Yelp | Similar Biz Recommender

Metis Data Engineering Project Proposal

Darren K. Lee

Data Engineering: Project Proposal

Hypothesis: *Can we estimate salary for given job posting?*

- **Data Ingestion and Storage:**
  - Download ~10gb of json files from Yelp Dataset, as warm start..
  - Store in noSQL database via MongoDB (locally to test, then stream to GCP service / BigQuery if possible)
- **Processing:**
  - Build a model via python web app (Google Colab or AWS Sagemaker - Jupyter Notebook)
  - Also try to deploy Docker Container for ease of version control
  - Build processing framework so that we can retrain/automate using the same or new features
- **Deployment:**
  - Deploy a pre-trained model for web app
  - Create a dashboard via *StreamLit* or Looker and demonstrate regression task accuracy and all other EDA

Data Pipeline | Yelp Similarity Recommender

DATA INGESTION

Web Download

Warm start with Yelp's dataset. Data will ideally come in a JSON structure, which can be ingested into a noSQL database

UNIT TESTING

`**import schedule**` command to ingest new data, and trigger new pipeline for ML task. **MVP will ONLY work with pre-trained businesses, will look into creating a way to pre-process ad-hoc payload to generate inference from model file

DATA STORAGE

MongoDB (noSQL)
Sample data in Jupyter Notebook via MongoClient

PROCESSING

AWS / GCP

~~AWS Sagemaker /Google Colab~~
Python web app instead
Aggregate and EDA using Pandas
Cosine Similarity and CountVectorizer will be used on Categorical data

DEPLOYMENT

Streamlit

Build a dynamic dataframe and also fine tune model parameters.
Using Streamlit, demonstrate how to adjust parameters to a given model and see how output changes

The Dataset

{'_id': ObjectId('622df55a6737c2d75b8d0585'),
'business_id': 'Pns2l4eNsfO8kk83dixA6A',
'name': 'Abby Rappoport, LAC, CMQ',
'address': '1616 Chapala St, Ste 2',
'city': 'Santa Barbara',
'state': 'CA',
'postal_code': '93101',
'latitude': 34.4266787,
'longitude': -119.7111968,
'stars': 5.0,
'review_count': 7,
'is_open': 0,
'attributes': {'ByAppointmentOnly': 'True'},
'categories': 'Doctors, Traditional Chinese Medicine, Naturopathic/Holistic, Acupuncture, Health & Medical, Nutritionists',
'hours': None}

Data is unstructured where there is nested meta data in the `category` and `attributes` field, which is where we will focus on per similarity. One unique document per business.

- Loaded ~4gb of data into MongoDB as local db (~150k documents)
- Using 100k records for modeling
- **TO-DO's:**
  - Will try loading in more tables in the next iteration
  - **MVP will ONLY work with pre-trained businesses, will look into creating a way to pre-process ad-hoc payload to generate inference from model file

is_open	Count	Notes
0 [NO]	20,380	Will use as payload**
1 [YES]	79,620	

The Model

Beanstalk Kids (is_open == 0 😢😪)

List of Categories:
['Used',
' Vintage & Consignment',
' Baby Gear & Furniture',
' Shopping',
' Maternity Wear',
' Musical Instruments & Teachers',
' Fashion']

Since they are closed, find me similar businesses to Beanstalk Kids.

The model will recommend Once Upon a Child!!

Count Vectorizer
For list of categories Across entire corpus *(list of lists)*

Cosine Similarity
Take Cos Sim of CountVectorizer object as an array()

Numpy Pickle
Export Numpy object as Model File for Deployment

Once Upon a Child (is_open == 1 😀😊)

['Used',
' Vintage & Consignment',
' Shopping',
' Baby Gear & Furniture',
' Fashion']

MVP (so far..)

jupyter Project_Proposal_Yelp_Dataset_Challenge Last Checkpoint: an hour ago (autosaved)

File Edit View Insert Cell Kernel Widgets Help

Not Trusted | Python 3 (ipykernel)

In [57]:

```
print (df.iloc[INDEX]['name'], ": ", df.iloc[INDEX]['categories'], df.iloc[INDEX]['attributes'])
print ()
for i in closest_index:
    if df.iloc[i]['is_open'] == 1:
        print (i)
        print (df.iloc[i]['name'], ": ", df.iloc[i]['categories'], df.iloc[i]['attributes'])
        print ('Cosine Similarity Score:', cosine_similarity(dist[INDEX].reshape(1,-1), dist[i].reshape(1,-1))[0][0
        print()
```

```
Beanstalk Kids :  ['Used', ' Vintage & Consignment', ' Baby Gear & Furniture', ' Shopping', ' Maternity Wear', ' Musi
cal Instruments & Teachers', ' Fashion'] {'BusinessParking': "{'garage': False, 'street': False, 'validated': False,
'lot': False, 'valet': False}", 'BusinessAcceptsCreditCards': 'True', 'RestaurantsPriceRange2': '4', 'WheelchairAcces
sible': 'True', 'BikeParking': 'True'}

55896
Once Upon a Child :  ['Used', ' Vintage & Consignment', ' Shopping', ' Baby Gear & Furniture', ' Fashion'] {'Business
AcceptsCreditCards': 'True', 'BusinessParking': "{'garage': False, 'street': False, 'validated': False, 'lot': True,
'valet': False}", 'WheelchairAccessible': 'True', 'BikeParking': 'False', 'RestaurantsPriceRange2': '2'}
Cosine Similarity Score: 0.9958262

6682
Goodwill Industries :  ['Used', ' Vintage & Consignment', ' Shopping', ' Fashion'] {'BusinessParking': "{'garage': Fa
lse, 'street': False, 'validated': False, 'lot': True, 'valet': False}", 'BusinessAcceptsCreditCards': 'True', 'BikeP
arking': 'True', 'RestaurantsPriceRange2': '1'}
Cosine Similarity Score: 0.99251384

20567
Donna's Consignment Boutique Consignment :  ['Used', ' Vintage & Consignment', ' Shopping', ' Fashion'] {'BusinessAcc
```

In [ ]:

Streamlit App Flow - Yelp Similar Businesses

Given Yelp `*categories*`, and `*attributes*`, can we provide similar businesses? Payloads will be picked from a business that is currently not open. This will key off the pipeline to suggest a similar business, and resort the list. Will use the current numpy pickle file in a Python web app to run Streamlit for the demonstration

Additional Top Level Filters to get better results:

Demonstrate these knobs in Streamlit app.

- Geographical (using lat long fields or city, state)
- Star Rating
- “Must have” category (e.i. Must have “DogsAllowed”)