MIT805 Assignment 2

Part 1 – MapReduce report

Stephan Kersop – u16095953

30 October 2022

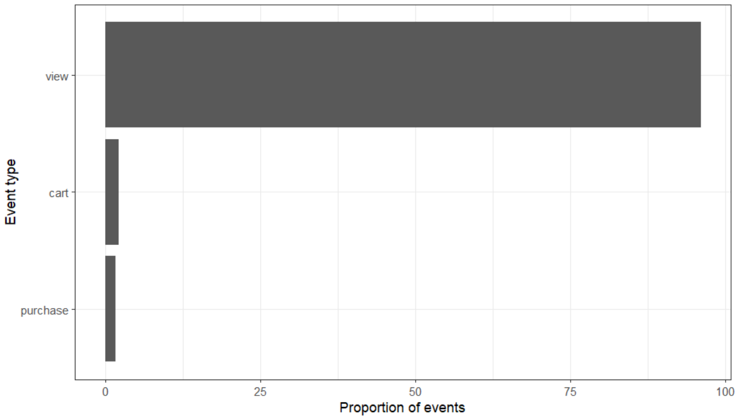
1. Introduction

For this project, I am working with the *eCommerce behaviour data from multi category store* dataset from kaggle.com [1]. This dataset contains event data from a multi-category online store for the months of October and November 2019. I will apply MapReduce algorithms and various visualisations to the October data from this dataset to gain new insights that can be used to create a competitive advantage in the online store space. This is part 1 of the assignment and focuses on the applied MapReduce algorithms and their results.

2. Summary from Assignment 1 – Data report

For Assignment 1, I wrote a report discussing the origins and structure of the dataset. I provide a condensed version of the discussion in this section.

Figure 1: Event types as a proportion of observations



The chosen dataset contains 9 fields (columns) for 42 448 764 events (rows), where an event represents a user interaction with the online store page. Data is stored as a single CSV file and contains no erroneous entries.

Most events were *view* events, with the small number of remaining events being split between *cart* and *purchase* events (figure 1). Only two fields, *category_code* and *brand*, are allowed to be empty, and these fields are empty for 32% and 14% of events, respectively.

The most popular product category is *electronics.smartphone* (henceforth *smartphone*), constituting 27.1% of all user interactions for October 2019 (Figure 2). The most popular product brands are all major smartphone manufacturers (Figure 3).

Figure 2: Four most popular item categories

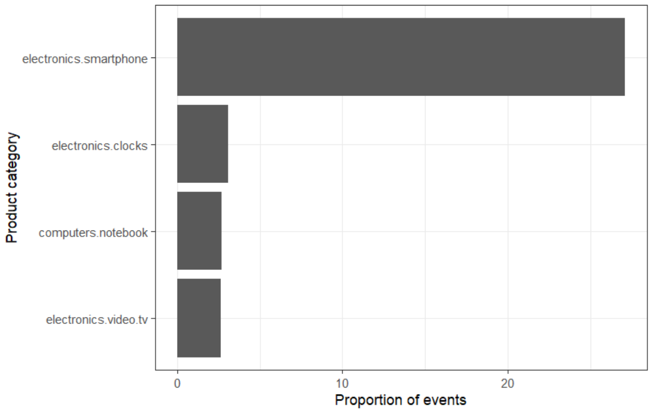
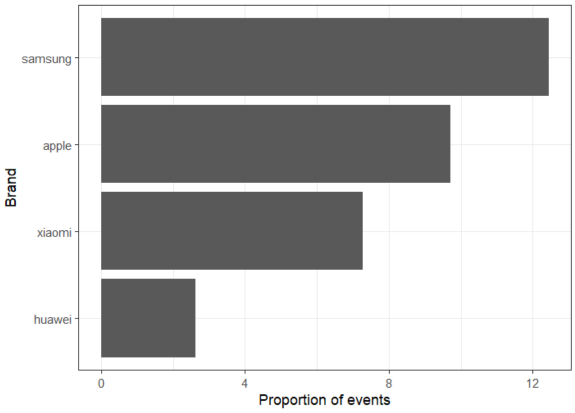


Figure 3: Four most popular product brands



Below, I have included a table from Assignment 1 which lists all fields in the dataset along with a short field description.

Table 1: List of fields and their descriptions

Field name	**Field description**
event_time	This field contains a date and time value for when the event occurred in the UTC time zone
event_type	Character string describing the type of event. Can be one of *view*, *cart*, *remove_from_cart*, or *purchase*
product_id	Integer ID for the product that was interacted with during the event
category_id	Integer ID for the product category that the interacted product is associated with
category_code	Character string describing the product category that the interacted product is associated with
brand	Character string of the brand name that the interacted product is associated with
price	Price of the product in an unknown currency
user_id	Integer ID for the user that created the event
user_session	Unique identifier for the user session that the event is associated with. User sessions can persist between events, but are seen as ended if no event occurs within the session for a certain amount of time

3. Objective

For this component of the assignment, the goal is to use MapReduce algorithms to reduce the dataset into meaningful results that can be used to obtain a competitive advantage in the online store space. Smartphones are the main driver behind store interactions (Figure 3) and understanding user sessions and interactions with products in this category as the focal point will potentially offer insight into how the store can target such user sessions to gain a competitive advantage.

To gain these insights, I will attempt to answer the following questions:

1. What are the most popular *smartphone* brands based on the number of purchases, and what is the mean purchase price for each of these brands?
2. What non-*smartphone* product categories are often interacted with in sessions where users interact with *smartphone* products?
3. What brands are often interacted with for non-*smartphone* products in sessions where users interact with *smartphone* products?
4. What product categories are popular in non-*smartphone*-related user sessions during timeframes with the highest number of *smartphone*-related interactions?

4. Methodology

To answer these questions, I will implement Hadoop MapReduce algorithms using Java, with pre- and post-processing done in *R*. Thereafter, I will generate visualisations of the data with *R*, using both the reduced datasets as well as the original dataset.

All four of the questions identified in section 3 require filtering to occur. Filtering in MapReduce is rather straightforward and utilizes only the *map* component of MapReduce. Simply put, I only map values that satisfy the filter condition. In pseudocode, this can be represented as:

```
map(key, record):
    if record satisfies the filtering condition:
        output key, value
```

Question one is focused on events related to the purchasing of smartphones. I can therefore apply the above pseudocode with the filtering condition being that the product category is *smartphone* and the event type is *purchase*. This makes up the map component of the algorithm. I output the key as the product brand and the value as a custom tuple that contains the price and a count. I can then calculate the mean and the number of sales for each brand in the reducer component of the algorithm. This is done using the following pseudocode, where the output value is once again an instance of the custom tuple:

```
reduce(key, list of values):
    initialise sum and count variables
    for each value:
        add value to the sum of values
        increase count variable
    calculate mean as sum/count
    output(key, value)
```

Questions two and three require a summarization of all non-smartphone products within user sessions that contained an interaction with a smartphone. I will therefore need to obtain a filtered dataset containing all events that form part of a user session in which an interaction with a smartphone occurred, but which does not represent an interaction with a smartphone itself. Obtaining such a set using MapReduce requires a two-step process. Firstly, I find all distinct user sessions that contained an interaction with a smartphone and save these session IDs as a text file. This is done using the simple filter framework discussed at the start of this section. Then, I perform an inner join between this list of sessions and the original dataset to filter out all non-smartphone-related events. This is done using a replicated join, a map-side process implemented using a HashMap. A replicated join can be represented in pseudocode as:

```
setup():
    initialise hashmap
    for each record in data to be joined on:
        map (in hashmap) the join field as a record id
map(key, record):
    get join field from record as an id
    check hashmap contents at id
    if hashmap contents at id is not null
        output(key, value)
```

Once again, no combiner or reducer is required, as I am filtering based on a list and filtering is purely a mapping task. This task will output a set of non-*smartphone*-related events that occurred in the same user session as a *smartphone* interaction. This dataset can then be used to answer questions two and three.

Now, I filter out events representing interactions with *smartphone* products. Once again, this can be done with the filter pseudocode from above, with the key being either the brand or the product category (depending on the task) and the value simply being one (1). In the reducer, I then sum the value of one (1) across all records with the same key, thereby producing an occurrence count, which is then used as the output value.

```
reduce(key, list of values):
    initialise sum variable
    for each value
        add value to the sum of values
    output(key, value)
```

Finally, question 4 is split into three parts. First, I find the distribution of smartphone interaction events across the hours of the day. This is done by applying the count methodology from questions two and three (without the filter component) to the original dataset, with the key being the hour of the day. This will then provide an occurrence count of smartphone-related events for each hour of the day. The second part of the task is then to retrieve the busiest two hours from this output. Since the output is always at most 24 lines long (one for each hour of the day), it is not sensible to perform this task using MapReduce. A simple Java application is used instead, which simply tracks through all the hours and compares them to find the two with the highest occurrence counts. Finally, I apply the standard filter-count methodology, but I filter the original dataset to contain only non-*smartphone*-related events for the peak hours identified in the first part of the task. This will then provide an occurrence count for each non-*smartphone*-related category for the hours when the most *smartphone*-related events occur.

5. Results

Question 1: *What are the most popular smartphone brands based on the number of purchases, and what is the mean purchase price for each of these brands?*

Table 2: Average price and number of devices sold by brand

Brand	Average Price	Number Sold
samsung	261	143 123
apple	889	115 345
xiaomi	208	38 776
huawei	215	21 882
oppo	222	10 891
other	247	2 714
vivo	245	2 025
meizu	126	1 686
honor	248	555
nokia	163	512
oneplus	690	509
overall	465	338 018

The results for the corresponding MapReduce task are set out in Table 2. For this task, only brands that sold more than 500 devices were considered on their own. All brands that sold less than 500 devices were aggregated under 'other'.

The average sales price for smartphones in October 2019 was 465. The most expensive brands on average were *apple* and *oneplus*, while the cheapest brands on average were *meizu*, and *nokia*. The variance for the average smartphone price per brand is 763. This variance is extremely large, suggesting that brands are not competing on equal footing and that smartphone products are very diverse.

Despite their low average sales prices, *meizu* and *nokia* are two of the least popular of the major smartphone brands, and despite its extremely high average sales price, *apple* is the second most popular smartphone brand, magnitudes above the third most popular brand in sales volume. This suggests that factors such as brand loyalty, brand-specific product characteristics, and variety in products offered may have a substantial impact on product popularity, whereas price has a much smaller impact.

Question 2: *What non-smartphone product categories are often interacted with in sessions where users interact with smartphone products?*

Results for this question (as tabulated in table 3) indicate that the categories most popularly interacted with during sessions that smartphones were interacted with are also electronics. Specifically, the top four categories are *headphones*, *clocks*, *notebooks*, and *tv*'s. The store should suggest products from these categories to users who are viewing *smartphone* products, as they are likely to be interested in these products as well.

Table 3: Number of interactions with various non-smartphone categories during sessions containing interactions with smartphone products

Category ID	Category Code	Count
2053013554658804200	electronics.audio.headphone	144 627
2053013553341792800	electronics.clocks	119 767
2053013558920217300	computers.notebook	93 241
2053013554415534300	electronics.video.tv	86 807
2053013558525952500	No category code	69 312
2053013555573162200	electronics.telephone	67 708
2053013553375347000	No category code	58 285
2172371436436455700	electronics.tablet	45 795
2053013553559896300	No category code	44 463
2053013563810776000	appliances.kitchen.washer	41 806

Question 3: *What brands are often interacted with for non-smartphone products in sessions where users interact with smartphone products?*

Table 4: Number of brands represented in non-*smartphone*-interactions during sessions containing interactions with smartphone products

Brand	Count
No brand	263 115
samsung	168 246
apple	145 743
xiaomi	132 388
sony	40 604
nokia	36 043
acer	35 616
lg	35 590
lucente	28 924
lenovo	27 009

Table 4 contains the results for this question. Interestingly, a large portion of the investigated events are not associated with a specific brand. This suggests that these events are related to items that offer alternatives to the major brands prevalent in the store: *samsung*, *apple*, and *xiaomi*. These brands are the second to fourth most prevalent in non-smartphone events that occurred within the same user session as a *smartphone* event. This makes intuitive sense when evaluating within the context of our question 2 results: these brands are major producers of *smartphone* accessories, *tablets*, *notebooks*, and *tv*'s. Similarly, the remaining brands on the list are all major tech brands.

Question 4: *What product categories are popular in non-smartphone-related user sessions during timeframes with the highest number of smartphone-related interactions?*

Table 5: Number of interactions with various non-smartphone categories during peak *smartphone* interaction windows

Category ID	Category Code	Count
2053013553559896300	No category code	253 135
2053013554415534300	electronics.video.tv	147 171
2053013558920217300	computers.notebook	146 526
2053013554658804200	electronics.audio.headphone	120 671
2053013563651392300	No category code	116 660
2053013563810776000	appliances.kitchen.washer	112 980
2053013556168753700	No category code	102 076
2053013565983425500	appliances.environment.vacuum	97 199
2053013561579406000	electronics.clocks	92 308
2053013563911439400	appliances.kitchen.refrigerators	81 914

Results show a large overlap between popular product categories in non-*smartphone* user sessions during the hours with the most *smartphone*-related sessions and product categories within *smartphone*-related sessions. Results are tabulated in table 5, with categories that do not appear in the top ten most popular categories in *smartphone*-related sessions (Table 3) being shaded. These categories include more appliances, an alternative category for clocks, and two unnamed product categories.

6. Conclusion

In this essay, I apply MapReduce algorithms to the eCommerce behaviour data from multi category store dataset to better understand smartphone-related user sessions. Results indicate that smartphone pricing and popularity are brand dependent, which I speculate is due to brand-specific product characteristics. Additionally, product categories that are popular within *smartphone*-related sessions, as well as non-*smartphone*-related sessions during times with a high number of *smartphone*-related sessions, are dominantly appliances and other electronics, often produced by the most popular *smartphone* brands. To gain a competitive advantage, the store can apply this knowledge by suggesting products from these brands and categories to users who are viewing *smartphone*-related products.

7. References

[1] "eCommerce behavior data from multi category store." https://www.kaggle.com/datasets/mkechinov/ecommerce-behavior-data-from-multi-category-store (accessed Sep. 08, 2022).