A Work Project, presented as part of the requirements for the Award of a Master's degree in Management from the Nova School of Business and Economics.

A RECOMMENDER SYSTEM FOR PINGO DOCE & GO NOVA

MIGUEL ÂNGELO SALGUEIRO BEZERRA

Work project carried out under the supervision

of:

Qiwei Han and Rui Tomás

04-01-2021

Abstract: Using the Design Science framework, and acknowledging the success of recommenders in e-commerce settings, this paper proposes the design and implementation of a recommender in a physical retail store (Pingo Doce & Go Nova). It allows to assess if the recommender can influence customers' decisions, increase sales, the number of unique products acquired, and understanding the customers. To develop it, the data was collected, curated, recommendation strategies were designed (loyalty, novelty, and related) and the customers were split into groups. The recommender will be deployed in the store app and, after, the results from the metrics will be analyzed.

Keywords: Data Science; Physical Retail; Recommender; Design Science Research Process

Acknowledgements: This work would have not been possible without all the support provided by my two advisors, Professor Qiwei Han and Mr. Rui Tomás. A special thanks for Jerónimo Martins and all their collaborators, for providing all the necessary data, their knowledge and this fantastic opportunity. Finally, I am very thankful and grateful for all my family, especially my parents and grandparents, as they were the ones who allowed and supported me throughout all this journey.

This work used infrastructure and resources funded by Fundação para a Ciência e a Tecnologia (UID/ECO/00124/2013, UID/ECO/00124/2019 and Social Sciences DataLab, Project 22209), POR Lisboa (LISBOA-01-0145-FEDER-007722 and Social Sciences DataLab, Project 22209) and POR Norte (Social Sciences DataLab, Project 22209).

Introduction

The evolution of technology, computers, and the associated computing power, to be more specific, enabled the appearance of several solutions to previously existing problems, but it also allowed the debut and rise of non-existing heretofore businesses and business models. One such case is e-commerce businesses/platforms, that have been thriving in the past years, which endured the adverse effects of the Covid-19 pandemic has had on the world's economy, adapted to the new conditions, and led to a boost in their performance. For instance, Amazon.com, Inc. reported that its net income in the first quarter of 2020 was smaller than that of the previous year, despite the increase of its net sales in the same period (Amazon.com, Inc. 2020). However, in the second and third quarters of 2020, its net sales and net income surpassed the registered levels in 2019 (Amazon.com, Inc. 2020; Amazon.com, Inc. 2020).

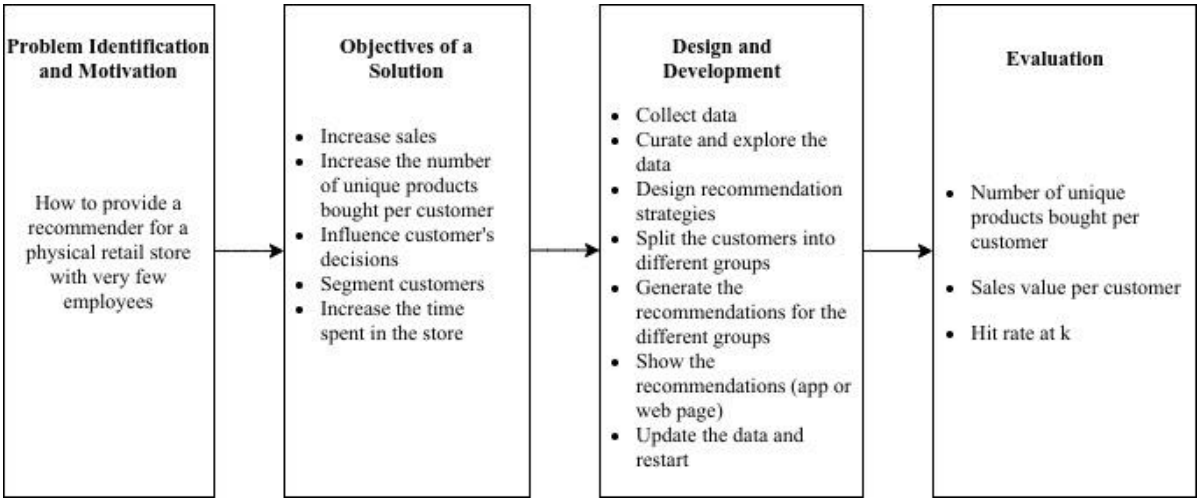
However, even before the Covid-19 pandemic, e-commerce platforms have been taking advantage of recommenders to help customers search through their "virtual inventory" of hundreds of thousands, if not millions, of products (Hinz and Eckert 2010). Despite the customers having an initial idea about the type of product they want, the set of available options is astonishing, forcing them to deal with the "paradox of choice" (Schwartz 2015). By retrieving relevant information for the customers to match their preferences, the recommenders help the customers reduce the search costs and the uncertainty associated with the information search (Ariely 2000). Furthermore, customers are not the only ones benefiting from that technology, as retailers can also capture value from its use. On the one hand, sales can be increased when recommenders augment up-selling and cross-selling chances. On the other, the customer's perception of the retailer's usefulness and loyalty can be improved (Schafer, Konstan and Riedl 2001; Pathak et al. 2010).

This technology enabled the success of e-commerce platforms, and in this paper, it will be presented and discussed how a recommender can influence the physical retailing industry,

which is being threatened by the fall of profit margins, by rising cost pressures and more competition from online retailers (Brynjolfsson and Smith 2000; Brynjolfsson, Hu and Rahman 2009; Lieber and Syverson 2012). For that purpose, in partnership with the company Jerónimo Martins (JM), a recommender will be developed to be deployed at the Pingo Doce & Go Nova (PD&Go Nova) store, located on the campus of Nova School of Business and Economics.

Furthermore, this paper will follow the Design & Development Centered Approach of the Design Science Research Process, presented by Peffers et al. (2006) also discussed by Hevner et al. (2004). The Design Science Research Process has been used to describe the development of other applications/technologies inserted in the Information Systems field, “the intersection of IT and (business) organizations” (Peffers et al. 2006). With this framework, one can frame the creation and the development process within the following structure: "problem identification and motivation," i.e., in which the problem is identified and how a solution can provide value; "objectives of a solution," in which the objectives of the proposed solution to the problem are announced; "design and development," in which it is described the creation process of the solution; "evaluation," in which it is discussed the metrics to evaluate the solution and how well it addresses the proposed objectives. Peffers et al. (2006) proposed two extra sections ("Demonstration" and "Communication"), which will not be addressed. Figure 1 presents the whole Design Science Research Process inherent to this project. However, outside that framework, this paper will include two other sections: Limitations and Challenges (discussion about the project limitations) and Future Steps (how the future should be approached and what more can be done having this project as a basis).

Figure 1 - Diagram of the Design Science Research Process for the Recommender for PD&Go Nova



Problem Identification and Motivation

The first step of the Design Science Research Process is identifying the problem to be tackled and why addressing it matters. The first thing to be acknowledged is that the goal of every firm is to make a profit, thus a tool that enables this is a desirable one. At PD&Go Nova, there are no cashiers, and the few employees working there are either responsible for preparing food for the customers or for replacing items, apart from a security guard. With this setting, it is possible to see that this store is different from a typical store of the physical retail industry, where one would be able to ask an employee for help to find a product. Also, since this store is in a university, its usual clientele, composed mostly of students, and their needs, are not similar to what one would find in a typical physical retail store.

As discussed before, e-commerce platforms use recommenders not only to help customers find the most adequate item for their needs (Hinz and Eckert 2010) but also to increase their sales and to develop a better perspective of them in the eyes of their shoppers (Schafer, Konstan and Riedl 2001; Pathak et al. 2010). Unlike that setting, this store has a lower set of unique articles to be purchased (around 2,000), a number which highly contrasts with the e-commerce platforms' numbers. Another difference that should be noted is that, in a physical retailer, the

customer can touch and feel the product, while e-commerce customers solely rely on the information provided. Additionally, in the physical retail store, the customer also has the onus of going to the store and grab the article, which is not something required with e-commerce platforms, as the only thing required from the customers are a few clicks or taps. Nevertheless, PD&Go Nova can take advantage of the recommender the same way as e-commerce platforms and to show the customers some products they did not know that was available there.

One additional detail about this store is that, on average, a customer takes 4 minutes from the moment he enters the store until he leaves it. Once again, such duration is quite different in comparison with a "regular" physical retail store, in which their journey takes about 46 minutes, according to Zeballos (2020), referring to the Americans who were 15 years or older while grocery shopping in the period of 2014 to 2017. Such duration indicates the customers tend to have a formed idea about what they want to purchase, which turns the task of the recommender into a very challenging one. Furthermore, from the gathered data, most of the transactions included only a handful of products (Table 1), which also helps explain such duration. Despite the recommendations being available, even if the customers are not inside the store, the app that allows them to enter the store and add products to their basket seems to be relevant only when they are waiting to enter the store and when they are inside it. For that reason and considering the small amount of time the customer is in the store, it is possible to comprehend why such a task is a difficult one.

Despite the demanding setting, a recommender is a tool with the potential to increase JM's profits, as it may trigger the customers to visit the store more frequently, and/or to purchase a wider variety of products, increasing the chances of the customers to spend more money.

Objectives of a Solution

The second step of the Design Science Research Process is defining the objectives of a solution to the problem identified before. Hence, in addition to an increase in JM's profits, the

recommender can be used to understand the customers better, as different recommendation strategies can be applied. Another goal is to assess if the recommender can influence people's purchasing behavior. Additionally, with different strategies, one may be able to distinguish and segment customers. One more goal is to increase the time the customer spends in the store; that is, the customer can spend more time within the store, not due to waiting in lines, but rather to travel more inside the store and grab products from areas he usually does not visit.

Design and Development

The next step of the Design Science Research Process is describing the creation of the solution to address the problem identified and fulfill the objectives aforementioned.

The data used for the recommender was provided by JM. It concerns the transactions that happened at the PD&Go Nova, which ranges from October 3, 2019, until December 27, 2020. Moreover, also provided was the dataset containing information related to the products according to Pingo Doce's market structure. Additionally, the dataset about the information of the customers, which was anonymized, was also provided. In Figures 2, 3, and 4, one can see a sample of the transactions', market structure, and customers' datasets, respectively. Although there were a wide variety of columns, only some were used, which can be found in Table 2 along with their descriptions.

Although "NIFCOUNTRY" has several missing values, this column still has value because it can help identify which customers are Portuguese and which ones are not, which was not explicit in the collected data. Table 3 describes the creation process of the "Portuguese" column.

In terms of data curation, repeated customer identifiers were removed, so that the three datasets could be merged, due to the common columns in the datasets. Then, commas were replaced by dots; the column "full_transaction_date" was created by combining the columns "ID_DIA" and "HORA_TXS"; and the lines with missing values were removed.

As some lines of the dataset corresponded to returns (when the values in the QTE column were negative, as zeroes corresponded to products in promotions), they were removed. This procedure brings an inherent limitation: by removing the returns without removing the corresponding purchases, the quality of the recommendations may be flawed, as keeping the returns and acknowledging them as an active feedback mechanism could be beneficial. However, it was not possible to do so because there were several cases with returns without a matching purchase or returns that happened before a purchase.

During the time in which the data was collected, there were items whose identifier (values of the ID_ARTIGO column) was changed and it was necessary to ensure it was up to date so that outdated items were not recommended or that the same product was not suggested twice. Table 4 shows the process to fix the problem of the items with more than one identifier.

Many recommenders suffer from the cold start problem, i.e., when the algorithm is required to make personalized recommendations to customers with no prior purchases or to recommend articles which were not previously considered by the recommender. To mitigate this issue, especially from the new customers' perspective, the recommender will not attempt to provide tailored recommendation to customers responsible for less than five transactions. The process to do the task mentioned before is described in Table 5.

The last task of preprocessing the data was to remove the entries related with the oven service, which is a service the customers can acquire when they purchase a pizza, and the recycling incentive from Nova SBE associated to several packages. Both are cases of products which should not be recommended.

Descriptive Statistics and Exploratory Data Analysis

After all these procedures, one can see a sample of the final dataset in Figure 5. Table 6 shows the descriptive statistics of the curated dataset, with products in each row. The main conclusion one can take from it is that the data is skewed to the right, that is the customers buy

few products and do not spend much money on each transaction. After analyzing Table 1, with the data grouped by the transactions, the conclusions that can be drawn are very similar. The picture does not change if the data is grouped by customers or even by products, only the perspective and the interpretation vary a bit: most of the customers are responsible for few transactions in which not much money is spent and, for the products, the majority of them sold only some dozens of units, while there are some outliers with thousands of units sold. Tables 7 and 8 show the aforementioned information, respectively. When considering the number of unique products acquired by customers, the situation holds: the majority of the customers acquired a few dozens of products, while few acquired more than a hundred. To illustrate that situation, there is Table 9 and Figure 6.

Recommendation Strategies

Loyalty

The simplest recommendation strategy developed was named loyalty. In this paper, loyalty is the strategy to which the recommended articles were bought more times by the customers, with the underlying objective of assessing the value of reminding the customers of the products they buy more frequently. Therefore, the recommendations a customer assigned to this group will be as follows: the top recommendation will be the item he bought the most times, followed by the second most frequent, and so on until the k-th, being k the number of personalized recommendations.

Unless the customer has had a bad experience, one can suppose these are the articles the customers like: sporadic purchases or items the customer disliked, which are likely to be products without many repeated purchases, will not be the top recommendations. On the other hand, the articles the customer keeps purchasing repeatedly should be the ones they like the most and/or find more valuable.

This strategy is flawed in terms of diversity, as only products acquired by the customers will be recommended. From this perspective, it is clear that in addition to the cold start problem referring to new items (those added to the available set), the rules behind the generation of the recommendations will not be able to suggest items never purchased by the customer, either already existing or new ones. Another limitation is that, due to the setup of this strategy, the number of tailored recommendations depends directly on the number of unique products the customer has previously purchased. For instance, a customer who only bought 6 different products will not be able to have 10 tailored recommendations.

Novelty

In this paper, the novelty strategy is not related to new products added to the set of available ones. Instead, it follows the same principle of the loyalty strategy, i.e., the recommendations are the items the customer has purchased before, sorted by the number of times they were acquired in descending order. The difference between the loyalty and novelty strategies is that the latter includes a time restriction of 30 days, i.e., a customer assigned to this group will be only recommended products not acquired in the previous 30 days.

The goal of this strategy is different from the loyalty one: while the latter aims to assess the value of recommending the products the customer acquires the most, the former aims at studying the impact of recommending items the customer has not bought in some time.

Moreover, as this strategy is built in the same way as the loyalty one, it suffers from the same flaws already discussed in the previous section. In the loyalty strategy case, there can be customers whose number of tailored recommendations does not reach k (the desired number of recommendations), due to the reduced number of different articles acquired. In the novelty case, the time restriction can restrict even further the possibility of providing the k recommendations because, in this case, the recommendations can only concern products that were not purchased

in the previous 30 days. It is worth mentioning that the number 30 was chosen arbitrarily, meaning that such value could be subject to be tested, to see which timeframe is the best.

Related

The last recommendation strategy is called related. The algorithm used to generate the recommendations of this strategy was Neural Collaborative Filtering (NCF), which can be found in Microsoft's Recommenders' GitHub repository ("Microsoft/Recommenders" 2020), which was based on He et al. (2017). Unlike the previous cases, in which the customers were considered individually, the algorithm behind this strategy is a Collaborative Filtering (CF) one; that is, it is an algorithm that uses "similarities between users and items simultaneously" ("Collaborative Filtering" 2020). This approach assumes that if a person has the same opinion/taste like a second one, it is more likely that the first's opinion/taste about another subject is more similar to the one of the second person than to that of a randomly chosen individual. Moreover, this algorithm is a deep-learning model, which "generalizes the matrix factorization problem with multi-layer perceptron" (González-Fierro 2018).

It was necessary to create one column with the "ratings," which had only one value (1) to note that there was an interaction (a transaction in this case) between the customer and the product in the cause. This was done because the only available data is the purchase history, and it is not possible to clearly identify what would be the negative purchase history, i.e., the items the customer considered buying but ended up leaving on the shelf. Naturally, more value would be gained if there was a rating/review system in place for the customers to classify how satisfied they were with the items they have bought.

The necessary columns for this algorithm were: customer identifier, product identifier, and the ratings column. However, one underlying assumption when developing these models is that the future will resemble the past, so the full_transaction_date will also be used to split the data

for the training and test sets, leaving the first 70% of the transactions for the former and the remaining 30% for the latter.

Before discussing the hyperparameters of the model, it is essential to note that the output is a list of the articles the customer has never purchased with the corresponding probability of them being recommended to that customer on a scale ranging from 0 to 1. Despite NCF's ability to recommend previously purchased products, their inclusion would overlap with the other recommendation strategies, which would prevent a clear analysis of the effect of the different strategies.

The hyperparameters available in this algorithm to be tuned are described in Table 10 along with the chosen values. To choose the values for each hyperparameter, several models were built, and evaluated according to the hit rate at k, being k the top number of items. In this setting, the transactions in the test set for each customer were looked at, and it was a "hit" whenever one of the articles acquired was the same as one of the recommended for that customer. In the end, the number of "hits" is divided by the number of transactions in the test set (Li 2019).

Unlike the previous recommendation strategies, the clients belonging to the related group will not see the products they have previously acquired as recommended. Instead, they will only see those which they have never purchased before. Furthermore, in this case, a lack of tailored recommendations can only occur if a customer has acquired all the available articles but k.

Experiment Groups

The context in which this store is inserted allows assessing if a recommender system works and to test which recommendation strategies are better. To measure that, the customers were randomly and uniformly assigned into four different groups, in which three will receive personalized recommendations according to the strategies described earlier and the other one will receive random recommendations and serve as the control group. As this store is inserted

in a school of a university and most of the customers are students, only customers who were active after September 1, 2020 were considered, both to save resources in generating recommendations and to provide more statistical significance to future results.

Although the experiment groups will be only composed of active customers, the purchase history from the inactive customers will be kept for training. The appearance of Covid-19 with its new rules and limitations could have been a force that led to covariate shift; that is, the input dataset changed (Quiñonero-Candela et al. 2009; Shimodaira 2000). In this case, the input dataset is mostly related to the transactions, and Covid-19 might have changed the way and the products the customers bought. However, that was not noticeable because when using the transactions pre-Covid-19 as the training set to generate recommendations, the hit rate at k for the customers pre-Covid-19 and "post"-Covid-19 were very similar. If there was indeed covariate shift, it would be likely that there would be noticeable differences in the test set for the customers before and after the pandemic.

As for the actual process of sampling the customers into the different groups, PlanOut, developed by Facebook to run experiments on its platform ("Facebook/Planout" 2020) was used. Despite its many functionalities, the sampling was done in a uniform way to the four groups according to the customer identifier. After sampling, it was checked if the customers were balanced in regard to the Portuguese variable and, although it was not directly involved in the sampling process, the customers were split evenly, as it can be seen in Table 11, thus it is believed there will be no issues regarding bias stemming from that variable.

Because some customers belonging to the loyalty and novelty groups cannot receive the k tailored recommendations from their assigned strategy, they will be removed from the experiment. Such measure is necessary, as more recommendations (default ones) will be presented to them to fill the remaining recommendation slots. Because of that, if a customer

purchases articles linked to the default recommendations, it signals the recommender may work, but it does not provide any insights regarding the effects of the strategy.

Data and Recommendations Update

It is necessary that the data (new transactions) is updated so that the recommendations are updated too. This is necessary because new customers can become active and new products can be bought, which can help improve the quality of the recommendations, especially those of the NCF algorithm. Due to the nature of the CF algorithms, one known limitation is the cold-start problem due to the lack of past information about customers and/or products in the system. In the first case, the absence of data about a customer prevents the system from generating tailored recommendations to unseen customers. The second case can arise when new products are added to the set of available products, which cannot be recommended if there is no prior information about them.

In this setting, personalized recommendations are only presented to customers responsible for more than 5 purchases, meaning that the new user problem is not directly addressed but it will not arise either. On the other hand, in terms of items, the cold-start problem was not addressed nor mitigated.

Updating the data allows customers which were not being showed personalized recommendations to be showed if they pass the 5 transactions threshold. Also, having more transactions from more customers allows a decrease of data sparsity. Additionally, due to the possibility of new articles being added to the set, updating the data allows for them to be considered and, consequently, recommended.

Due to the relatively low number of customers, items, and transactions, updating the data and generating recommendations can be done on a weekly basis. Such conditions allow for the NCF model to be retrained every week with the whole dataset: the training time only increases

about 20 to 30 seconds every week. Such practice would not be advisable, or even possible, on a larger scale, that is, if we were considering a larger store with more customers, products, and transactions, or multiple stores.

Deployment

After processing and curating the data, the recommendations should be displayed to the store's customers in one page of the PD&Go Nova app, which is necessary for the customers to enter the store and add products to their baskets. That platform is the most desirable due to the compulsory nature it has in the customers' journey in the store. Additionally, in case the recommender and the recommendations were to generate a new and optional app, the impact of the recommendations would be lowered, as the customers would have the onus of downloading it and only if they desired.

By placing the recommendations in an existing app which needs to be used, and making them visible as a default option removes friction from this process and makes the customers more likely to pay attention to them. Such choice is derived from one study in which it was observed that countries with "'Opt Out' Policies Increase Organ Donation". This happened not because the people were unwilling to donate them, as that is unknown, but rather because the default option was that they were volunteers for the organ donation (Davidai, Gilovich and Ross 2012). As in that case, by taking advantage of people's inertia of changing settings, it is possible to increase the impact of some measures due to predefined options.

However, if the app was unable to accommodate the recommender and to keep the friction of the process as low as possible, an alternative would be that the app redirected the customer into a web page where he could see and interact with the recommendations. In order to turn such a scenario into reality, one could use a combination of FastAPI, Streamlit and Docker

(Shaji 2020). With those tools, it was possible to generate a prototype that served as a demonstration of how the recommender could be deployed.

With FastAPI, one would develop the backend of the infrastructure, which could be used to handle data preprocessing, splitting the customers into the experiment groups, and generating the recommendations. However, to simplify the process and make the backend faster, the only necessary method is a Get, which implies that the backend is only responsible for returning the generated recommendations, while all the previous process until the recommendations' generation is handled by another tool (like Jupyter Notebook or a Python Script). In the developed prototype, get_recommendations_by_customer was the name of the method and it would accept as parameters a customer identifier (provided by the app); a recommendation strategy; and the number of recommendations to be displayed ("Fastapi" 2020; Shaji 2020).

As all the recommendations were previously generated, combined together into a single dataset (a sample from the December 27, 2020 can be seen in Figure 7), and turned into a file, such a file needs to be provided to the backend. Then the Get method would return the top k recommendations for the customer using all the logic described for each recommendation strategy as a JSON file. Unfortunately, as not all products had a picture, these were not included. In Figures 8 and 9, one can see the Swagger of the get_recommendations_by_customer method, with an example of 4 recommendations for a customer belonging to the related group.

With Streamlit, one can generate a web page that would show the customers their recommendations. As it works as a mixture of HTML/CSS and Python, the generated page can be adjusted to the language the customer has chosen to use in his app. In addition to that, it is possible that the customers interact with the recommendations. In the prototype, the customers could check the boxes with the articles they liked and, at the bottom of the page, they could submit it, which would store the list of the items they were interested in. In Figure 10, it is possible to see how the web page could look on a smartphone. In it, there are two

recommendations, with space for the respective images, the "interest" checkboxes, and the "submission" button (Shaji 2020; "Streamlit 0.73.0 Documentation" 2020).

Recommender System Evaluation

The final step for this project with the Design Science Research Process is evaluating the recommender to assess how it behaved in addressing the problem identified in the beginning and how well it achieved the proposed goals of increasing sales, increasing the number of unique products acquired by customer, and of influencing the customer's decisions. Unfortunately, as the recommender has not yet been deployed, the results of some metrics discussed below can only be taken from a theoretical standpoint and not from real data.

Using the experiment groups as defined above, evaluating the recommender can be done from different perspectives. As the recommendations are to be presented on a weekly basis, the evaluation should also be considered in those terms. One criterion to evaluate the recommender and its strategies is to assess how many different items are acquired by each customer, on average. Although this metric is not directly related to the "accuracy" of the algorithm, it should be considered because the customers are inserted in a physical setting in which they are responsible for picking up the items they want to purchase. The rationale to include this metric is that, while searching for recommended products, the customers are likely to look at other articles, which they had not thought about until that point, which they may end up buying. Furthermore, such metric can aid distinguishing the recommendation strategies leading to a larger number of different products being purchased.

Before the deployment of this technology, the discussion of the results from this metric can only be done in an abstract manner, as there is no indication of how the customers will behave. For instance, it is not possible to predict if the customers will buy a more diversified set of products because they are being recommended articles they have never purchased before but

that other customers like them had acquired (related strategy) or because they are willing to try new products which were randomly recommended (control group). However, it is also possible that the customers start buying a narrower set of products because they find the recommendations not relevant or that they do not bring them more value. However, as already mentioned, while the recommender is not deployed, these scenarios are only speculative ones, and "true" conclusions can only be drawn after data is collected while the recommender system is being used.

Another metric that can be taken into consideration when evaluating the recommender is to look at the revenue each customer generates, on average. The logic and factors behind this metric are similar to that of the previous one, but its goal is different, as it allows to assess which recommendation strategies lead to higher revenues. Again, predicting results from this metric before the deployment can only be a mental exercise. As the conclusions stem from theoretical scenarios, and not real data, further actions shall not be taken because, currently, there are no signals on how the recommender or the strategies would impact the customers and, consequently, the revenues.

Thus, despite the loyalty and novelty strategies suggest products the customers have previously acquired, one cannot say if these clients will end up purchasing more than those belonging to the control or the related groups. Furthermore, as it is not possible to predict the behavior of the customers toward recommendations of items they have not purchased or toward random ones, one cannot say if they will end up spending more, because those are products they have not considered before, or less, since they do not find the recommendations valuable or interesting.

In addition to the metrics enunciated, whose results before the deployment cannot be discussed to draw conclusions, there is a more traditional metric that can be used to judge the recommender: the hit rate, described before (Li 2019). In the previous cases, discussing the

results could only be a mental exercise about what to expect in the future. However, for the hit rate, it is possible to draw conclusions based on the results from historical data.

In this case, the hit rate 6 was utilized, as it was the number of tiles with recommendations on the home page of the PD&Go Nova app. The method followed to calculate the hit rate was the same discussed above, with the necessary adjustments for each of the defined strategies. Specifically, for the novelty and loyalty strategies, the top 6 recommendations were those items with the highest number of purchases, sorted in descending order; for the related strategy, the top 6 were those with the highest "prediction", sorted in descending order. As the recommendations are generated for a week, the test set for each set of recommendations are the transactions that occurred in the week after their generation. As the starting date of the recommendations was November 15, 2020, the test set for this set of recommendations included the transactions ranging from November 16 until December 27, split by week.

Table 12 displays the hit rate at 6 for the 4 experiment groups, from November 15 until December 27, 2020. From the table, one can see that the hit rate at 6 for the control group is the lowest, being always below 1%. The related strategy is the treatment group with the lowest hit rate at 6, with around 3%. The novelty group is the one with the second-highest levels of hit rate at 6, ranging from around 6.71% up to about 14.31%. The loyalty group is the one with the highest hit rate at 6 values, ranging from about 28.28% up to around 36.70%.

Although the customers' behavior may change after the deployment of the recommender and despite the reduced period under analysis, personalized recommendations, regardless of the chosen strategy, appear to yield better results in what concerns the hit rate at 6, as the values for all the treatment groups outperformed that of the control group.

Always bearing in mind the caveats mentioned in the paragraph above, one could believe that the simpler strategies (loyalty and novelty) have a higher hit rate at 6. However, it is important to remember that the recommendations presented to the related group are only of

items the customers have not purchased, while for the other two treatment groups, the recommendations are items they have acquired before.

One preliminary conclusion is that comparing the two loyalty strategies (since the novelty strategy follows the same principles as the loyalty one), having a time restriction hinders the hit rate at 6 performance. As both strategies try to gauge the importance of the customers being reminded of the articles they have already purchased, only after the recommender is deployed is it possible to understand the true impact. However, even before deployment, one could predict that up to a certain level of k, the hit rate at k for the loyalty strategy would keep outperforming the novelty one, due to the time restriction imposed during the setup.

Another conclusion to which one could be led is that the algorithms based on heuristics are better than the one which uses machine learning techniques, especially if the only metric under consideration is the hit rate at k. However, it is important to consider the differences between the type of recommendations. Hence, before making such claims, it is necessary to let the recommender be deployed so that "true" conclusions can be drawn.

From the situation above, it is noticeable how evaluating the recommender, and the strategies with only one metric can lead one to sub-optimal conclusions. For that reason, one should consider this set of metrics to make the best decisions and to prove this point, it is necessary to resort to unobserved, yet possible, scenarios. When considering the set of unique products acquired by the customers as the only metric, it is possible that the group assigned to the related strategy will have the highest number of different products acquired per customer because the recommendations that the group sees are all of items not previously purchased. However, if the hit rate at k is added to the equation, the scenario changes, as that strategy is the one with the lowest values. In this scenario, picking one of the two metrics would be a mistake instead of analyzing both within the context. It is possible one could be before a case in which the recommendation in week 1 was a product the customer did not use to buy. In

addition to the usual set of items he already purchased, such a recommendation might lead him to repeat that purchase across many weeks after the customer saw the recommendation. With the hit rate, only in week 1 will that recommendation be acknowledged as a hit, and will the strategy be "rewarded." To make the analysis more complete and "fair," adding more perspectives is advisable. In this specific case, in addition to the usual amount of revenue that customer generated, there was an increase due to that new product. Without the number of different articles acquired and the revenues, this effect would not be recognized, and the strategy could be deemed not worthy of being used in the future.

These are the three different metrics that can be used to evaluate the recommender and to evaluate different aspects of the customers' behavior. Despite the discussion of the speculative scenarios associated with the first two metrics and the analysis of the preliminary results from the hit rate, more definitive conclusions can only be taken after the recommender is inserted in the app and some time passes in order to provide them with more robustness. Once more, it is fundamental that the analysis is not solely focused on one metric at a time but that the three are taken together along with the context in which they are inserted.

Limitations and Challenges

Despite the many potentialities of this project, there are equally some limitations and challenges which need to be considered.

The first one is that there are no strategies involving content-based filtering algorithms or hybrid ones. Concerning the content-based filtering algorithms, it was not possible to implement one. It was not because it was unfit for the context, as it would be appropriate to study how that strategy would perform in comparison with the others. The reason why it was not developed was that there was not enough data and/or features that would allow its creation.

Regarding the hybrid recommender, despite the tendency to perform better than both the content-based filtering and the collaborative filtering algorithms, it could not be correctly

implemented. LightFM was the model that was attempted to be turned into a recommender, but it did not perform as required. It was supposed to generate a list of recommendations which would be sorted according to the corresponding probability/likelihood of the customers to purchase them ("Microsoft/Recommenders" 2020; "Lightfm - Lightfm 1.15 Documentation" 2020). It was observed that the top recommended articles were products which had been previously acquired. The issue, however, was that products the customers had never acquired had no assigned probability or likelihood to be acquired. In other words, after the products which had been previously purchased, the following recommendations would have as much quality as those generated for the control group, as they would be randomly sorted. Therefore, having a dedicated experiment group for an algorithm with this behavior would, essentially, be a diversion of time (it would be necessary to prepare the data and train the algorithm); computing power (to curate the data and train the model); and statistical significance (as there would be less customers receiving tailored recommendations from the other strategies).

Another limitation of this project is also a factor that can increase its success: the fact that the customers need to physically go to the store and grab the products from the shelf. Since the reasons why such factor can be considered a boosting aspect were already mentioned, they will not be reiterated. On the other hand, it can be considered a hindrance because the customers may not be able to find where the recommended articles are. Following that line of thought, if the customers are unable to find their desired items, they will not purchase them, which hinders the performance of the recommender.

Such factor is also related to the search costs the customers incur when looking for the recommended products. If it is true that the recommender can be useful for the customers to discover items which they did not know about or that they had never thought of, it is also true that if the search costs are too high, i.e., the articles are not easily accessible, some recommendations may never become a sale.

In addition to the already identified limitations, it is also necessary to consider how to attribute sales directly to the recommendations. As it happens with ads, especially in an online setting, it is very difficult to attribute a sale to a specific ad. As the purchasing journey starts when the need for a product is born, and it does not end right after the purchase is made (Court et al. 2009), due to the display of different ads in more than one place during the journey, it is not easy to directly assign a sale to a specific ad. To prove that, such attribution can be done in more than one way (Gupta and Davin 2015). Despite A/B testing being the gold standard and one of the most reliable ways to attribute actions to certain circumstances, it only allows the identification of the effects of the recommendation strategies. Nonetheless, as the customers do not interact with the recommendations, by clicking them, for instance, which could be a feature to add to the app in the future, that would reveal the customer's interest, it is not possible to know if the recommendations had a role in influencing the customer's purchase decision.

Furthermore, it is necessary to remember the importance of having an explanation of how the recommendations are generated. On the one hand, it is possible to argue that the ideal number of recommendations being presented for all the recommendation strategies is chosen from an A/B testing perspective or from heuristics. The same rationale can be applied for the time restrictions of the novelty algorithm. When considering the loyalty, the novelty, and the control group, one can easily explain how the recommendations are generated. However, although one can explain the architecture and how the NCF algorithm works, that model is considered a black box model: after the input of the data, one only sees its output without being truly able to understand what the process was that led to the outcome. Currently, explaining how the algorithms work is of the utmost importance, as it is necessary that the customers keep trusting this technology and the company so that they keep using it and they keep being customers at PD&Go Nova.

One more limitation which needs to be considered is the covariate shift. This phenomenon was not noticeable, as the hit rate at k using the same recommendation strategies for a different set of customers ("old," the customers from the last academic year, and the "new" ones, those active during the semester of Fall 2020/21) was very similar. Although it did not happen from the previous academic year to this year, one cannot guarantee that it will not be observed in the future, as it is possible that the customers' behavior in the store may change, which would mean that the purchase history of previous customers will not be helpful to generate meaningful and interesting recommendations for new customers.

The last challenge which needs to be noted is a metric that would aid in evaluating the performance of the recommender, which is spatial distribution. To see the spatial distribution of the sales, concerning the number of transactions, or the number of products sold, or just the sales value, a planogram (document/file which displays how a store and its items are organized) was necessary. Although there is a planogram, when merging it with the transactions' data, it resulted in a large number of missing values (around 80%). This situation happened because the planogram had only about 1,000 unique products, while more than 2,000 were present in the transactions' dataset. As only about 50% of the transactions' unique items are represented in the planogram, representing only around 20% of all the transactions, the analysis of values associated with a spatial distribution metric would not provide valuable insights that could be turned into future actions.

Future steps

Besides the deployment of the recommender engine, which will allow a proper evaluation of the recommender by itself along with the different recommendation strategies, improving the quality of the data regarding the products' features is of paramount importance because it

would allow the development of a content-based recommender. Having such an algorithm would add yet another perspective to the analysis and fill some gaps which were not addressed.

Moreover, updating the planogram is fundamental; otherwise analyzing the spatial distribution of sales will not be possible. Without this metric and its analysis, it is not possible to know if the store is organized and is displaying the products in an optimized way, being it in terms of sales, number of unique items acquired per customer, or any other relevant metric. Without it, it is not possible to challenge the *status quo* and to infer if it is indeed the superior way to organize and display the products or if there is a better way.

Linked to the layout and the organization of the products, one could implement the logic behind the recommendation strategies of the app in a complete offline setting. The decision of which recommendation strategy(ies) should be transposed from the online to the offline (and their order if they cannot all be applied at the same time) could stem from the analysis of the existing recommendation strategies set in place, which are the ones described in this paper. With this procedure in mind, one can understand why covering as much as possible of the recommendation strategy spectrum is important and why having a content-based recommender would matter. Despite its differences, one could take Amazon as an example, with its online platform, to which the proposed recommender engine would very roughly correspond, and its physical store where such techniques are applied and tested, whose correspondence would be the physical PD&Go Nova store.

Furthermore, the recommendations presented to the related group are only of products the customers have not bought before, which was done with the purpose of isolating the effect of solely recommending such articles. However, since the NCF algorithm can generate recommendations of items that the customers have previously acquired, it can be considered to be an option for the recommender, as it is able to include the effect of repeated purchases and

use the Collaborative Filtering "setting" to take advantage of similarities between customers to suggest products not yet purchased by them.

Lastly, this project can be the first step for the physical retail industry to reinvent itself, as techniques and strategies used in the online setting can be ported into the offline one, which can impact how the stores are managed, both in terms of their layout (the shelves and the products in them) and in terms of the set of products and their quantities in the store. Moreover, this recommender can also be disruptive in terms of the customer experience, as it becomes hybrid, in the sense that their journey can include aspects from both the online environment (with the recommendations, and possibly more and more information about the products) and from the offline settings (as they will still go to the stores and have the ability to touch and feel the products). Again, with the possibility of the customer experience changing, companies from this industry and managers have are new opportunities to seize and challenges to address.

Conclusion

To sum up, as recommenders have been thriving and delivering great results in e-commerce settings, this paper proposes implementing one in the physical retailing industry. It can serve the purpose of increasing sales, the number of unique products purchased by the customers, as well as understanding the customers and their shopping behavior better. To do so, the customers were split randomly and uniformly into four different groups to test three different recommendation strategies (loyalty, novelty, and related) against a control group. After the recommender is deployed in the PD&Go Nova app and the customers use it, the recommender can be evaluated afterward, using metrics such as the number of unique items acquired by the customers, the sales values, and the hit rate at k. After its analysis, more steps can be taken to improve both the recommender and how the physical retail stores do business.

References

Amazon.com, Inc. 2020. "Amazon.Com Announces First Quarter Results". https://ir.aboutamazon.com/news-release/news-release-details/2020/Amazoncom-Announces-First-Quarter/.

Amazon.com, Inc. 2020. "Amazon.Com Announces Second Quarter Results". https://ir.aboutamazon.com/news-release/news-release-details/2020/Amazon.com-Announces-Second-Quarter-Results/.

Amazon.com, Inc. 2020. "Amazon.Com Announces Third Quarter Results". https://press.aboutamazon.com/news-releases/news-release-details/amazoncom-announces-third-quarter-results/.

Ariely, Dan. 2000. "Controlling The Information Flow: Effects On Consumers' Decision Making And Preferences". Journal Of Consumer Research 27 (2): 233-248. doi:10.1086/314322.

Brynjolfsson, Erik, and Michael D. Smith. 2000. "Frictionless Commerce? A Comparison Of Internet And Conventional Retailers". Management Science 46 (4): 563-585. doi:10.1287/mnsc.46.4.563.12061.

Brynjolfsson, Erik, Yu (Jeffrey) Hu, and Mohammad S. Rahman. 2009. "Battle Of The Retail Channels: How Product Selection And Geography Drive Cross-Channel Competition". Management Science 55 (11): 1755-1765. doi:10.1287/mnsc.1090.1062.

"Collaborative Filtering". 2020. Google Developers. https://developers.google.com/machine-learning/recommendation/collaborative/basics.

Court, David, Dave Elzinga, Susan Mulder, and Ole Jørgen Vetvik. 2009. "The Consumer Decision Journey". Mckinsey & Company. https://www.mckinsey.com/business-functions/marketing-and-sales/our-insights/the-consumer-decision-journey.

Davidai, Shai, Thomas Gilovich, and Lee D. Ross. 2012. "The Meaning Of Default Options For Potential Organ Donors". Proceedings Of The National Academy Of Sciences 109 (38): 15201-15205. doi:10.1073/pnas.1211695109.

"Facebook/Planout". 2020. Github. https://github.com/facebook/planout.

"Fastapi". 2020. Fastapi.Tiangolo.Com. https://fastapi.tiangolo.com/.

González-Fierro, Miguel. 2018. "Neural Collaborative Filtering On Movielens Dataset". Github. https://github.com/microsoft/recommenders/blob/master/examples/00_quick_start/ncf_movielens.ipynb.

Graham, Scott. 2018. "Notebooks/02_Model/Ncf_Deep_Dive.Ipynb · Gitee 极速下载 /Recommenders - Gitee.Com". Gitee. https://gitee.com/mirrors/recommenders/blob/eeec884530bf08eccf6aae6ff692207422b2de1d/notebooks/02_model/ncf_deep_dive.ipynb.

Gupta, Sunil, and Joseph Davin. 2015. "Marketing Reading: Digital Marketing". Core Curriculum Readings Series. Boston: Harvard Business Publishing 8224.

He, Xiangnan, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. "Neural Collaborative Filtering". WWW '17: Proceedings Of The 26Th International Conference On World Wide Web, 173-182. doi:10.1145/3038912.3052569.

Hevner, Alan R., Salvatore T. March, Jinsoo Park, and Sudha Ram. 2004. "Design Science In Information Systems Research". MIS Quarterly 28 (1): 75-105.

Hinz, Oliver, and Jochen Eckert. 2010. "The Impact Of Search And Recommendation Systems On Sales In Electronic Commerce". Business & Information Systems Engineering 2 (2): 67-77. doi:10.1007/s12599-010-0092-x.

Li, Susan. 2019. "Evaluating A Real-Life Recommender System, Error-Based And Ranking-Based". Medium. https://towardsdatascience.com/evaluating-a-real-life-recommender-system-error-based-and-ranking-based-84708e3285b.

Lieber, Ethan, and Chad Syverson. 2012. "Online Versus Offline Competition". Oxford Handbooks Online. doi:10.1093/oxfordhb/9780195397840.013.0008.

"Lightfm — Lightfm 1.15 Documentation". 2020. Making.Lyst.Com. https://making.lyst.com/lightfm/docs/lightfm.html.

"Microsoft/Recommenders". 2020. Github. https://github.com/microsoft/recommenders.

Pathak, Bhavik, Robert Garfinkel, Ram D. Gopal, Rajkumar Venkatesan, and Fang Yin. 2010. "Empirical Analysis Of The Impact Of Recommender Systems On Sales". Journal Of Management Information Systems 27 (2): 159-188. doi:10.2753/mis0742-1222270205.

Peffers, Ken, Tuure Tuunanen, Charles E. Gengler, Matti Rossi, Wendy Hui, Ville Virtanen, and Johanna Bragge. 2006. "The Design Science Research Process: A Model For Producing And Presenting Information Systems Research". DESRIST International Conference On Design Science Research In Information Systems And Technology, 83-106.

Quiñonero-Candela, Joaquin, Masashi Sugiyama, Anton Schwaighofer, and Neil D. Lawrence. 2009. Dataset Shift In Machine Learning (Neural Information Processing Series). MIT Press.

Schafer, J. Ben, Joseph A. Konstan, and John Riedl. 2001. Data Mining And Knowledge Discovery 5 (1/2): 115-153. doi:10.1023/a:1009804230409.

Schwartz, Barry. 2015. The Paradox Of Choice: Why More Is Less. New York, N.Y.: Harper Perennial.

Shaji, Amal. 2020. "Serving A Machine Learning Model With Fastapi And Streamlit". Testdriven.Io. https://testdriven.io/blog/fastapi-streamlit/.

Sharma, Abhishek. 2019. "Neural Collaborative Filtering". Medium. https://towardsdatascience.com/neural-collaborative-filtering-96cef1009401.

Sharma, Sagar. 2017. "Epoch Vs Batch Size Vs Iterations". Medium. https://towardsdatascience.com/epoch-vs-iterations-vs-batch-size-4dfb9c7ce9c9.

Shimodaira, Hidetoshi. 2000. "Improving Predictive Inference Under Covariate Shift By Weighting The Log-Likelihood Function". Journal Of Statistical Planning And Inference 90 (2): 227-244. doi:10.1016/s0378-3758(00)00115-4.

Shaji, Amal. 2020. "Serving A Machine Learning Model With Fastapi And Streamlit". Testdriven.Io. https://testdriven.io/blog/fastapi-streamlit/.

Zeballos, Eliana. 2020. "USDA ERS - More Americans Spend More Time In Food-Related Activities Than A Decade Ago". Ers.Usda.Gov. https://www.ers.usda.gov/amber-waves/2020/april/more-americans-spend-more-time-in-food-related-activities-than-a-decade-ago/.

Appendix

Table 1 - Descriptive Statistics for the curated dataset grouped in transactions and summed

	QTE	VALOR_PVP
count	327668.000	327668.000
mean	2.238	2.654
std	2.517	3.373
min	0.000	-5.120
25%	1.000	0.790
50%	2.000	1.840
75%	2.000	3.380
max	196.000	248.390

Observation: the minimum values of QTE and VALOR_PVP refer to products inserted in promotion settings.

Figure 2 - Sample of the Transactions' Dataframe

	ID_DIA	ID_TXS	ID_LOJA	ID_POS	ID_OPERADOR	ID_HORA	N_TRANSACCAO	ID_EAN	ID_ARTIGO	UMB	...	NUM
0	03-10-2019	2019100340900022222000000685	4.09	2	2222	17	685.000	2000001533161	254381	UN	...	
1	03-10-2019	2019100340900114444000000719	4.09	11	4444	18	719.000	2671878000000	879462	UN	...	
2	04-10-2019	2019100440900103333000000872	4.09	10	3333	12	872.000	2672070000000	896315	UN	...	
3	03-10-2019	2019100340900114444000000737	4.09	11	4444	18	737.000	2000003474424	895831	UN	...	
4	03-10-2019	2019100340900125555000000650	4.09	12	5555	15	650.000	2634505000000	252724	UN	...	
...	...	...	...	...	...	...	...	...	...	...	...	...
452208	12-03-2020	2020031240900970708000005146	4.09	97	708	16	5.146	5601009928910	563519	UN	...	
452209	12-03-2020	2020031240900970708000005146	4.09	97	708	16	5.146	2000001878248	256272	UN	...	
452210	12-03-2020	2020031240900970708000005146	4.09	97	708	16	5.146	5601009988648	865448	UN	...	
452211	12-03-2020	2020031240900970708000005146	4.09	97	708	16	5.146	5601009972852	357370	UN	...	
452212	19-05-2020	2020051940900960707000003556	4.09	96	707	8	3.556	5601009951222	598142	UN	...	

452213 rows × 46 columns

Figure 3 - Sample of the Market Structure Dataframe

	COD_AREA	DESC_AREA	COD_DIVISAO	DESC_DIVISAO	COD_FAMILIA	DESC_FAMILIA	COD_CATEG	DESC_CATEG	COD_SUB_CATEG
0	1004	BEBIDAS	2021	REFRIGERANTES	3059	BEBIDAS REFRESCANTES	4236	ICE TEA	6326
1	1001	MERCEARIA + PET FOOD	2002	ALIMENTACAO CORRENTE	3002	ALIMENTACAO PEQUENO ALMOCO	4010	CEREAIS DE PEQUENO ALMOCO	9754
2	1004	BEBIDAS	2303	AGUAS	3058	AGUA	4228	AGUAS LISAS	9779
3	1013	MEAL SOLUTIONS	2006	TAKE-AWAY	3237	CHURRASCARIA	4837	SALSICHA	8687
4	1001	MERCEARIA + PET FOOD	2002	ALIMENTACAO CORRENTE	3002	ALIMENTACAO PEQUENO ALMOCO	4010	CEREAIS DE PEQUENO ALMOCO	9256
...	...	...	...	...	...	...	...	...	...
2247	1004	BEBIDAS	2022	CERVEJA	3264	SIDRA	4981	SIDRA	9455
2248	1002	PERECIVEIS ESPECIALIZADOS	2010	PADARIA/PASTELARIA	3307	CAFE E BOLOS	4973	BEBIDAS	8160
2249	1003	PERECÃ VEIS NÃ)O ESPECIALIZADOS	2016	CONGELADOS	3045	SOBREMESAS E GELADOS	4174	GELADOS	6015
2250	1003	PERECÃ VEIS NÃ)O ESPECIALIZADOS	2016	CONGELADOS	3044	COMPONENTES REFEICOES CONGELADAS	4167	PREPARADO CARNE E PEIXE CONGELADO	5983
2251	1006	PRODUTOS PESSOAIS	2031	HIGIENE PESSOAL	3077	PRODUTOS PARA CABELO	4306	CHAMPO	6598

2252 rows × 14 columns

Figure 4 - Sample of the Customers' Dataframe

	ID	USERID	NIFCOUNTRY	PHONECOUNTRY	CUSTOMERGUID	CUSTOMERSTATUSID	LOCALE	CUSTOMERFILEID	REFERRALCODE	DEV
0	53663	18869	NaN	351	02fea7f8-3b5f-4edc-b3c6-ee0ceb2a8fcf	2	pt-PT	?	Tnwt5	5C3E79D2-4B30-E7876A8
1	1148	2328	NaN	351	2f65132e-8cfa-47b2-acbf-90cf7d56c4ec	2	pt-PT	?	9LWsS	A283374A-I 494B-A290D72
2	2555	3735	NaN	351	7ba44f24-13cd-44bd-bcec-af913241628f	2	en-GB	?	iu0TE	FA3261CC-477A-0943F05I
3	44167	9348	NaN	351	db2b5da3-9a8f-45d8-ab30-77c322f58d71	2	pt-PT	?	NUbLB	ba3de8b7b67
4	47307	12491	NaN	351	8b42aa34-e6bf-40de-b473-7fa51868007c	2	pt-PT	?	OLkTJ	3c0f5059dab
...	...	...	...	...	...	...	...	...	...	
21984	71274	27834	NaN	351	7cca14f5-01cd-4dda-94af-1120bdc6ee4a	2	pt-PT	?	kuoRL	10325DFA-4046-C52D511
21985	49526	14718	NaN	351	e0a02448-9683-408f-b794-81f59cf13d50	2	pt-PT	?	Ggdpn	202156db3fa
21986	56750	21978	351.0	351	0ba0dd7b-3e23-402d-8625-cc2b1ffd4fd9	2	en-GB	?	m9QQ2	A97C75F1-45DF-0CAE77A
21987	48153	13338	NaN	351	07f6697f-3223-4e8f-97ce-542eda1660f7	2	pt-PT	?	jVdwV	218C72EE-4AB7-65F123F2
21988	47213	12397	NaN	351	eccf106f-8fa9-4d8d-a21a-4fcf687efb93	2	pt-PT	?	79YV8	5a66ade96af

21989 rows × 29 columns

Table 2 - Columns used from the three dataframes and respective description

Column Name	**Description**
ID_ARTIGO	Identification number of the product
QTE	Number of products acquired
CARD_NUMBER	Hashed identification of the customer
HORA_TXS	Time when the transaction occurred (hours, minutes, seconds)
ID_DIA	Date when the transaction happened
VALOR_PVP	Sales value (price paid by the customers)
COD_ARTIGO	Identification number of the product (equal to ID_ARTIGO)
DESC_ARTIGO	Name of the product
USERID	Number identification of the customer
LOYALTYCARDNUMBER	Hashed identification of the customer (equal to CARD_NUMBER)
CUSTOMERGUID	Hashed identification of the customer
NIFCOUNTRY	Country indicator of the customer's fiscal number
PHONECOUNTRY	Country indicator of the customer's phone number
LOCALE	Language of the customer's app

Table 3 - Creation Process of the Portuguese Column

Step	Description	Reasoning
Step 1	Replace the missing values from the NIFCOUNTRY column with zeroes.	
Step 2	If the NIFCOUNTRY value was 351 (country code indicator of Portugal), the customer was considered Portuguese, if not it would be considered Non-Portuguese.	As the fiscal number is the most likely indicator of a person's nationality, if a customer had a fiscal number, it would be considered Portuguese or Non-Portuguese.
Step 3	If the customer did not have a fiscal number associated, it was looked at the PHONECOUNTRY column: if the value was different from 351, it would be considered Non-Portuguese. If it was 351 it would pass to the next step.	Many Non-Portuguese kept their phone numbers due to new Roaming policies in Europe, but there are customers who did not come from Europe and that chose to change their phone number.
Step 4	If the language of the app was Portuguese, they were considered Portuguese and if it was English they would be considered Non-Portuguese.	The majority of this cases are customers which do not come from former Portuguese colonies, hence only these will be misclassified as Portuguese (if they have the app in English) and Portuguese customers will be considered Non-Portuguese if they do not have a fiscal number associated and the app in English

Observation: as this information is not explicit and it is just a proxy, there may be customers who are misclassified. Either Non-Portuguese ones who were considered Portuguese (which can be the case of customers from former Portuguese colonies which have a Portuguese phone number and their app in Portuguese as well) or Portuguese customers who were considered Non-Portuguese (customers without an associated fiscal number, a Portuguese phone number but the app in English.

Table 4 - Process to Fix the ID_ARTIGO column (products with more than one identifier

Step	Description
Step 1	Identify the products with the same name (DESC_ARTIGO) but with more than one identifier (ID_ARTIGO).
Step 2	The products whose identifier changed in a clean way (i.e., one identifier until a certain day and another from the following day onward) were immediately replaced. The other cases moved to the next step to manual analysis.
Step 3	There were cases alike the ones described in Step 2 but whose identifiers co-existed during 1-2 days. These products had their identifiers also standardized. If the situation was different, they were considered exceptions. Until December 27, 2020, there were 2 exceptions.

Table 5 - Process to identify and remove customers responsible for less than 5 transactions

Step	Description
Step 1	Select the lines corresponding to each customer
Step 2	Get the number of different values of the full_transaction_date column (remember that the dataset does not have transactions per line, but rather products)
Step 3	Identify the customers with less than 5 transactions and remove them

Figure 5 - Sample of the curated dataset

	ID_DIA	ID_ARTIGO	QTE	VALOR_PVP	CARD_NUMBER	HORA_TXS	DESC_ARTIGO	USERID	NIFCC
0	03-10-2019	254381	1.0	0.58	883FDDD04CD505A47513281DE203C11B920D7FEB59BEB9...	174524	MERENDA MISTA 95 G	2814.0	
1	03-10-2019	879462	1.0	2.99	2BA0673C0E7E92C045B6D8D03B01B2ACAE811A76E3241F...	180538	SANDES DE PRESUNTO, BRIE E COMPOTA	1521.0	
2	04-10-2019	896315	1.0	1.99	8A1390B098CA94820130AD3C8D97022E642D61B10AE2B4...	123521	MELOA CANTALOUPE CUBOS LAB	1943.0	
3	03-10-2019	901241	1.0	0.60	2BA0673C0E7E92C045B6D8D03B01B2ACAE811A76E3241F...	184907	AMERICANO	1521.0	
4	03-10-2019	10005260	1.0	1.99	A7350A09D86AE924AA3ECDFC303C2B95199AF33346A9ED...	191824	FRANGO ASSADO METADE	3012.0	
...	...	...	...	...	...	...	...	...	
620611	23-12-2020	803884	1.0	0.85	ED88D476ED4C88B301E3796244424762D3DB74AC92F65D...	173728	BAT FRITA LISA CAMPONESA PINGO DOCE 170G	27246.0	
620612	23-12-2020	894517	1.0	0.99	ED88D476ED4C88B301E3796244424762D3DB74AC92F65D...	173728	BAT FRT PINGO DOCE SAB TRUFA NEGRA 150GR	27246.0	
620613	23-12-2020	861817	1.0	1.19	ED88D476ED4C88B301E3796244424762D3DB74AC92F65D...	173728	IOG LINDAHLS LÍQ PESSEGO/MARACUJA 330ML	27246.0	
620614	23-12-2020	875117	1.0	1.99	77FAF236448B1DF6FB4F07AA627C73B07CDBE62F73FF48...	181619	GARFOS MADEIRA PURE 25UN - 81189	34208.0	
620615	23-12-2020	875119	1.0	2.99	77FAF236448B1DF6FB4F07AA627C73B07CDBE62F73FF48...	181619	COLHERES MADEIRA PURE 25UN - 81191	34208.0	

620616 rows × 15 columns

Table 6 - Descriptive Statistics of the Curated Dataset

	QTE	VALOR_PVP
count	620616.000	620616.000
mean	1.181	1.401
std	0.946	1.390
min	0.000	-2.200
25%	1.000	0.600
50%	1.000	0.950
75%	1.000	1.990
max	100.000	97.930

Observation: alike Table 1, the minimum values of QTE and VALOR_PVP refer to products in promotion

Table 7 - Descriptive Statistics for the curated dataset grouped by customers and summed

	QTE	VALOR_PVP
count	7006.000	7006.000
mean	104.658	124.146
std	141.615	170.277
min	5.000	1.640
25%	24.000	29.555
50%	56.000	67.960
75%	126.000	150.157
max	1962.000	2362.270

Table 8 - Descriptive Statistics for the curated dataset grouped by products and summed

	QTE	VALOR_PVP
count	2205.000	2205.000
mean	332.532	394.452
std	1466.259	1810.252
min	1.000	0.100
25%	12.000	25.110
50%	55.000	98.640
75%	190.000	268.320
max	42537.000	69006.710

Table 9 - Descriptive Statistics for the curated dataset considering the unique products acquired by the customers

	unique_products
count	7006.000
mean	34.540
std	32.480
min	1.000
25%	13.000
50%	24.000
75%	44.000
max	421.000

Figure 6 - Histogram of Number of Unique Products Acquired per Customer

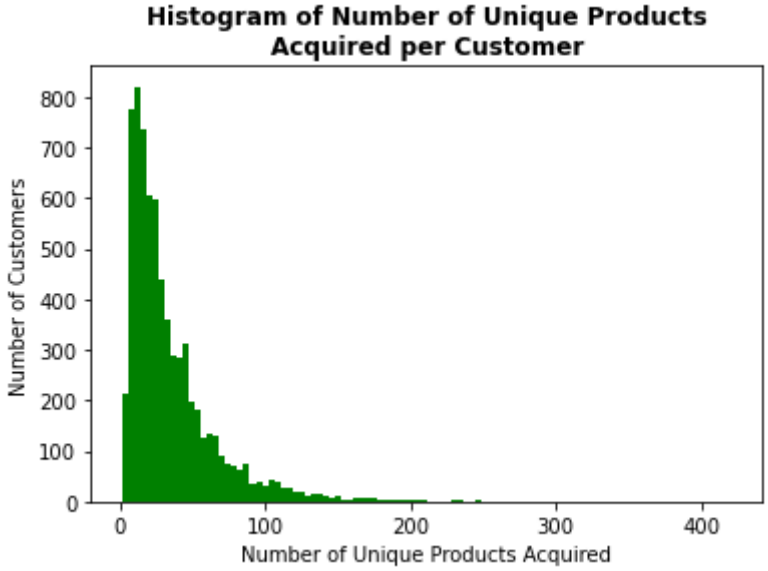


Table 10 - Hyperparameters of the NCF algorithm

Hyperparameter	**Chosen Values**	**Description**
model_type	Neumf	How the matrix factorization will be handled and it can take as values gmf (Generalized Matrix Factorization), mlp (Multi-Layer Perceptron) and neumf (Neural Matrix Factorization, which is a combination of gmf and mlp) (He et al. 2017; González-Fierro 2018; Sharma 2019).
n_factors	4	"It controls the dimension of the latent space," and it refers to the number of layers in the model (He et al. 2017; Graham 2018).
layer_sizes	[16,8,4]	"Sizes of the input layer (and hidden layers) of MLP" (He et al. 2017; Graham 2018).
n_epochs	25	Number of times the “dataset is passed forward and backward through the neural network” (He et al. 2017; Sharma 2017).
batch_size	7500	“Total number of training examples present in a single batch” (He et al. 2017; Sharma 2017).
learning_rate	0.01	“Step size at each iteration while moving toward a minimum of a loss function” (He et al. 2017).

Table 11 - Distribution of the customers according to the experiment group

		USERID
LOCALE	experiment_group	
en-GB	Control	487
	Loyalty	504
	Novelty	475
	Related	475
pt-PT	Control	467
	Loyalty	483
	Novelty	450
	Related	451

Figure 7 - Sample of recommendations from December 27, 2020

	ID_ARTIGO	num_purchases	USERID	group	prediction	CUSTOMERGUID	LOYALTYCARDNUMBER
0	902807.0	6.0	24810.0	novelty	0.0	17c7ea06-cf8f-4bbc-94d9-e58dea53c168	EDF7E01D3CF4D1CA70A0FAB356A80ADD788F25716CAF22...
1	879460.0	6.0	24810.0	novelty	0.0	17c7ea06-cf8f-4bbc-94d9-e58dea53c168	EDF7E01D3CF4D1CA70A0FAB356A80ADD788F25716CAF22...
2	10004898.0	5.0	24810.0	novelty	0.0	17c7ea06-cf8f-4bbc-94d9-e58dea53c168	EDF7E01D3CF4D1CA70A0FAB356A80ADD788F25716CAF22...
3	905807.0	4.0	24810.0	novelty	0.0	17c7ea06-cf8f-4bbc-94d9-e58dea53c168	EDF7E01D3CF4D1CA70A0FAB356A80ADD788F25716CAF22...
4	764091.0	3.0	24810.0	novelty	0.0	17c7ea06-cf8f-4bbc-94d9-e58dea53c168	EDF7E01D3CF4D1CA70A0FAB356A80ADD788F25716CAF22...
...	...	...	...	...	...	...	...
60251	862426.0	0.0	26365.0	control	0.0	11eb4da0-587b-4369-89a4-219738844e09	0930846C2754B438B493690164257489428B7D32E87C12...
60252	263114.0	0.0	26365.0	control	0.0	11eb4da0-587b-4369-89a4-219738844e09	0930846C2754B438B493690164257489428B7D32E87C12...
60253	918410.0	0.0	26365.0	control	0.0	11eb4da0-587b-4369-89a4-219738844e09	0930846C2754B438B493690164257489428B7D32E87C12...
60254	598150.0	0.0	26365.0	control	0.0	11eb4da0-587b-4369-89a4-219738844e09	0930846C2754B438B493690164257489428B7D32E87C12...
60255	897394.0	0.0	26365.0	control	0.0	11eb4da0-587b-4369-89a4-219738844e09	0930846C2754B438B493690164257489428B7D32E87C12...

60256 rows × 7 columns

Figure 8 - FastAPI Prototype Backend Swagger Call

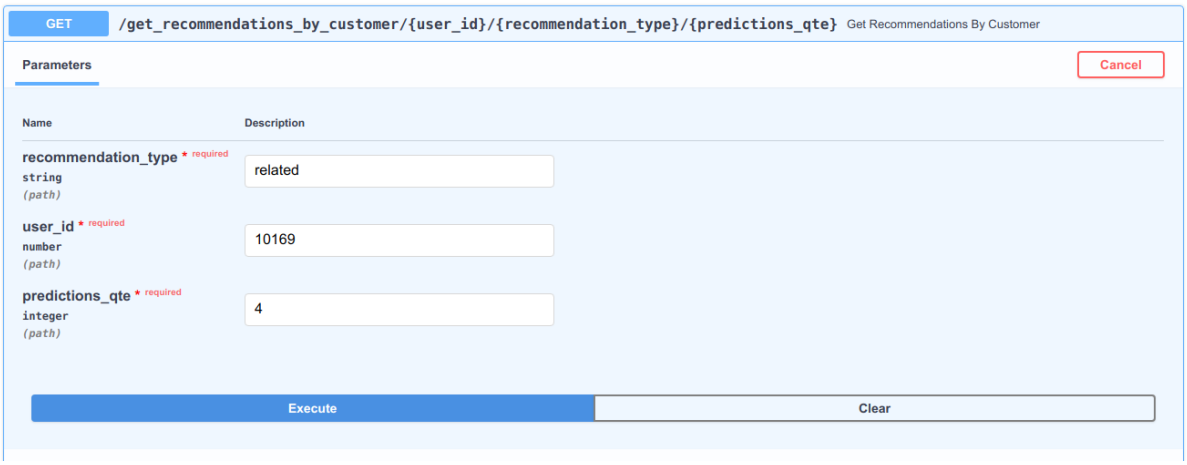


Figure 9 - FastAPI Prototype Backend Swagger Response

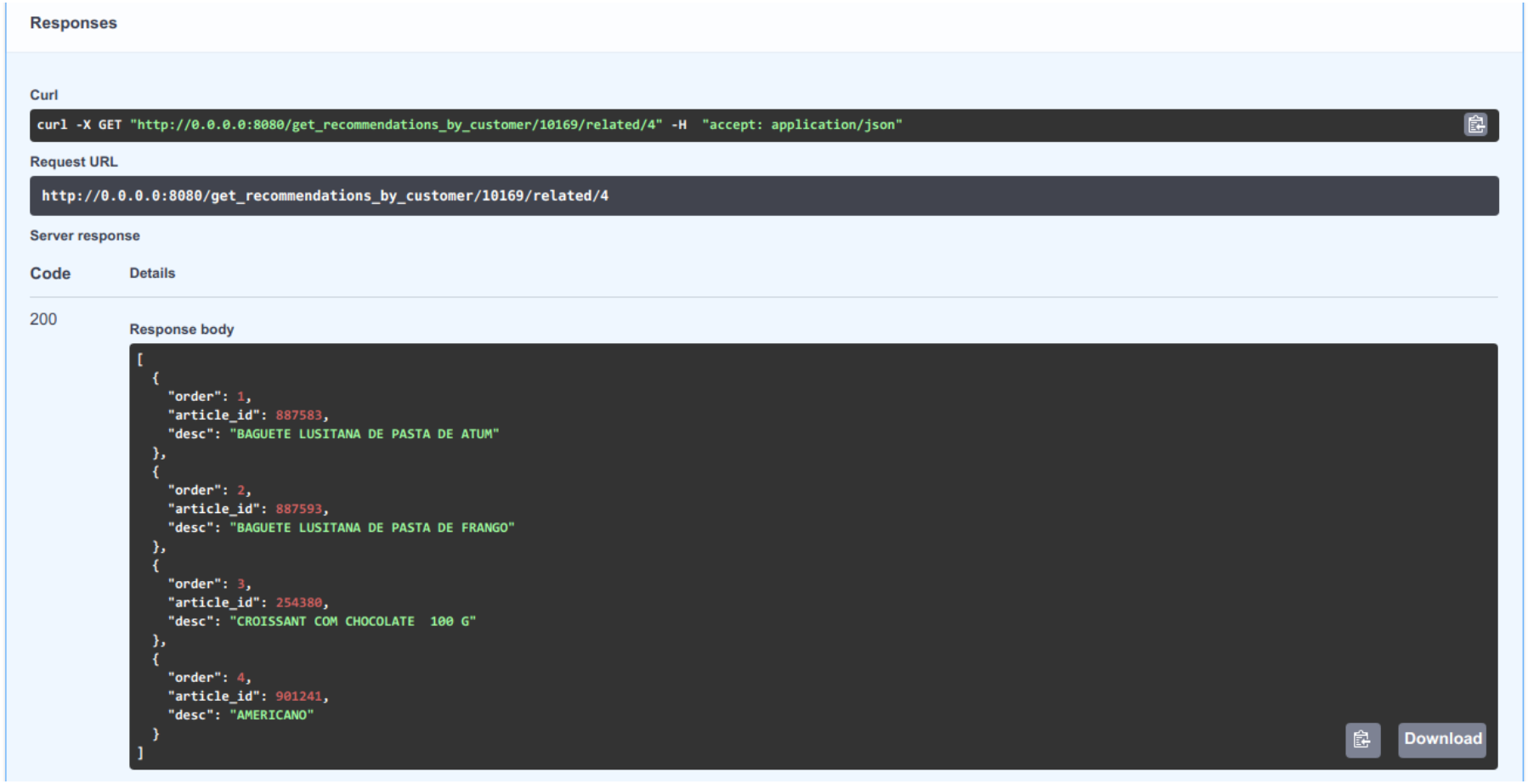


Figure 10 - Smartphone web page prototype created with Streamlit

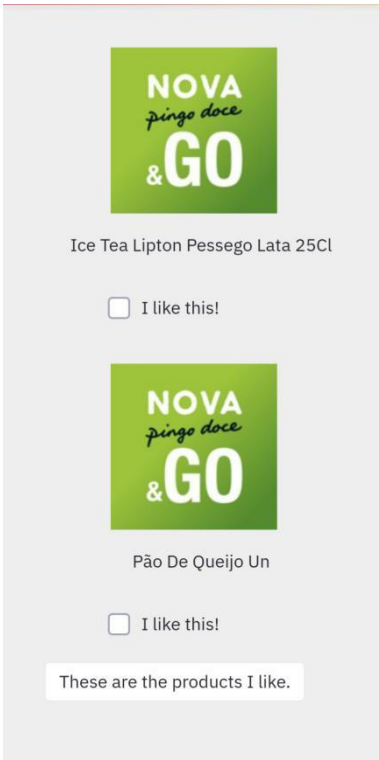


Table 12 - Hit rate at 6 for the period from November 15, 2020 until December 27, 2020

	date	loyalty	novelty	related	control
0	2020-11-15	0.302885	0.084746	0.034523	0.001881
1	2020-11-22	0.282799	0.087791	0.037486	0.000833
2	2020-11-29	0.315044	0.097046	0.042182	0.002074
3	2020-12-06	0.298635	0.143123	0.025943	0.002166
4	2020-12-13	0.367021	0.082062	0.033274	0.000000
5	2020-12-20	0.320789	0.067086	0.034700	0.006438