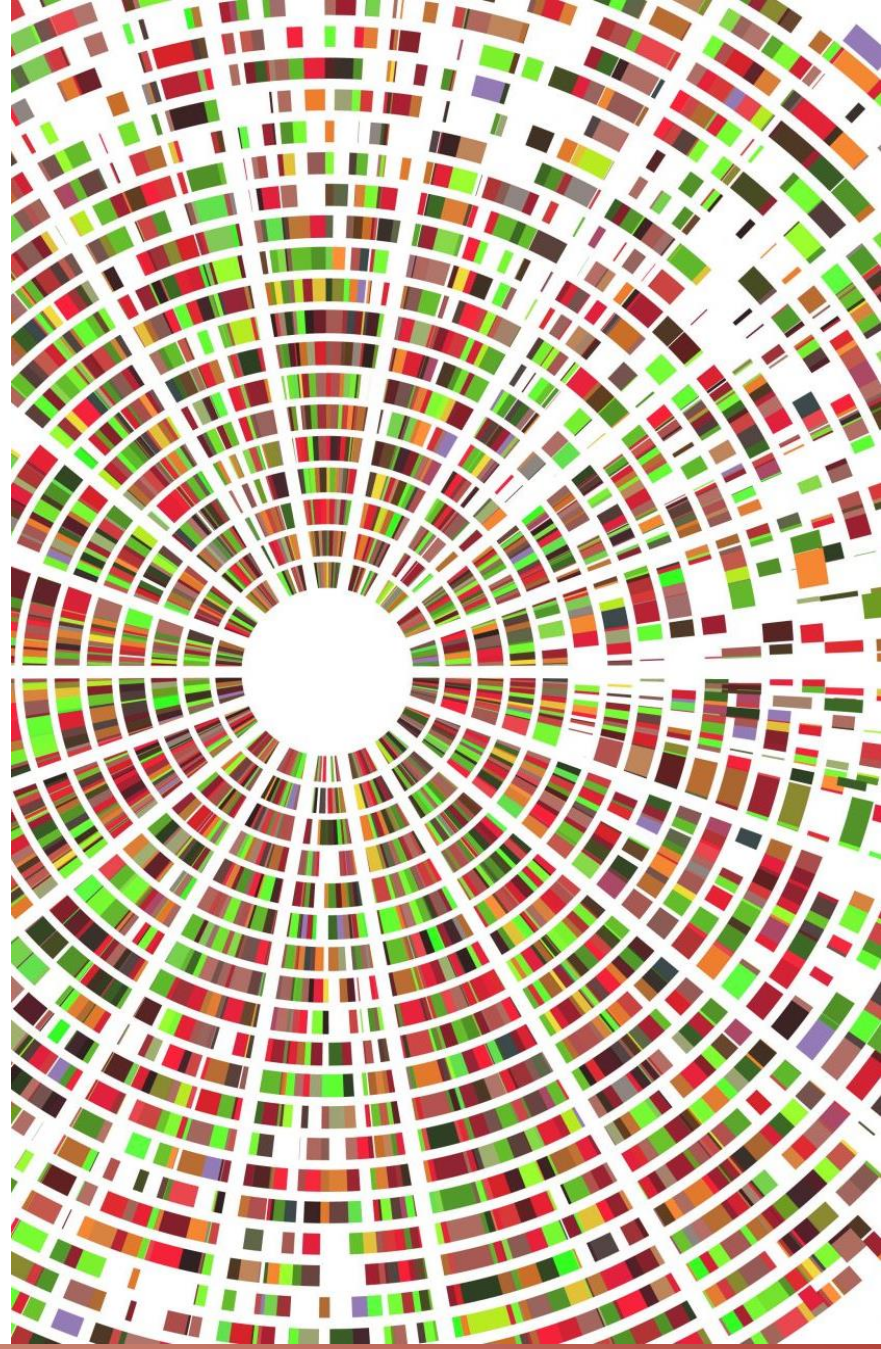
IDS 561- FINAL PROJECT

AMAZON RECOMMENDATION SYSTEM

SUBMITTED BY,

- ARCHANA SINGH
- CHAITRA SRIRAMA
- NIKITA BAWANE



PROBLEM STATEMENT

- ✓ While shopping online, **consumers** crave for firsthand info about **product review and experience**.
- ✓ Whereas **Ecommerce companies** like Amazon use this data to help them **increase their average order value** through their recommendation systems
- ✓ Our focus is to **build a recommendation system** by using product and review rating information for **Amazon - Health and Personal Care** related products by using the **techniques of collaborative filtering (ALS)**.

DATA

The data was obtained from the *UCSD repository* that contains datasets used for research in the domain of recommender systems.

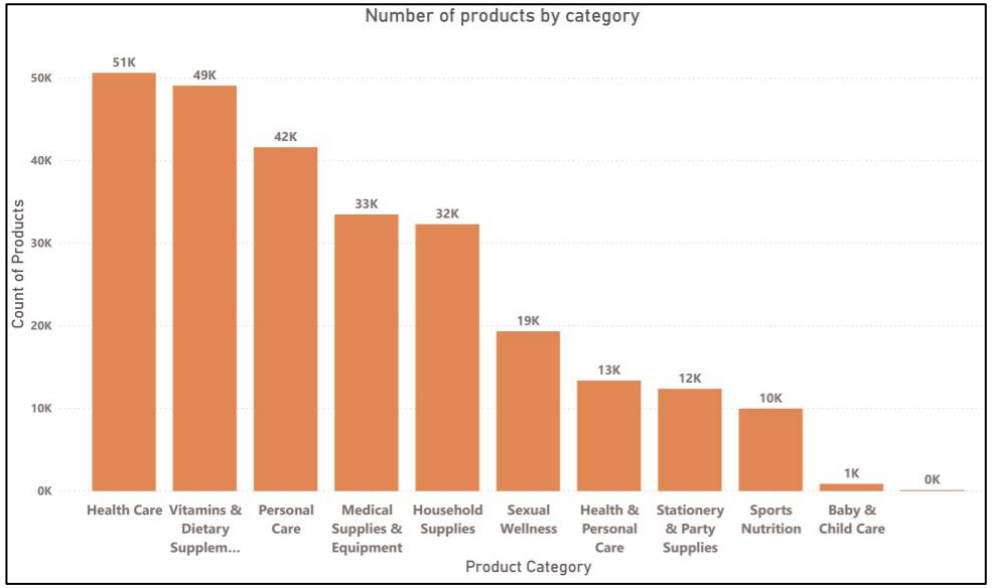
Link: **https://cseweb.ucsd.edu/~jmcauley/datasets.html**

Meta Data

- •The meta data contains data about the product information.
- •Column Names : product id (asin),title, price, imUrl, related, salesRank, category, brand
- •file type - json.gz (337 MB)
- •There were 263032 unique products.

Ratings

- •This file only contains data regarding user, product and ratings.
- •Column Names : user,item,rating, timestamp. (file type - csv)
- •Approx 3 Million ratings.
- •Aprrox 1.8 Million unique users



TECHNOLOGY USED

Database: **Mongo DB**

- Document oriented database – stores data from **JSON and CSV** file.
- Database is **setup on AWS** – up to *512 MB free space* provided for each account.
- **Mongo Atlas, a DBaaS**, lets us setup cloud database to store files.
- **Mongo Compass** provides a **GUI** to manage all MongoDB actions.
- **Why?**
  - Metadata file is highly unstructured.
  - MongoDB can be hosted on cloud.

Back End: **Pyspark and PyMongo**

- **PyMongo, driver in Python** to access MongoDB, is used to *setup connection and pull data from MongoDB*.
- **We build the recommendation system using Pyspark**.
- Used ALS algorithm to build the recommendation system

- **Why?**
  - Pyspark allows us to handle huge amount of data.
  - PyMongo driver allows us to connect with MongoDB.

Front End: **Power BI**

- Power BI is used for **data exploration and to visualize results** from the recommendation system.

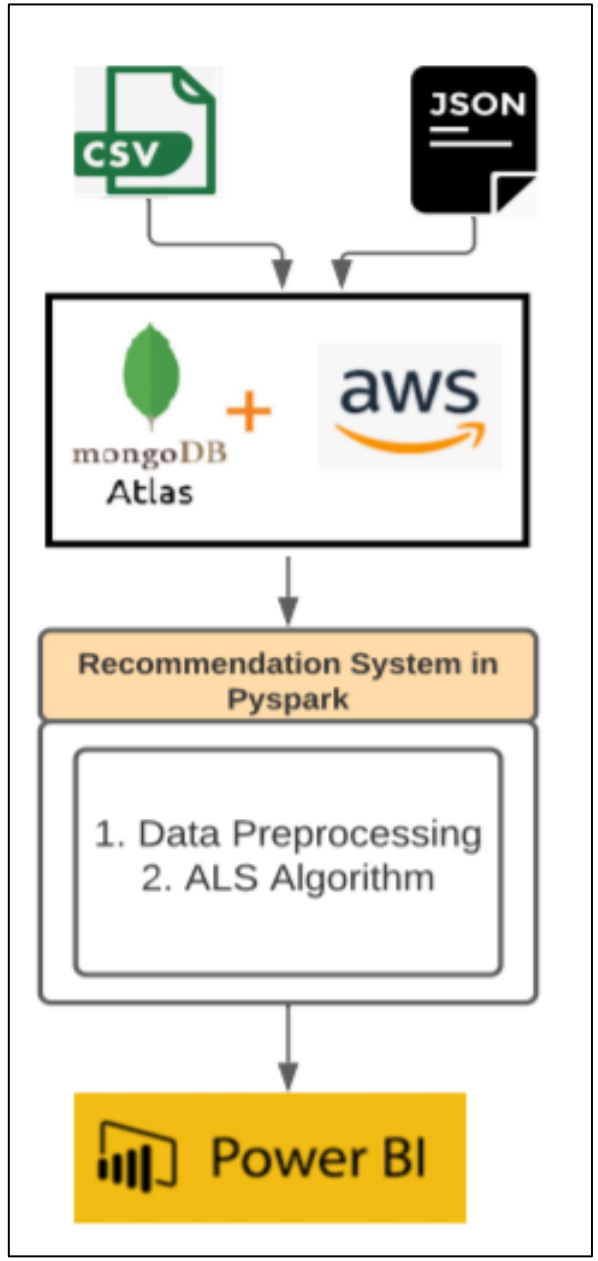
- **Why?**
  - It has wide range of connectors to different data sources like MongoDB, MySQL, Cloud Databases(AWS) etc.
  - Easy to collaborate.
  - Cost effective – Power BI is free!

IMPLEMENTATION

- Created two mongo DB Atlas accounts – each account got up to free 512 MB space on cloud.
- A **default cluster (Cluster 0)** is assigned. All other clusters are paid.
- Basic setup – **admin access**; IP Address as **0.0.0.0/0** to **allows access from anywhere**.
- MongoDB Compass used as GUI to access MongoDB cloud accounts.
- Establish connection to our MongoDB Atlas account; Upload 2 files – metadata and rating files

- Metadata file : 3 columns retained, ASIN number, Title, Categories.
- Category column unstructured - hierarchical categorical structures, multiple category names.
- Alphanumeric characters – **User ID and ASIN** number, not accepted by PySpark ALS. Converted to distinct numerical values using scikit-learn's **LabelEncoder.**
- To avoid NaN values to predictions, set **cold start to "Drop".**
- Performed **3-fold cross validation** and the best model was with **Rank = 25; Regularization Parameter = 0.1, RMSE =2.08**

- ALS System **generates top 5 recommendations** for each user based on ALS. We can change the number of recommendations we wish to get.
- Results from ALS is cleaned and sent to Power BI to visualize the following results –
    - Highly recommended products across all users.
    - Most frequently recommended product categories



RESULTS

Output from ALS Algorithm

	user_index	Recommendation_1	Recommendation_2	Recommendation_3	Recommendation_4
0	148	item_index=258823, rating=1.9737238883972168	item_index=185376, rating=1.9575668573379517	item_index=57139, rating=1.9392974376678467	item_index=183884, rating=1.9349370002746582
1	463	item_index=6945, rating=4.920039653778076	item_index=186827, rating=4.860890865325928	item_index=153268, rating=4.860890865325928	item_index=185376, rating=4.859927654266357
2	471	item_index=228192, rating=3.6455678939819336	item_index=117686, rating=3.5809648036956787	item_index=28488, rating=3.566378116607666	item_index=188371, rating=3.566378116607666
3	496	item_index=162252, rating=4.776736259460449	item_index=63664, rating=4.754911422729492	item_index=231914, rating=4.584717273712158	item_index=228092, rating=4.5077972412109375
4	833	item_index=183884, rating=4.809751510620117	item_index=153268, rating=4.806126117706299	item_index=186827, rating=4.806126117706299	item_index=185376, rating=4.805229663848877
5	1088	item_index=153268, rating=3.9552128314971924	item_index=186827, rating=3.9552128314971924	item_index=185376, rating=3.9120266437530518	item_index=183884, rating=3.906235933303833