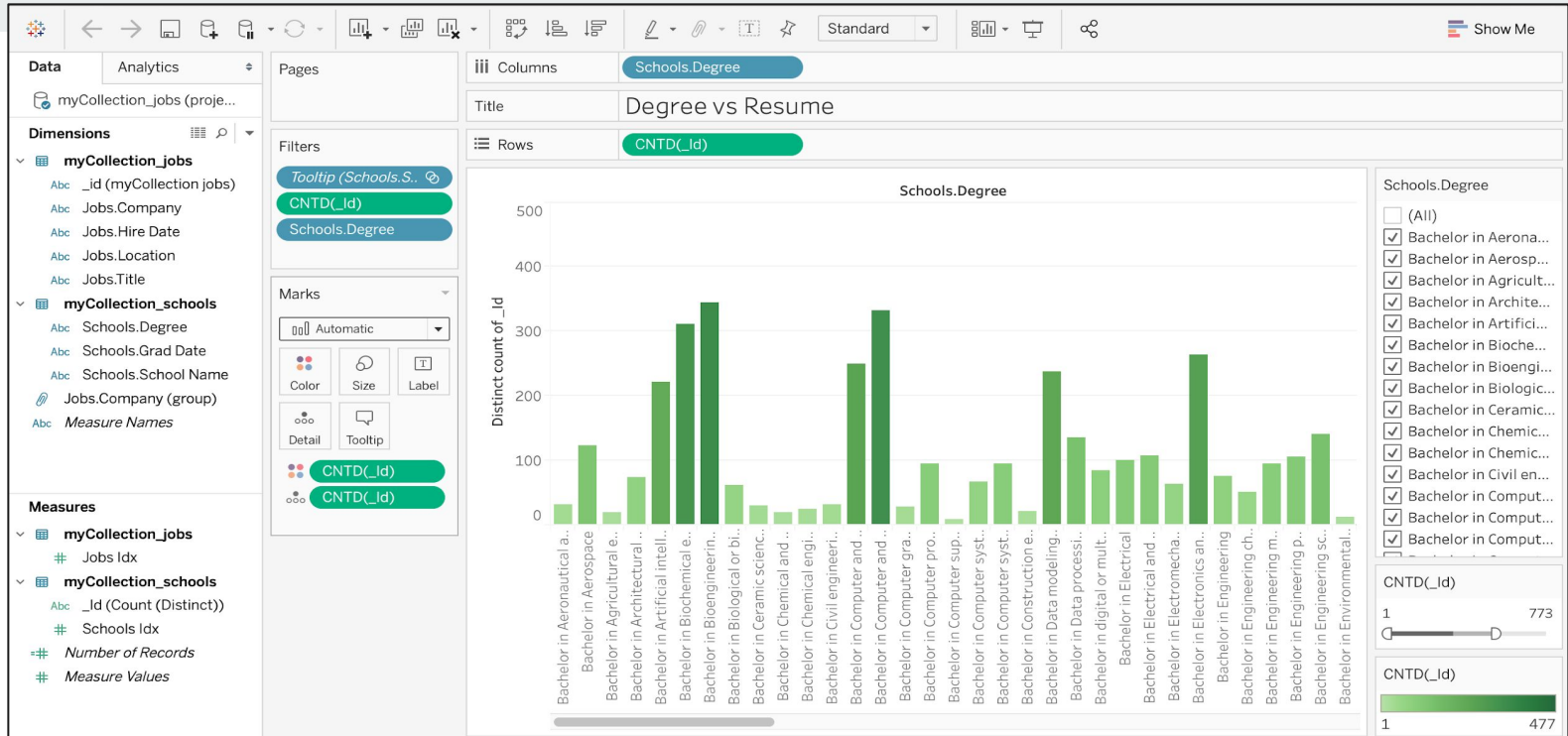
myCollection_schools myCollection_jobs

Sort fields: Data source order

Show aliases Show hidden fields 49,432 rows

myCollection_jobs _id (myCollection j...	myCollection_jobs Jobs.Company	myCollection_jobs Jobs.Hire Date	myCollection_jobs Jobs.Location	myCollection_jobs Jobs.Title	myCollection_jobs Jobs Idx	myCollection_schools _Id	myCollec School
78aac576b4b151da	Cambridge Semantics			Software Engineer	0	78aac576b4b151da	Bachel
77d50820b6c23ffc	Tableau Software	2016		Software Engineer	0	77d50820b6c23ffc	Bachel
77d50820b6c23ffc	Alarm.com	2017		Software Developer	1	77d50820b6c23ffc	Bachel
77d50820b6c23ffc	SocialEffort Inc	2009		R&D Software develo...	2	77d50820b6c23ffc	Bachel
77d50820b6c23ffc	Morgan Stanley	2015		Software Developer	3	77d50820b6c23ffc	Bachel
77d50820b6c23ffc	NonprofitMetrics		Modesto, CA	Software Engineer	4	77d50820b6c23ffc	Bachel
217fdbac1df6737	Harris Corporation	2018	Santa Clara, CA	Software Engineer	0	217fdbac1df6737	Maste
217fdbac1df6737	Harris Corporation	2018	Santa Clara, CA	Software Engineer	0	217fdbac1df6737	Bachel
b6a4e9496fce140b	Galorath Incorporated	2012	Laguna Beach, CA	Software Engineer	0	b6a4e9496fce140b	Bachel

Tableau sheets.



In the above sheet we are generating the information about the count of resumes filtered out on the basis of their degree earned.

We can also use the filter to choose count of any particular degree earned.

Demo



<https://public.tableau.com/profile/suraj.nair1535#!/vizhome/Resumeactivitytracker/Dashboard1?publish=yes>

Benefit over existing systems



- Resume information is a semi-structured data.
- Kafka is used here as a safety buffer.
- MongoDB provides the best of NoSQL DB.
- Using Tableau users can visually interact with data to get insights faster, and make critical decisions.