RECOMMENDATION SYSTEM USING YELP DATASET

ABHAY KASTURIA
ADITYA PRIYADARSHI
GAUTAM VASHISHT
VARUN NANDU
XINGXING LIU

Outline

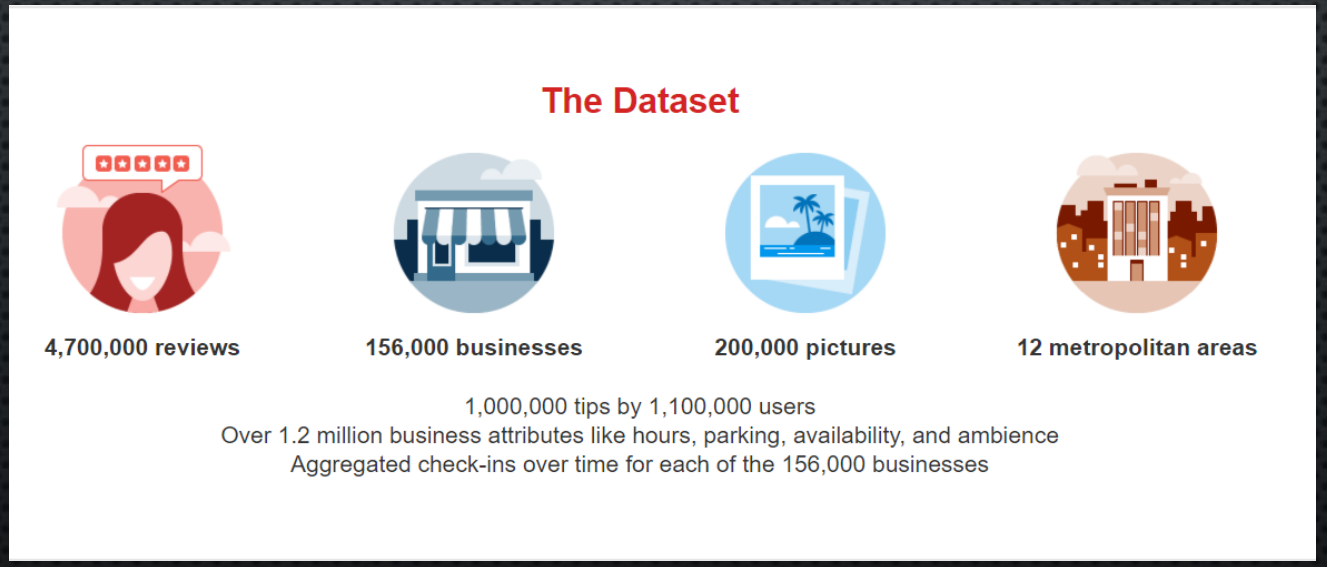
- Introduction
- Data Preprocessing
- Methods
- Evaluation and results

INTRODUCTION

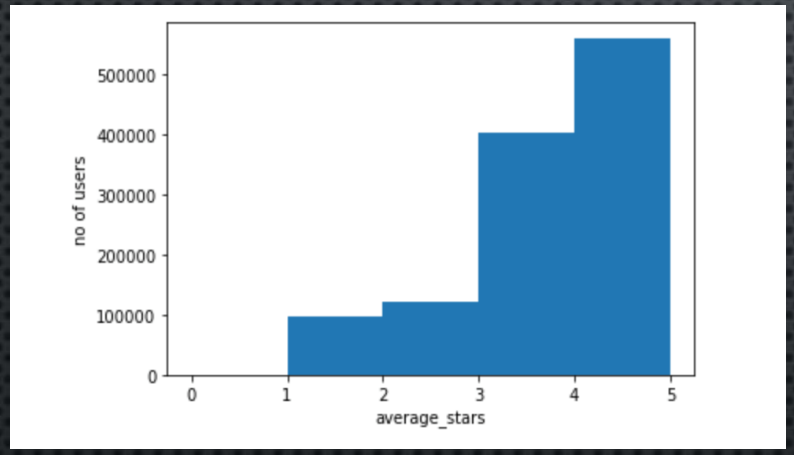
- RECOMMENDATION SYSTEM
- WHAT? WHY?

PROJECT

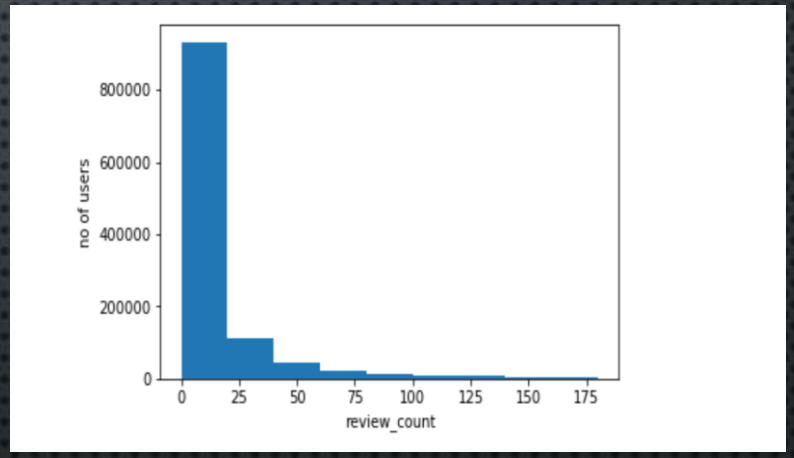
- USE YELP DATASET TO RECOMMEND RELEVANT BUSINESSES TO USERS
- BASED ON THE ACTIVITY OF USERS ON YELP
- ACTIVTY = REVIEWS GIVEN BY USER TO BUSINESS
- EVALUATE SEVERAL DATA MINING METHODOLOGIES USED FOR RECOMMENDATIONS



BASIC DATA ANALYSIS

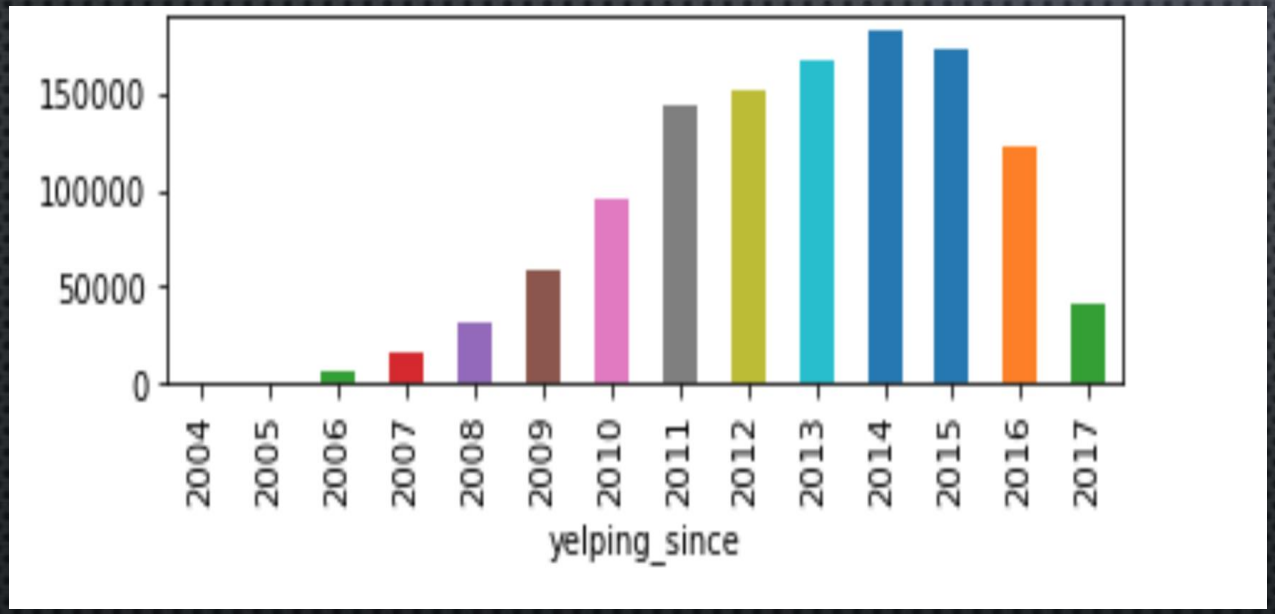


Average Star Ratings

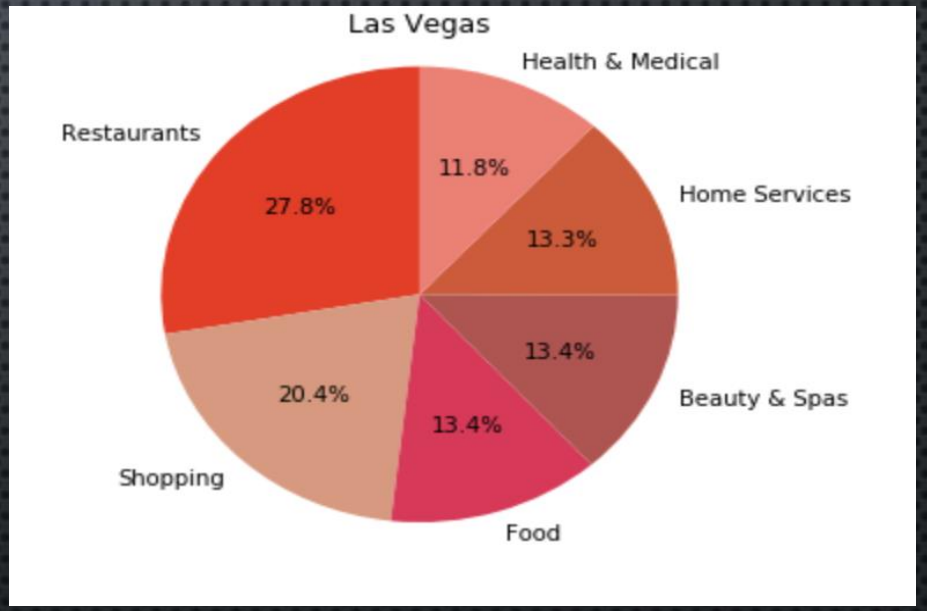


Average #Reviews

BASIC DATA ANALYSIS



User Growth



Popular Categories in Las Vegas

Data Preprocessing

- Data Filtering:
  - Users with reviews >= 20 ~20K records
  - Business with Category = Restaurant ~ 48K records
- Data Transformation:
  - UserId and BusinessId mapped to serial numbers
- Data Division:
  - Based on time
  - Training data – Reviews till 2015 ~ 900K
  - Testing data – Reviews after 2015 ~ 19K
  - Testing data review categorization - Positive(>3.7), Negative(<2.6), Neutral(rest)

Methods

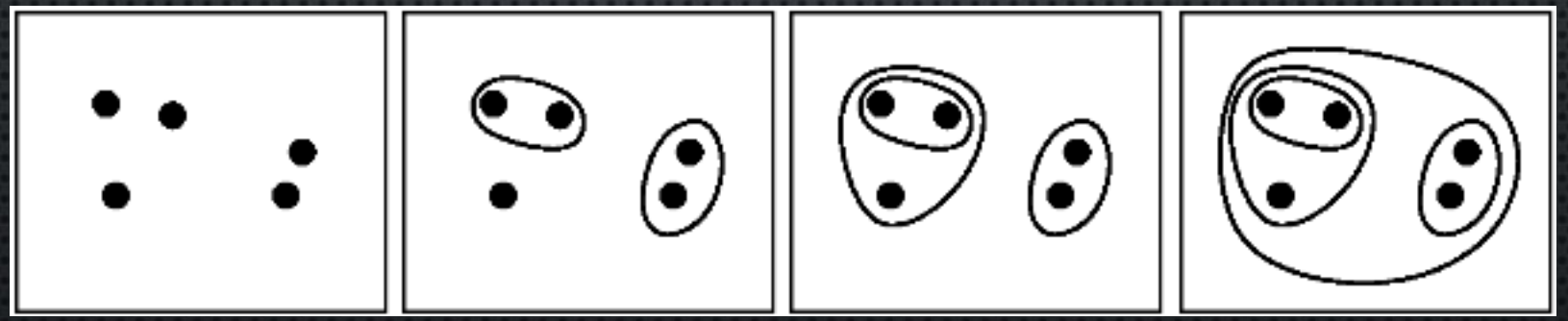
- Clustering
  - Agglomerative
- Collaborative filtering
  - User – User similarity based
  - Matrix factorization
- Deep Learning

Clustering - Approach

- Assumptions – UserID, Location and Type Of Business
- User preference vector of top 15 categories based on rating – what if no rating?
- Business vector – must contain 3 of top 15 categories.
- Use agglomerative Hierarchical clustering to cluster users

AGGLOMERATIVE HIERARCHICAL CLUSTERING

Number Of Clusters?



Recommend businesses

- Calculate average preference vector
- Calculate dot product with each business vector
- Sort the businesses based on decreasing value of the above dot product
- Take out first 20 businesses as your top recommendations

CLUSTERING - RESULTS

- HIT RATIO
  - POSITIVE(>3.7), NEGATIVE(<2.6), NEUTRAL(REST)
- NORMALIZED DISCOUNTED CUMULATIVE GAIN (NDCG)

City	Hit Ratio	NDCG
Las Vegas	0.58	0.78
Toronto	0.61	0.75
Phoenix	0.62	0.80
Charlotte	0.55	0.77
Edinburgh	0.60	0.81
Pittsburgh	0.44	0.78
Montral	0.57	0.80
Scottsdale	0.69	0.91
Cleveland	0.64	0.89
Madison	0.64	0.78
Stuttgart	0.5	0.71
Mesa	0.41	0.52
Tempe	0.35	0.69
Henderson	0.65	0.88
Mississauga	0.22	0.63
Average	0.55	0.78

User-User based collaborative filtering

- Created Utility matrix for users vs business filled with rating
- Centralized each rating vector of user to mean zero
- Replace the missing values with zero
- Averaged rating value would be 0, positive or negative where 0 represents mean rating or missing ratings and positive represents higher ratings than mean rating

RATING PREDICTION

$$predr_{(i,k)} = \frac{\sum sim(u_i, u_j) * r_{(j,k)}}{\#users(j)}$$

- PREDICTION FOR USER I FOR BUSINESS K IS GIVEN BY THE FORMULA ABOVE.
- PREDICTED THE RATINGS THAT USER WILL GIVE TO THE NEW BUSINESS
- FIND TOP K SIMILAR USERS WHICH HAVE ALREADY RATED THE ITEM 'I' AND RETURNED THE WEIGHTED AVERAGED OF RATINGS WITH WEIGHT AS SIMILARITY

Recommending top 10 business to user

- Predict rating for business for the user which one have not used yet
- Sort the array on basis of rating
- Return the top 10 business

COLLABORATIVE FILTERING MATRIX FACTORIZATION USING ALS

$$predr_{ui} = q_i^T . p_u$$

- Q – VECTOR OF BUSINESS AND P- VECTOR OF USERS
- PREDICTED THE RATINGS BASED ON THE DOT PRODUCT OF THE USER AND THE BUSINESS VECTOR

$$min_{q,p} \sum_{(u,i)} (r_{ui} - q_i^T p_u)^2 + \lambda(\| q_i \|^2 + \| p_u \|^2)$$

- BOTH Q AND P ARE UNKNOWNS, EQUATION IS NOT CONVEX.
- ALS FIXES ONE AND OPTIMIZES THE OTHER AND ROTATE BETWEEN FIXING THE Q AND FIXING THE P. THUS, THE NAME – ALTERNATING LEAST SQUARES.
- THIS ENSURES THAT EACH STEP DECREASES EQUATION UNTIL CONVERGENCE.

Collaborative Filtering
Matrix Factorization using ALS

$$min_{q,p} \sum_{(u,i)} (r_{ui} - q_i^T p_u)^2 + \lambda(\| q_i \|^2 + \| p_u \|^2)$$

- Parameters for ALS:
  - Lambda - Regularization Parameter
  - Iterations – Max If Convergence not reached
  - Rank - the number of features to use (aka the number of latent factors).

- Parameters for our ALS model:
  - Lambda = 0.15
  - Iterations = 20 (Slow computers)
  - Rank = 100 (Low Memory Computers)

COLLABORATIVE FILTERING MATRIX FACTORIZATION USING ALS

$$min_{q,p} \sum_{(u,i)} (r_{ui} - q_i^T p_u)^2 + \lambda(\| q_i \|^2 + \| p_u \|^2)$$

- RESULTS:

	MSE - Training	MSE - Testing
Non-Reg	0.02	2.2
Reg	0.04	1.4

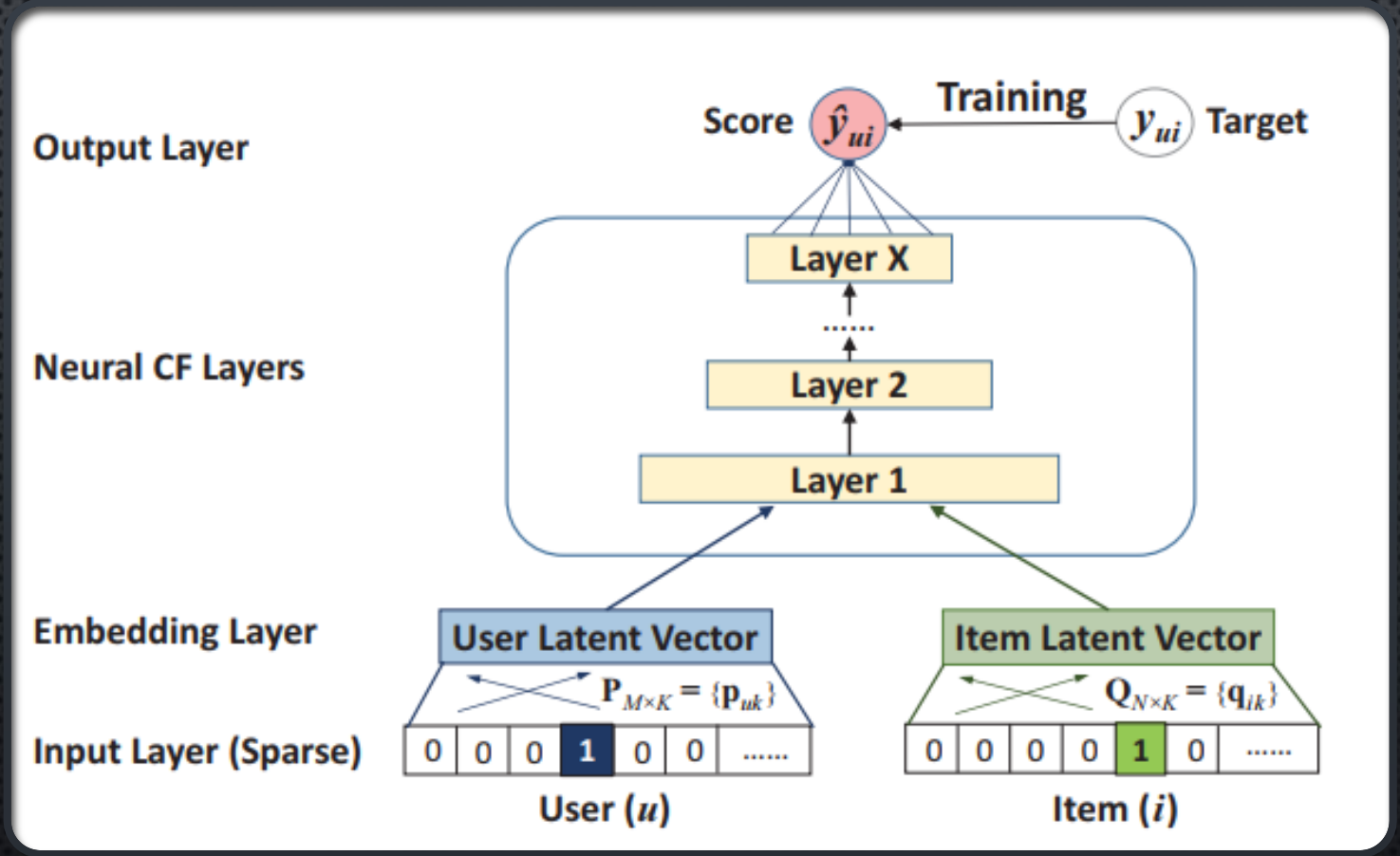
	Hit Ratio(Threshold = 1.5)
Reg	0.77

DEEP LEARNING - APPROACH

- Uses multi-layer perceptron (MLP) to learn user-item interaction
- Uses implicit feedback rather than explicit reviews

$$y_{ui} = \begin{cases} 1, & \text{if interaction (user u, item i) is observed} \\ 0, & otherwise \end{cases}$$

DEEP LEARNING - ARCHITECTURE



Deep Learning - Results

- Trained model using 4 hidden layers with [64,32,16,8] factors.
- Used hit ratio and ndcg to measure quality
- Best model – hit ratio (0.8197) and NDCG (0.6220)

CONCLUSION

	Hit Ratio
CF - ALS	0.77
DNN	0.81
Clustering(avg across cities)	0.55

- WE SEE THAT DNN GIVES US THE BEST HIT RATIO

Future works

- Utilize textual and image based data to help improve the recommendations.
- Utilize hybrid recommendation system techniques to use user behaviour and user characteristics(demographics of users) to improve recommendations.