Amazon Product Review Recommender System

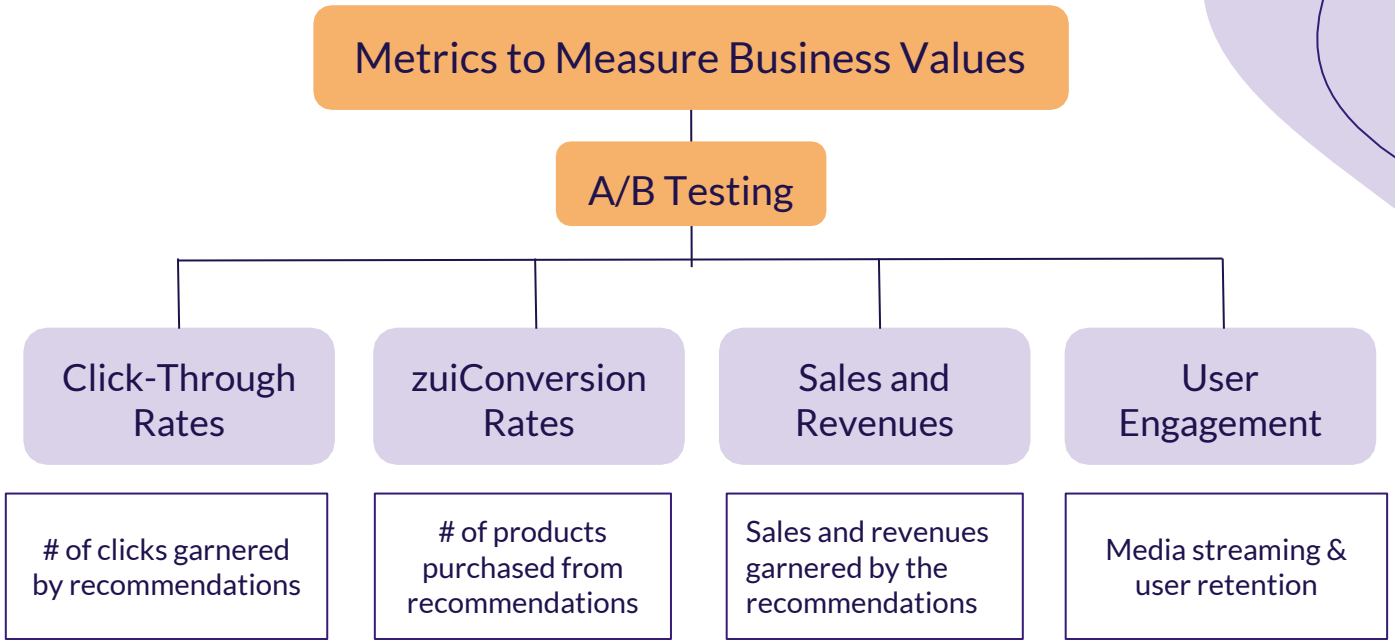
Presented by Sijie Li, Ruixuan Zheng,
Yu Wang, Hongyi Wang

TABLE OF CONTENTS

PROBLEM STATEMENT

Amazon would like to help customers find the “next” right product by building a recommender system using historical review ratings and product popularities in selective categories.

BUSINESS VALUES



"35% of Amazon.com's revenue is generated by its recommendation engine."

- From McKinsey Insights

PROJECT PLAN

Downloaded from dataset API and transferred to RCC HDFS for storage.

DATA

Used Spark API and SQL to read, extract, and manipulate data in PySpark.

EDA

PySpark

MODEL

Used ALS and NLP for building recommender system .

Composed the deliverables in slices.

REPORT

DATA PROFILE

This dataset contains **product reviews** and **metadata** from Amazon, including 41.13 million reviews spanning **May 1996 - July 2014.**

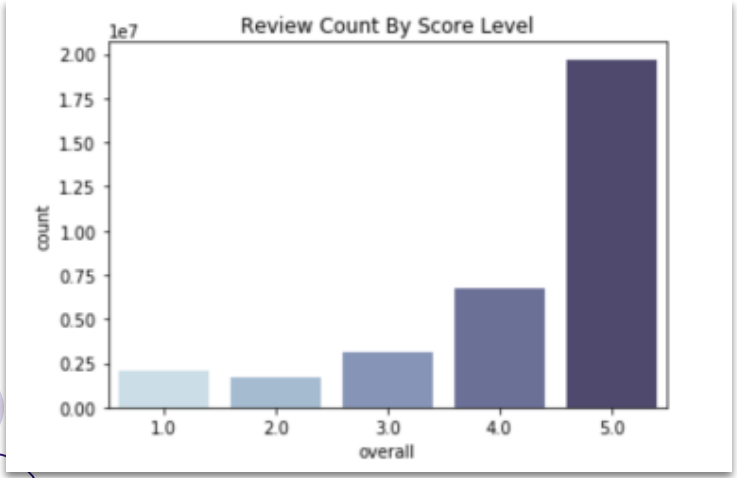
(See all schemas in Appendix)

	Features	Rows	Volumn
Score Dataset	'User ID' 'Rating' 'Review'	41.13 million reviews	9.9 GB
Product Metadata	'Product ID' Product Info	9.4 million products	3.1 GB

RATING INFORMATION

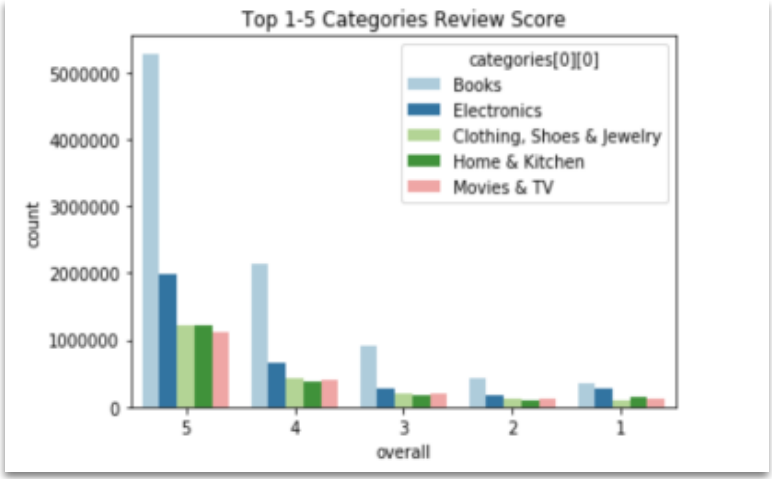
> 50% were full score rating

In all categories, around 55% of rating counts were 5-Star rated.



'Books' had the most reviews

Among all categories, 'Books' had the most reviews and 5-Star ratings.



REVIEW COMMENTS

Most ‘Mentioned’ words in ‘Books’ category

After stopping most neutral words, the reminding words occurred in user’s review comments.

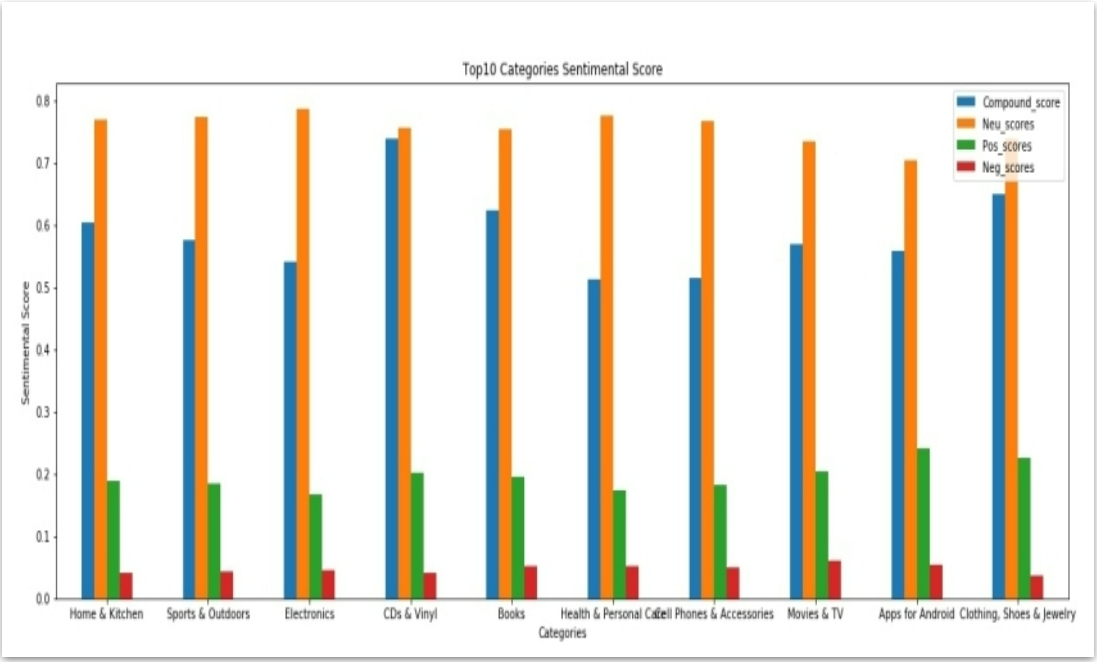
Top 10 categories review comments are processed to show in word cloud. See more in Appendix.

(Example of book category in word cloud)

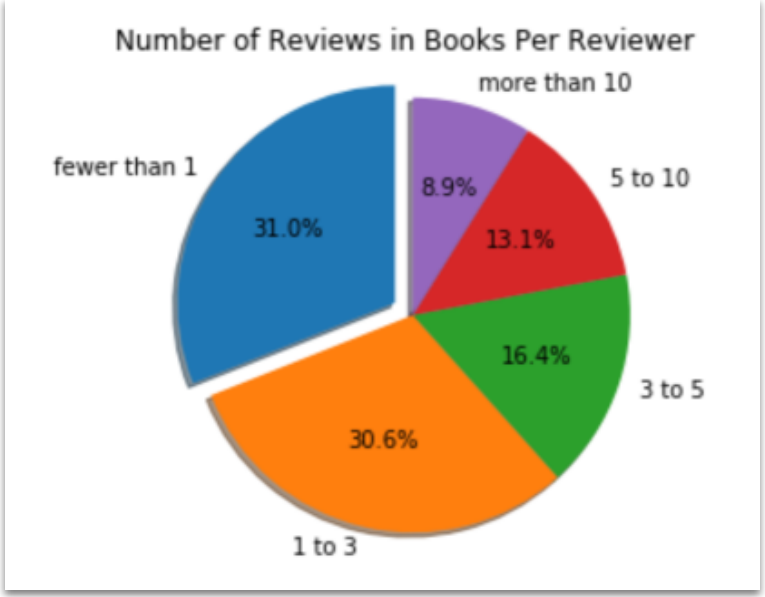
REVIEW SENTIMENT SCORE

More reviews express positive sentiment.

Level of sentiments in reviews among all categories. In general, there were more positive sentiments than negatives.



REVIEWERS INFORMATION



(Example of book category)

Most ‘engaged’ reviewers posted > 10 reviews

By visualizing the count of reviewers regardless ratings, the 8.9 % reviewers posted >10 reviews, while 61% of reviewers posted 0 - 3 reviews.

Top 10 categories review-counts summary are visualized to show in pie charts. See more in Appendix.

PRODUCT INFORMATION

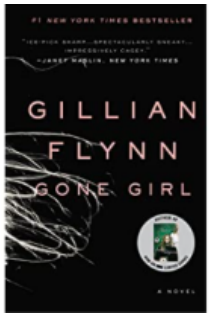
Top-rated
Candy Crush

Review count: 13550
Category: Apps & Games

Runner-up
Kindle Fire 1st

Review count: 11981
Category: Devices

Second runner-up
<<Gone Girl>>



Review count: 10552
Category: Books

RECOMMENDER SYSTEM: RATINGS

Collaborative Filtering › Making automatic predictions (filtering) about the interests of a user by collecting preferences or taste information from many users.

Spark ALS › Alternating least square(ALS) matrix factorization will take a large matrix and factor it into some smaller representation of the original matrix through alternating least squares.

Recommending Books

User-Item
Model RMSE: 2.5

Next **2 books** to purchase for **users 1 & 12**

```
+------+--------------------------------------------+
|userId|recommendations                             |
+------+--------------------------------------------+
|1     |[{40836, 6.119079}, {406113, 6.0965824}]    |
|12    |[{973624, 16.178135}, {1123290, 15.787145}]|
+------+--------------------------------------------+
```

RECOMMENDER SYSTEM: REVIEWS

Keywords Extraction: TF-IDF

- Term frequency–inverse document frequency (**TF-IDF**), is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus.
- **Term frequency**, tf(t,d), is the relative frequency of term t within document d

$$\mathrm{tf}(t,d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}},$$

where ft,d is the number of times that term t occurs in document d

- **Inverse document frequency**, is the logarithmically scaled inverse fraction of the documents that contain the word

$$\mathrm{idf}(t, D) = \log \frac{N}{|\{d \in D : t \in d\}|}$$

Where N is the total number of documents in the corpus

$|\{d \in D : t \in d\}|$ is the number of documents where the term t appears

- **Term frequency, inverse document frequency**

$$\mathrm{tfidf}(t, d, D) = \mathrm{tf}(t, d) \cdot \mathrm{idf}(t, D)$$

RECOMMENDER SYSTEM: REVIEWS

Keywords Extraction: TF-IDF

- We get get several keywords from the tf-idf algorithm, but in order to reduce the data volume and increase computational speed, we only select the top 20 important words in every corpus, of which have the top 20 highest tf-idf scores.
- We group by the product id (**asin**) and get the keywords for every product.
- We group by the reviewer id (**reviewerID**) and get the keywords for every reviewer.

RECOMMENDER SYSTEM: REVIEWS

Keywords Matching

Generate a mapping from keywords to products

```
+----------+--------------------+
|        id|                  kw|
+----------+--------------------+
|0684871408|role,learning,app...|
|1597486310|creepy,quaint,pro...|
|1598692585|light,liked,readi...|
|B00EVDG7HW|enjoyed,talented,...|
|B00KHTMFDE|copy,free,receive...|
+----------+--------------------+
```

```
+-------+-----------------------------------
|kw     |id
+-------+-----------------------------------
|abandon|[0553370758]
|abba   |[1414318677]
|able   |[096798176X, 1583551972, B00HUACEIY,
|abraham|[0934893543]
|access |[0073524581, 1934148091]
+-------+-----------------------------------
```

Collect and sort products
based on keywords for a reviewer

```
+--------------+--------------------+
|            id|                  kw|
+--------------+--------------------+
|A142DRNE2KWF3N|john,grisham,man,...|
|A1AXNELEE02O08|expected,admit,li...|
|A1AYSS3R6K58VZ|box,example,think...|
+--------------+--------------------+
```

```
product[:10]
```

```
['B00EG0GXJC',
 'B00GW6NLP2',
 'B00BPYTNNK',
 'B0090GN6UG',
 '0471289280',
 '0060283181',
 'B00768D6DC',
 'B00D06YI68',
 'B00EMMW6EU',
 'B00DVRB81O']
```

SUMMARY

Models

- Collaborative filtering using SparkALS was able to predict users' preference based on ratings.
- Using keywords extraction and matching was able to find the label from users' review (text) and match users with similar taste on products.

Limitations

- Improvements on ALS model could be done through stringindexer and parameter adjustments for better predictions.
- TF-IDF does not extract keywords with understanding to semantics and could be upgraded to some advanced keywords extractions methods, such as Topic Modeling and PageRank.

Thank you!
Any questions?

Appendix: Data

```
scores.printSchema()
```

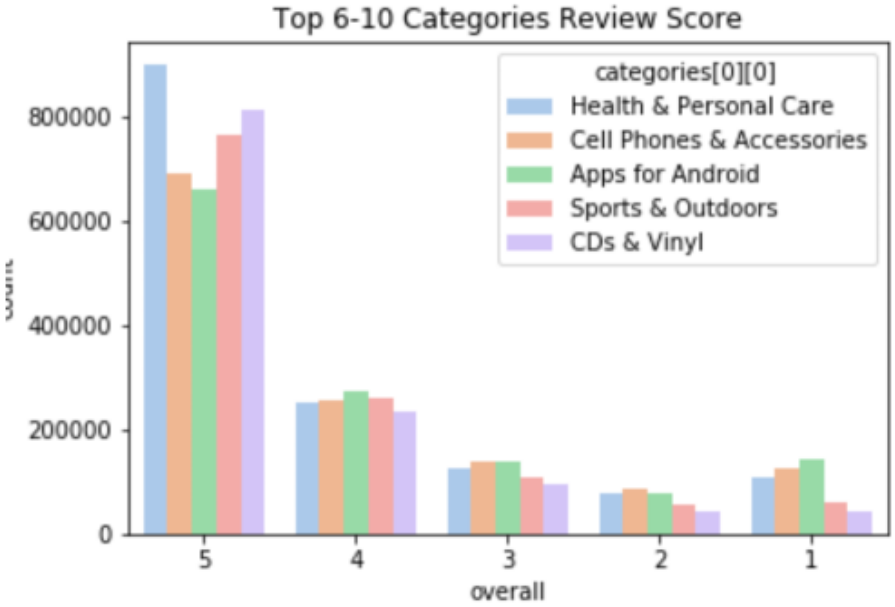
```
root
 |-- asin: string (nullable = true)
 |-- helpful: array (nullable = true)
 |    |-- element: long (containsNull = true)
 |-- overall: double (nullable = true)
 |-- reviewText: string (nullable = true)
 |-- reviewTime: string (nullable = true)
 |-- reviewerID: string (nullable = true)
 |-- reviewerName: string (nullable = true)
 |-- summary: string (nullable = true)
 |-- unixReviewTime: long (nullable = true)
```

Two tables: Scores/rating table on the left and metadata table regarding product information on the right.

```
metadata.printSchema()
```

```
root
 |-- _corrupt_record: string (nullable = true)
 |-- asin: string (nullable = true)
 |-- brand: string (nullable = true)
 |-- categories: array (nullable = true)
 |    |-- element: array (containsNull = true)
 |    |    |-- element: string (containsNull = true)
 |-- description: string (nullable = true)
 |-- imUrl: string (nullable = true)
 |-- price: double (nullable = true)
 |-- related: struct (nullable = true)
 |    |-- also_bought: array (nullable = true)
 |    |    |-- element: string (containsNull = true)
 |    |-- also_viewed: array (nullable = true)
 |    |    |-- element: string (containsNull = true)
 |    |-- bought_together: array (nullable = true)
 |    |    |-- element: string (containsNull = true)
 |    |-- buy_after_viewing: array (nullable = true)
 |    |    |-- element: string (containsNull = true)
 |-- salesRank: struct (nullable = true)
 |    |-- Appliances: long (nullable = true)
 |    |-- Arts, Crafts & Sewing: long (nullable = true)
 |    |-- Automotive: long (nullable = true)
 |    |-- Baby: long (nullable = true)
 |    |-- Beauty: long (nullable = true)
 |    |-- Books: long (nullable = true)
 |    |-- Camera &amp;amp; Photo: long (nullable = true)
 |    |-- Cell Phones & Accessories: long (nullable = true)
 |    |-- Clothing: long (nullable = true)
 |    |-- Computers & Accessories: long (nullable = true)
 |    |-- Electronics: long (nullable = true)
 |    |-- Gift Cards Store: long (nullable = true)
 |    |-- Grocery & Gourmet Food: long (nullable = true)
 |    |-- Health & Personal Care: long (nullable = true)
 |    |-- Home &amp;amp; Kitchen: long (nullable = true)
 |    |-- Home Improvement: long (nullable = true)
 |    |-- Industrial & Scientific: long (nullable = true)
 |    |-- Jewelry: long (nullable = true)
 |    |-- Kitchen & Dining: long (nullable = true)
 |    |-- Magazines: long (nullable = true)
 |    |-- Movies & TV: long (nullable = true)
 |    |-- Music: long (nullable = true)
 |    |-- Musical Instruments: long (nullable = true)
 |    |-- Office Products: long (nullable = true)
 |    |-- Patio, Lawn & Garden: long (nullable = true)
 |    |-- Pet Supplies: long (nullable = true)
 |    |-- Prime Pantry: long (nullable = true)
 |    |-- Shoes: long (nullable = true)
 |    |-- Software: long (nullable = true)
 |    |-- Sports &amp;amp; Outdoors: long (nullable = true)
 |    |-- Toys & Games: long (nullable = true)
 |    |-- Video Games: long (nullable = true)
 |    |-- Watches: long (nullable = true)
 |-- title: string (nullable = true)
```

Appendix: EDA



Among all categories, ‘Books’ had the most reviews and 5-Star ratings. Here organized the categories ranked at 6 - 10 places. All categories had most 5 - star ratings.

Appendix: EDA

Most Mentioned Words in Apps for Android Category

Most Mentioned Words in CDs & Vinyl Category

Appendix: EDA

Most Mentioned Words in Cell Phones & Accessories Category

Most Mentioned Words in Clothing, Shoes & Jewelry Category

Appendix: EDA

Most Mentioned Words in Electronics Category

Most Mentioned Words in Health & Personal Care Category

Appendix: EDA

Most Mentioned Words in Home & Kitchen Category

Most Mentioned Words in Movies & TV Category

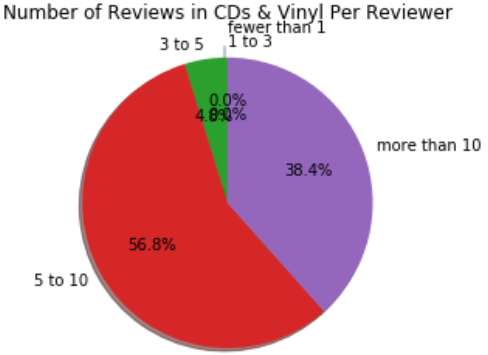
Appendix: EDA

Most Mentioned Words in Sports & Outdoors Category

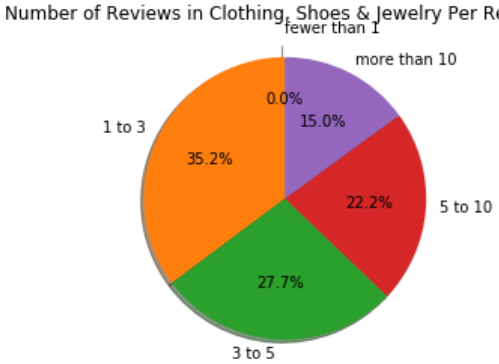
From all the Wordcloud, we can see that the most mentioned words in every categories are closely related to the categories themselves (e.g., album for CDs & Vinyl Category; movie, film for Movies & TV Category).

Plus, reviewers tend to comment positively for products with “great”, “good”, “like” being the most frequently mentioned words in almost all categories.

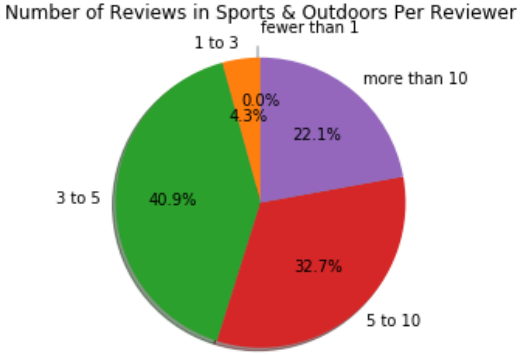
Appendix: EDA



In CDs & Vinyl category, reviewers with 5 to 10 reviews rank first, followed by reviewers who have reviews more than 10.



In Clothing, Shoes & Jewelry category, reviewers with 1 to 3 reviews rank first, followed by reviewers who have reviews from 3 to 5.



In Sports & Outdoors category, reviewers with 3 to 5 reviews rank first, followed by reviewers who have reviews from 5 to 10. Those who have reviews fewer than 1 have the smallest share.