INGRAM MICRO

Section B

Yvonne Lu,

Kevin Raja,

Yuyang Wang,

Viktoriia Pinsker,

Meng(Mona) Xu

Faculty Advisor: John Turner

Table of Contents

I. Executive Summary

In the last couple of years the global eCommerce field has grown rapidly, expecting to increase to 5 trillion USD in 2022 (Statista). With an increasing number of companies competing for customer's attention, one of the most advantageous features that help boost user interaction and enhance shopping potential is product recommendation.

Product recommendation engines have the potential to alter the way websites connect with retailers and end users, provide customers with a refined user experience, and maximize their return on the investment driven by the information they can gain on customers' personal preferences and purchases. By using advanced algorithms such as machine learning and artificial intelligence to full advantage, a recommendation system can assist with showing customers the relevant products they want or need.

II. Introduction

Ingram Micro is a global leader in technology and supply chain services. They are the largest distributor of technological products that bring approximately 47 billion dollars in revenue and ship about 1.5 billion units per year. Ingram Micro aims to create growth opportunities within the small midsize business markets as more businesses use technology to add scale, enhance and improve productivity. Ingram Micro is positioned at the intersection of thousands of vendor, reseller and retailer partners. They offer more than 280,000 products (such as desktop and notebook personal computers, servers and workstations, personal digital assistants, and wireless devices) with inventory from 1,700 technology manufacturers and personal computer suppliers. On top of that, Ingram Micro is delivering logistics and supply-chain management services to increase efficiency for companies in the international technology supply chain.

Ingram Micro specializes in 3 market segments: corporate technology providers, consumer marketers, and VAR. In order to better understand what is relevant to their clients, Ingram Micro needs to be aware of the end users' purchase decisions. Having analyzed the patterns behind the end users' buying behavior and the similarities between them, the company can integrate a recommendation system that will recommend resellers only those products most desirable by their consumers.

III. Problem statement

Considering Ingram Micro's great market share and capability in the technology and supply chain sector and its place at the intersection of vendor, reseller and retailer, it is much necessary and needed to for Ingram micro be operating a diversified, efficient, fast, reliable recommendation engine to provide its end-users with up-to-date, relevant and easily accessed product recommendations in efforts to help with their purchasing decisions. However, most companies merely incorporate a static recommendation system that wasn't able to produce personalized recommendation. Such methods output inflexible results that do not support further business development and user engagement.

In that, the goal of this project is to leverage a network association model and a latent factor model, via python, to construct a customized recommendation engine that provides a range of diversified top 10 products to Ingram Micro's end-users. We believe our recommendation engine will produce recommendations in real time that are personalized to users and also beneficial to Ingram Micro's business development. We predict this accurate information will not only increase customer loyalty and engagement but also increase Ingram

Micro's total sales and profits. More specifically, we believe our recommendation engine will help with customers' decision making and increase Ingram Micro's average carted value.

We have also developed ways to test the engine's accuracy. They will provide us with an initial glance at its performance. However, additional testing strategies could also be further developed to monitor its performance based on other metrics and business implications, should they change in the future.

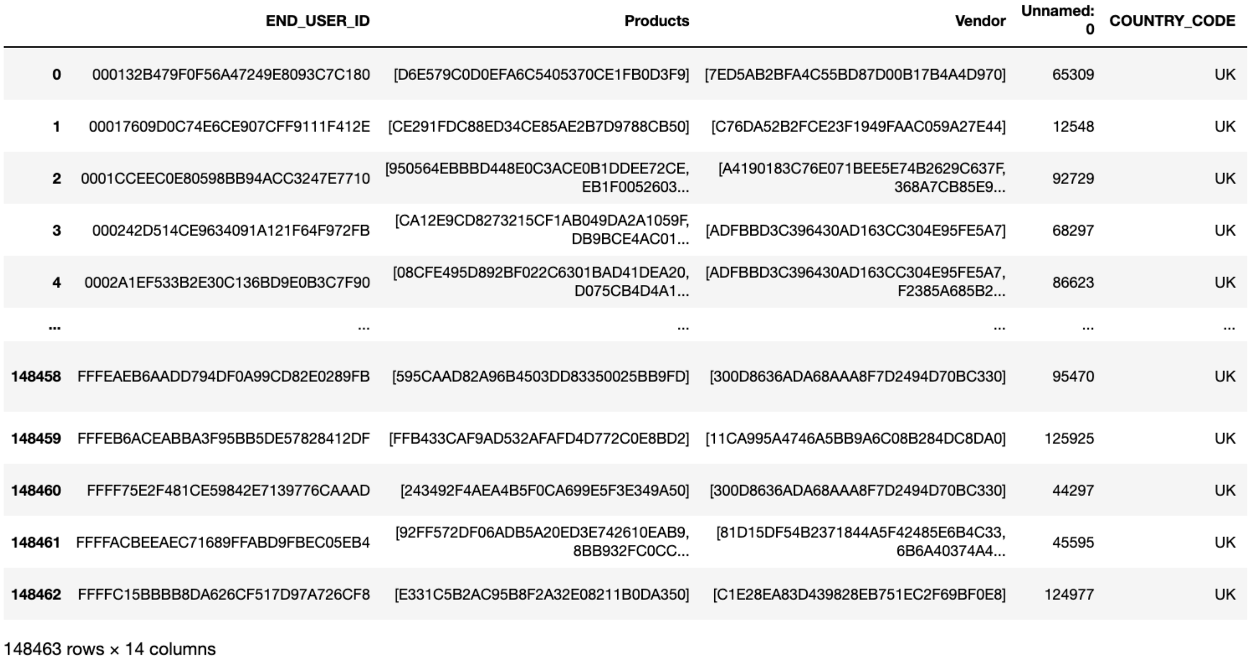
IV. Data Description

We were given four datasets: Invoice, Vendor, Product, and End User. These files contain information on Ingram Micro's clients in the United Kingdom exclusively. Below is a detailed structure of each of them:

- Invoice data:
    - Invoice number;
    - Invoice date;
    - Invoice Line;
    - vendor_key (vendor identifier in Ingram Micro);
    - End_user_key (end user identifier in Ingram Micro);
    - product_key (product identifier in Ingram Micro);
    - Quantity_shipped;
    - binned Unit_price_in_US$.
- Vendor data:
    - vendor key;
    - vendor_category;
    - vendor_SubCategory.
- Product data:
    - product_key;
    - product_category;
    - product_SubCategory;
    - product_description.
- End_user data:
    - End_user_key;
    - State;
    - Country;

- Category1 (Level 1 Verticals);
- Category2 (Level 2 Verticals);
- Employees_total (Estimated total employees);
- binned YEAR STARTED.

The product file contains information about advanced solutions, hybrid IT, and cyber security items.



	END_USER_ID	Products	Vendor	Unnamed: 0	COUNTRY_CODE
0	000132B479F0F56A47249E8093C7C180	[D6E579C0D0EFA6C5405370CE1FB0D3F9]	[7ED5AB2BFA4C55BD87D00B17B4A4D970]	65309	UK
1	00017609D0C74E6CE907CFF9111F412E	[CE291FDC88ED34CE85AE2B7D9788CB50]	[C76DA52B2FCE23F1949FAAC059A27E44]	12548	UK
2	0001CCEEC0E80598BB94ACC3247E7710	[950564EBBBD448E0C3ACE0B1DDEE72CE, EB1F0052603...	[A4190183C76E071BEE5E74B2629C637F, 368A7CB85E9...	92729	UK
3	000242D514CE9634091A121F64F972FB	[CA12E9CD8273215CF1AB049DA2A1059F, DB9BCE4AC01...	[ADFBBD3C396430AD163CC304E95FE5A7]	68297	UK
4	0002A1EF533B2E30C136BD9E0B3C7F90	[08CFE495D892BF022C6301BAD41DEA20, D075CB4D4A1...	[ADFBBD3C396430AD163CC304E95FE5A7, F2385A685B2...	86623	UK
...	...	...	...	...	...
148458	FFFEAEB6AADD794DF0A99CD82E0289FB	[595CAAD82A96B4503DD83350025BB9FD]	[300D8636ADA68AAA8F7D2494D70BC330]	95470	UK
148459	FFFEB6ACEABBA3F95BB5DE57828412DF	[FFB433CAF9AD532AFAFD4D772C0E8BD2]	[11CA995A4746A5BB9A6C08B284DC8DA0]	125925	UK
148460	FFFF75E2F481CE59842E7139776CAAAD	[243492F4AEA4B5F0CA699E5F3E349A50]	[300D8636ADA68AAA8F7D2494D70BC330]	44297	UK
148461	FFFFACBEEAEC71689FFABD9FBEC05EB4	[92FF572DF06ADB5A20ED3E742610EAB9, 8BB932FC0CC...	[81D15DF54B2371844A5F42485E6B4C33, 6B6A40374A4...	45595	UK
148462	FFFFC15BBBB8DA626CF517D97A726CF8	[E331C5B2AC95B8F2A32E08211B0DA350]	[C1E28EA83D439828EB751EC2F69BF0E8]	124977	UK

148463 rows × 14 columns

For those four datasets, we have Invoice ID, Product ID, EndUser ID, and Vendor ID as unique identifiers. Based on these four unique identifiers, we can connect the four separate files with each other. When doing this we have 148,463 unique customers in our data with 31924 unique products.

V. Network Association Model

A. Model Overview

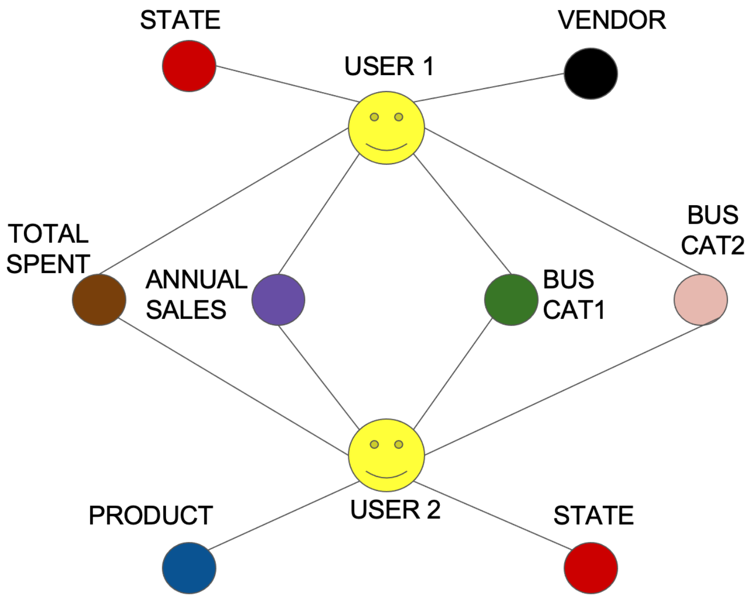
The Network Association Model constructs a network graph for all users by matching similarity between each user's characteristics/attributes and purchase history. These attributes include business category, annual sales, products purchased, and total spent. Once we have that graph, we can run the recommendation function by inputting a user into the engine. The recommendation function will identify five users that share the most common attributes with the input user. This means they have the most connections to the input user in the graph. Then finally, out of those five users, we get the top ten most purchased products between them as recommendations.

B. Preprocessing

For our preprocessing, we added a total spent column in the invoice dataset that shows the total amount spent on each user. We binned annual sales and total spent into different bins to make it easier to classify. Then we joined all the datasets together to test our Network Model

C. Network Graph

Below is an example subgraph of our Network Model. We get similarity between users that connects users based on different attributes. In the graph, we have two users that are very similar to each other, matched by four attributes (Total Spent, Annual Sales, Business Category 1, Business Category 2). The other nodes that are connected to the users are other attributes that each user has that are not shared between them. In the example below, User2 is one of the most similar customers for User1. A stronger match would have more attributes being connected together.



D. Testing

In order to test our models, we used two different sampling methods. Our first sample is a random sample of 250 customers which is used to replicate an average customer. Our second sample for testing is taking the top 250 highest spending customers (big spenders). Our assumption that the highest spending customers are most likely to actually purchase the products recommended to them because they are loyal customers so we believe that strong performance on this sample is especially important because it shows that the model can increase profitability.

We tested the recommendation engine for both samples in three different cases: best case, worst case, and by date.

In the best case, we observe how the recommendation engine performs on users with long purchase histories by training the model on every user and every purchase made by that user.

In the worst case, we observe how accurately the recommendation engine will perform on brand new users, assuming that the customers in our sample have no purchase history.

Lastly, we tested the engine by taking product purchases before August 2020 and used them to generate recommendations in order to predict purchases after August 2020. The by date test is what we consider a more realistic scenario.

Below is an overview on how we split the testing and training sets in each of the three test cases:

	Training	**Testing**
Best Case **Test how model works when users have large purchase history**	Whole dataset	250 Customers from training
Worst Case **Test how model works when users have no purchase history (read their mind)**	Remove purchase history of customers in test set	250 Customers from training
By Date **More Realistic Scenario - Use limited purchase history to predict purchases**	Dataset - purchased before August 1st 2020 cutoff	250 Customers - purchased before and after August 1st 2020 cutoff

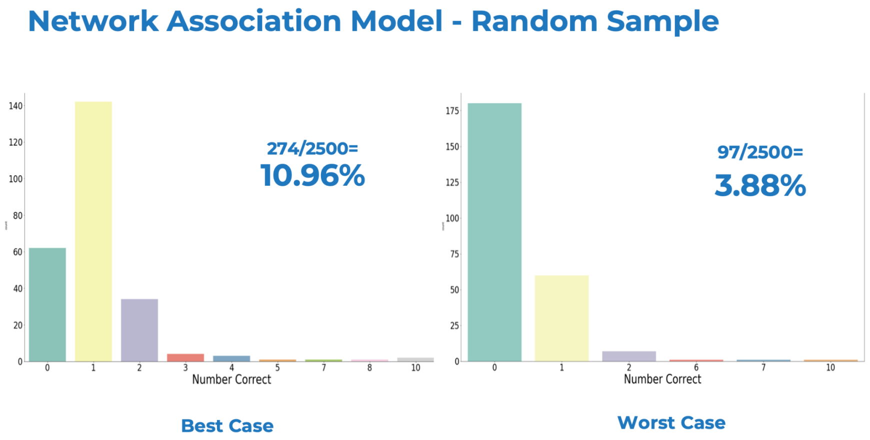
We used precision rate as our validation metric for all our tests:

$$\text{Precision} = \frac{\text{Number of Correct recommendations}}{\text{Total number of products recommended}}$$

*we define a correct recommendation as a product that is recommended for a customer that turns up in that customer’s purchase history

*Industry standard precision rate is 3-5%

The bar chart below shows the number of correct recommendations per user in our random sample tests for best case and worst case with the percentages being the precision of the model in each case.



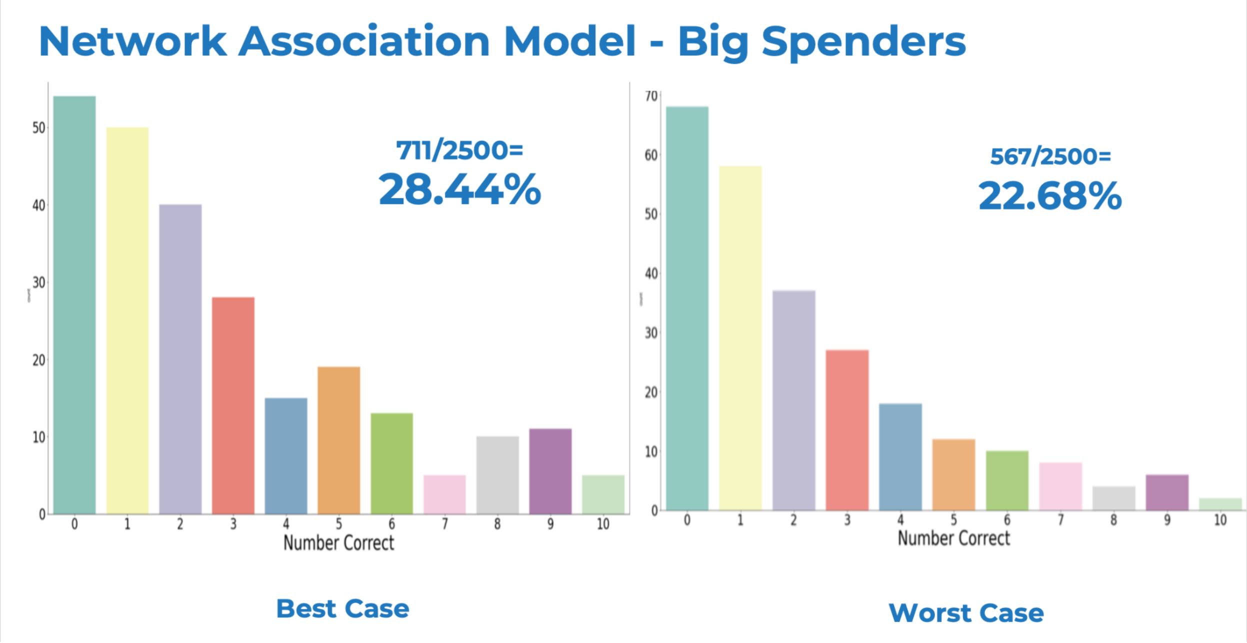
Network Association Model - Random Sample

Best Case

Worst Case

The x-axis on the graph is the number of recommendations that were actually purchased and the y-axis is the number of customers that had that amount of correct recommendations. For example, in the Network Model best case scenario, there were about 60 customers that had zero correct recommendations but the rest of the customers had at least one correct recommendation. We followed the same logic for all bar charts in this report.

For a random sample, we obtained 10.96% precision in the best case and 3.88% in the worst case.



Network Association Model - Big Spenders

711/2500=
28.44%

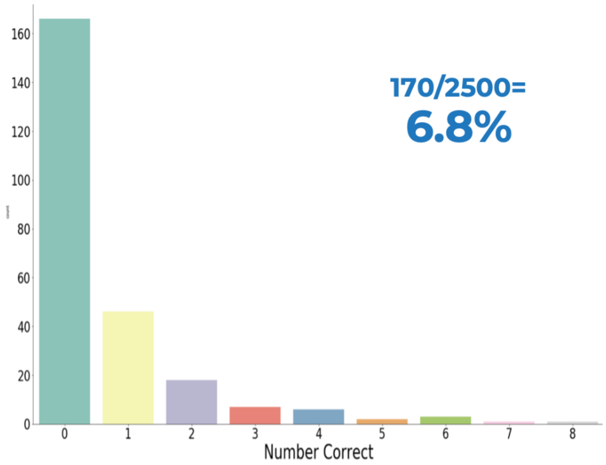
Best Case

567/2500=
22.68%

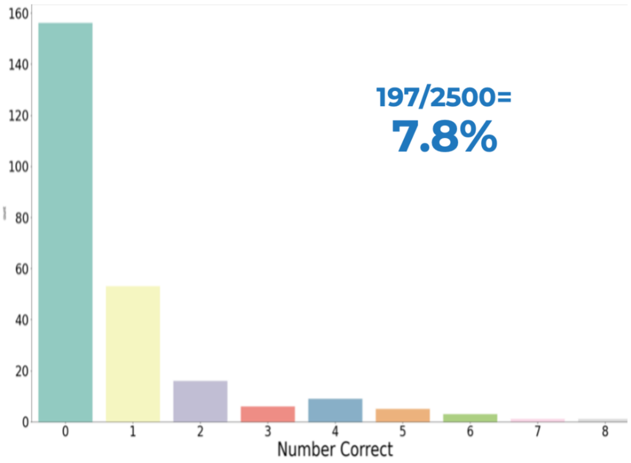
Worst Case

The graphs above shows a distribution of correct recommendations for big spenders. For big spenders, we obtained a 28.44% precision rate in the best case and 22.68% in the worst case. Since big spenders are relatively similar in both cases, it shows that the model is good at giving recommendations to big spenders.

Network Association Model - By Date



Random Sample



Big Spenders

The graph above shows the by date test case. We obtained 6.8% precision in the random sample and 7.8% for the big spenders.

E. Model Feedback

Some of the positives we got out of the Network Model is that it performed favorably in all cases and it fixes the issue of customers having no purchase history by recording user characteristics in addition to their product purchases. The biggest positive is that this model runs extremely fast on our server.

However, one downside is that the network model is difficult to narrow down recommendations when there is a tie in the top ten most popular products. It also requires a lot of data to be recorded in order to obtain each user's characteristics.

In order to improve the network recommendation engine, we would need to improve the process of narrowing down close customers and popular products purchased by close customers.

If Ingram Micro decides to record more end user characteristics then we could provide more customized product recommendations.

VI. Latent Factor Model

A. Model Overview

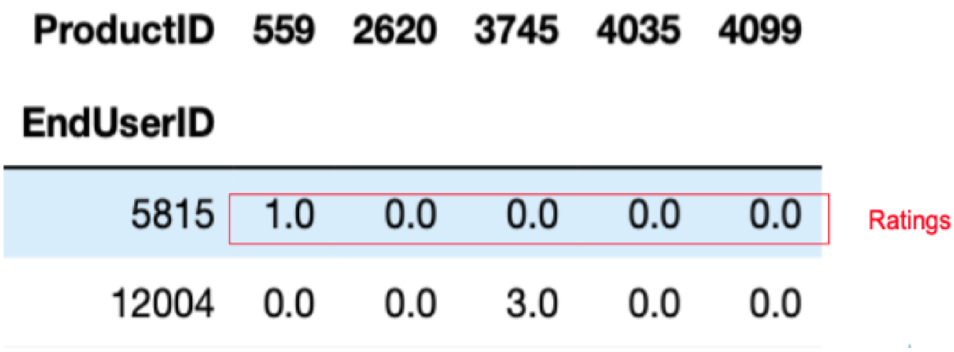
The Latent Factor model uses the product ratings of products that a customer previously purchased to help predict how that customer will rate products that they have not purchased yet. It does this through the use of latent factors (variables that are not explicitly recorded) to match products together for each user

B. Preprocessing

We were not provided with product ratings for each user in our dataset so we decided to devise a reasonable formula to determine the product ratings. We have a variable called unite price which gives the amount a user spent on a product. We create a variable called total spent which is the sum to the unite price for each user. Finally, we estimated the ratings for each product each user purchased using the formula: **unite price/ total spent *100**, which is the percentage of their total the user spent on the product.

C. Model Process

First we create a matrix of **products and users** with values inside the matrix being the product ratings. The value for product rating will be zero if the user did not purchase the product. The dimensions of the actual matrix are the number of users x the number of products. A small section is shown below:

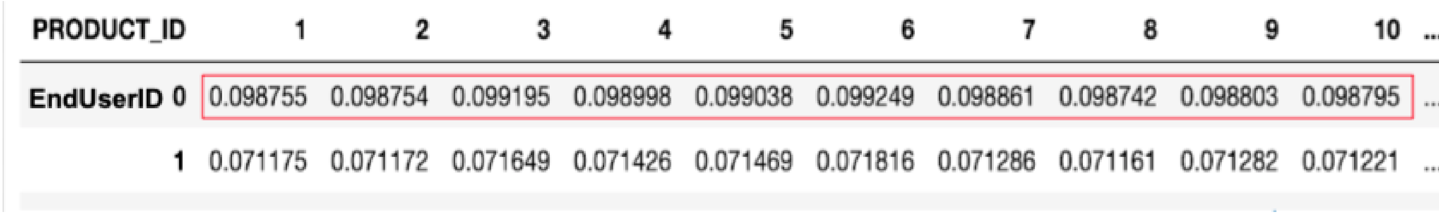


ProductID	559	2620	3745	4035	4099
EndUserID					
5815	1.0	0.0	0.0	0.0	0.0
12004	0.0	0.0	3.0	0.0	0.0

Ratings

* For end user 5815 they gave product 559 a rating of 1. They gave product 2620 a rating of 0 because they didn't purchase the product

Due to the fact that the matrix has a large number of 0 values we have to normalize the ratings. We Normalize Ratings by subtracting the mean of each column from each rating in its column. Then we use the algorithm **Truncated Singular Value Decomposition (SVD)**. This algorithm uses a set number of latent variables to compute a prediction matrix which contains predicted ratings of each product for each customer. Due to SVD, ratings are now on a scale from 0-1, but it has the same dimensions of the previous matrix. To use the prediction matrix we take a userid as an input then we look at that user's corresponding row in the matrix. Then we take the top 10 values in that row. The corresponding products for those values are given as recommendations for that user. An example of the prediction matrix is shown below:

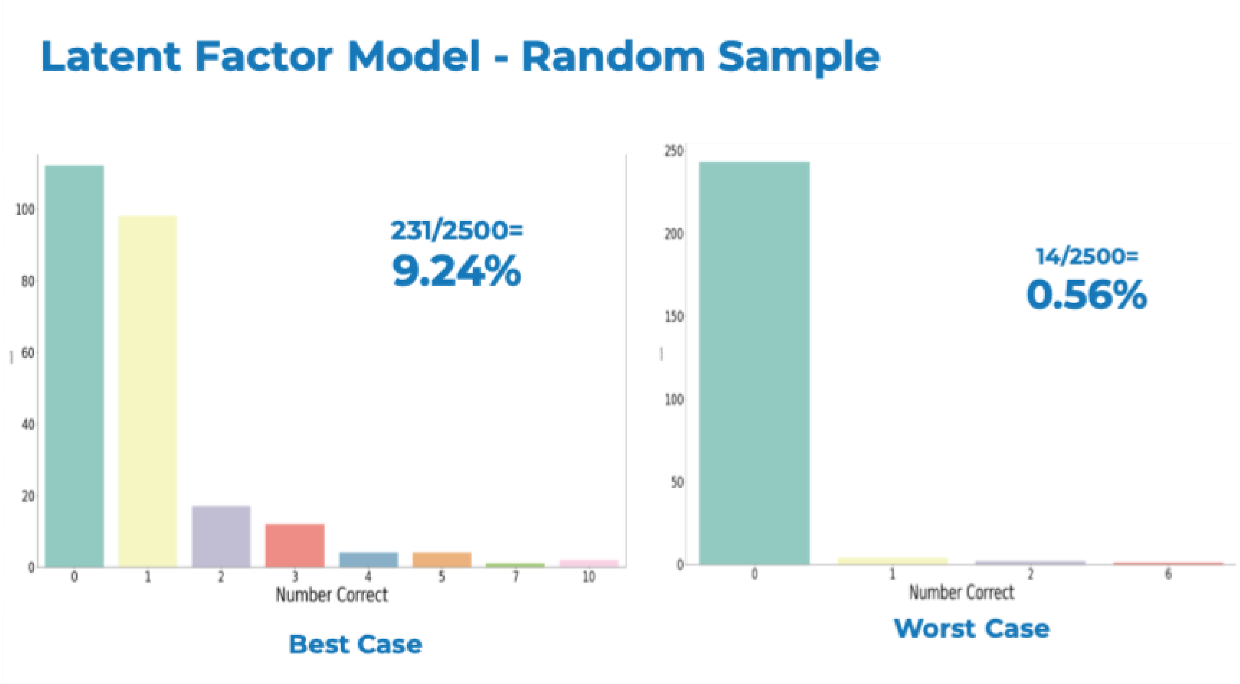


PRODUCT_ID		1	2	3	4	5	6	7	8	9	10	...
EndUserID	0	0.098755	0.098754	0.099195	0.098998	0.099038	0.099249	0.098861	0.098742	0.098803	0.098795	...
	1	0.071175	0.071172	0.071649	0.071426	0.071469	0.071816	0.071286	0.071161	0.071282	0.071221	...

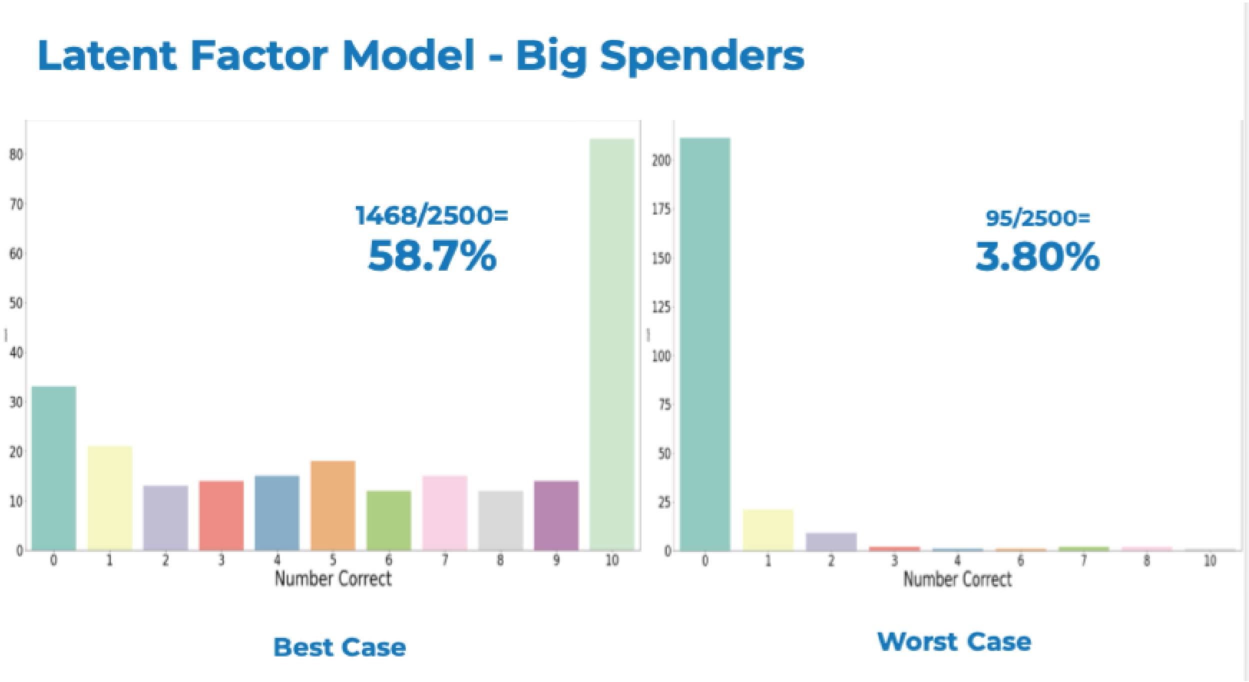
* For the row with end user 0 take the top ten values look at the corresponding product ids and give those as recommendations for user 0

D. Testing

We follow the exact same testing protocol for our Latent Factor Model as we did for Network Model. Again the bar charts below show the number of recommendations (x-axis) correct for each customer (y-axis)



For the random sample of 2550 customers, we obtained 9.24% precision in the best case and 0.56% in the worst case.



Latent Factor Model - Big Spenders

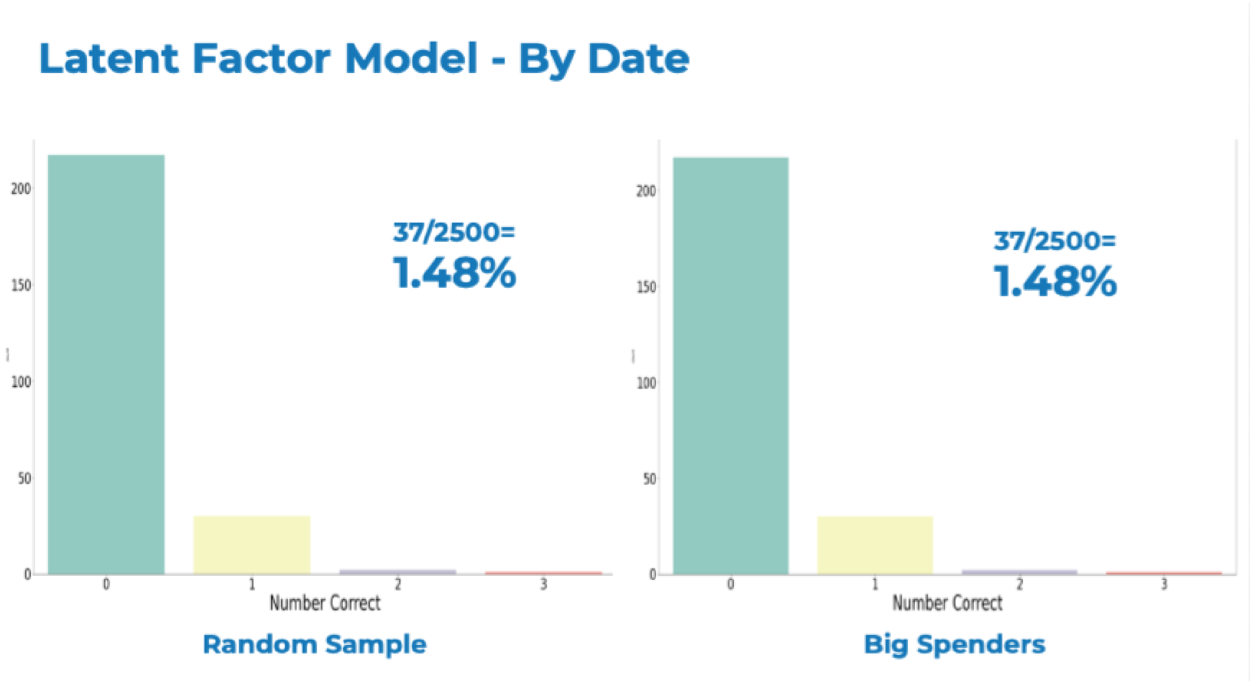
1468/2500=
58.7%

95/2500=
3.80%

Best Case

Worst Case

The graphs above shows a distribution of correct recommendations for the top 250 biggest spenders. For big spenders, we obtained a 58.7% precision rate in the best case and 3.80% in the worst case. The model performs exceptionally well on big spenders when we keep their purchase history in the best case. Even in the worst case it performs within industry standards for big spenders.



Latent Factor Model - By Date

Random Sample

Big Spenders

The graph above shows precision in the by date test case. We obtained 1.48% precision in the random sample and 1.48% for the big spenders.

E. Model Feedback

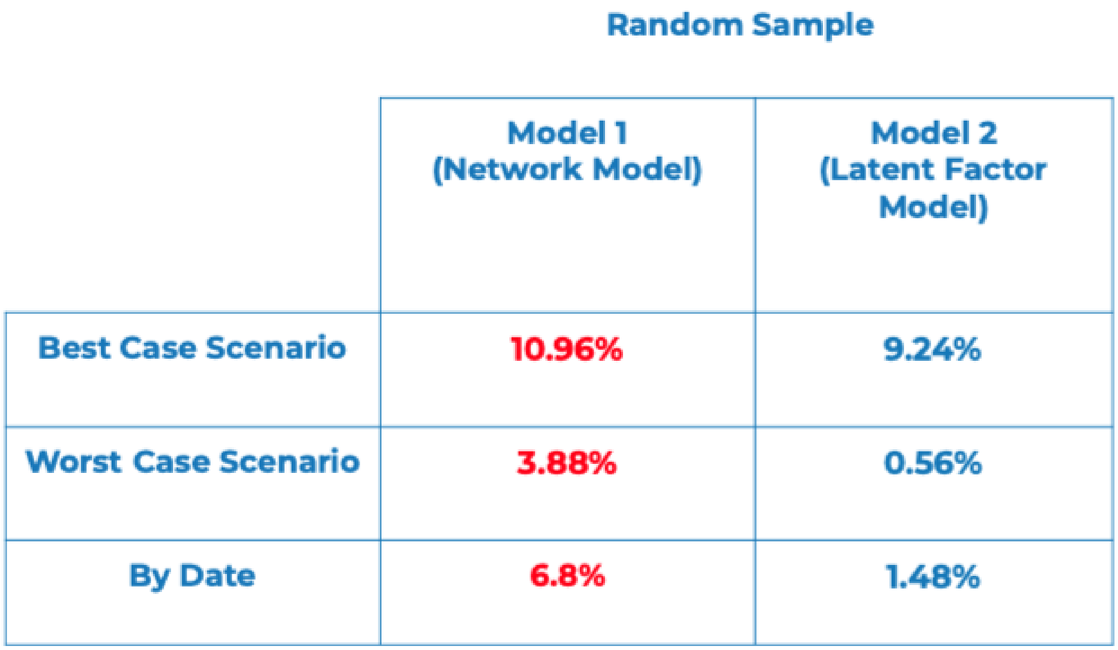
Some of the positives we got out of the Latent Factor Model is that it performed exceptionally in the best case. This is because it works well when customers in the training set have a large number of purchases. It also requires less data to be recorded as we only need customer id, product id, and product rating to build and train the model. This model is also used by Netflix which has one of the best personalized recommender systems in use today. So it has a proven track record of success in other industries.

Some negatives from this model are that it suffers from a cold start. If a user has too small of a purchase history then it will be difficult to predict product ratings for them. That is the reason for its poor performance in the worst case and by date tests. It is also a much slower model to run than the network model

In order to improve the latent factor model, Ingram Should consider recording customer ratings for products in their dataset so there is a standard 1-5 star rating scale. To improve the latent factor model accuracy you can also increase the number of latent factors for the svd algorithm to a number greater than 50. There also should be a better way to determine what to insert into the matrix when a customer has not purchased a product instead of just 0.

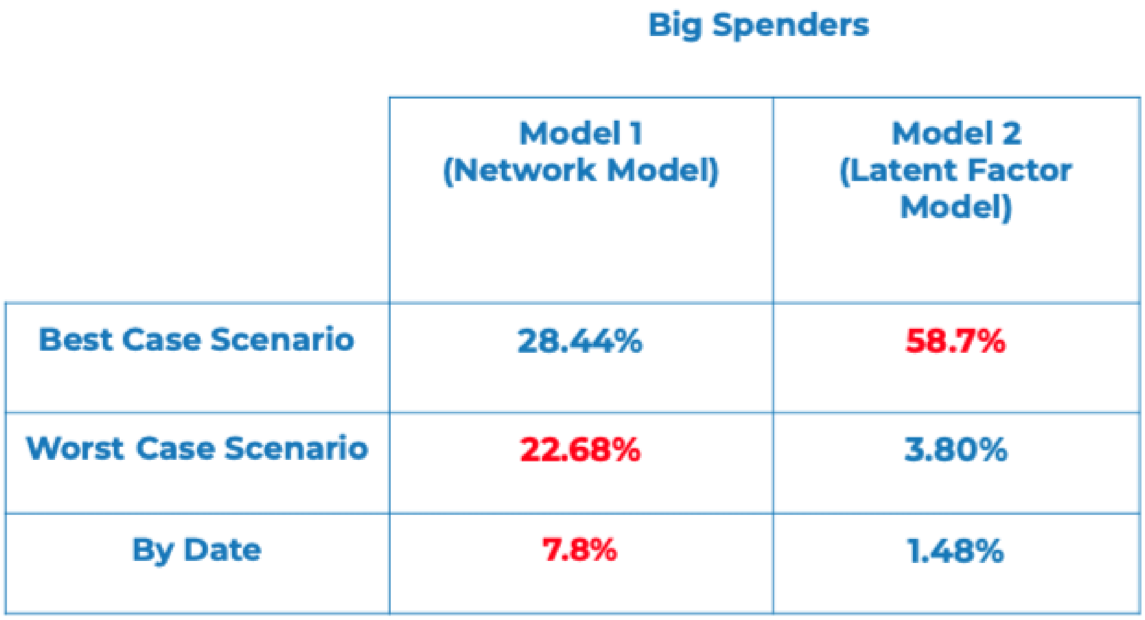
VII. Compiled Test Results

Below we have compiled all our test results for precision for each sample:



Random Sample

	Model 1 (Network Model)	Model 2 (Latent Factor Model)
Best Case Scenario	10.96%	9.24%
Worst Case Scenario	3.88%	0.56%
By Date	6.8%	1.48%



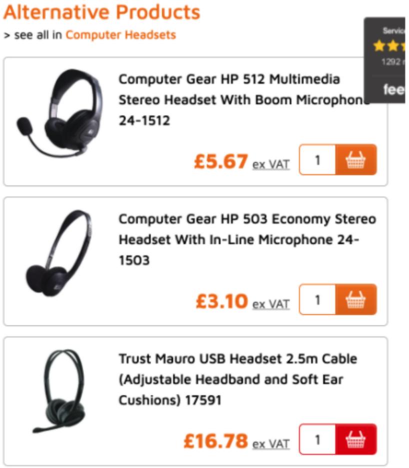
Big Spenders

	Model 1 (Network Model)	Model 2 (Latent Factor Model)
Best Case Scenario	28.44%	58.7%
Worst Case Scenario	22.68%	3.80%
By Date	7.8%	1.48%

Our Network Model out performs the latent factor model almost all the time so we are recommending that Ingram Micro use this model as a recommendation engine.

VII. Results

Below is a comparison between recommendations retrieved from an Ingram Micro partner reseller's website and a sample of recommendation results that are generated by our Network model engine:



Alternative Products

> see all in Computer Headsets

Computer Gear HP 512 Multimedia Stereo Headset With Boom Microphone 24-1512

£5.67 ex VAT 1

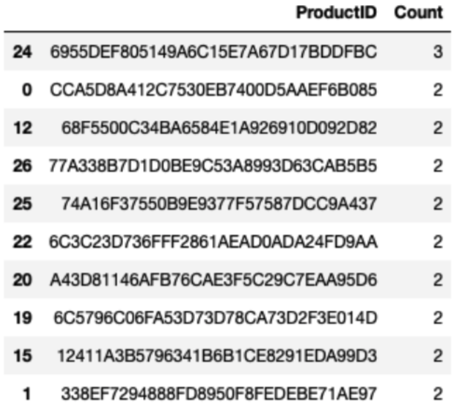
Computer Gear HP 503 Economy Stereo Headset With In-Line Microphone 24-1503

£3.10 ex VAT 1

Trust Mauro USB Headset 2.5m Cable (Adjustable Headband and Soft Ear Cushions) 17591

£16.78 ex VAT 1

VS



	ProductID	Count
24	6955DEF805149A6C15E7A67D17BDDFBC	3
0	CCA5D8A412C7530EB7400D5AAEF6B085	2
12	68F5500C34BA6584E1A926910D092D82	2
26	77A338B7D1D0BE9C53A8993D63CAB5B5	2
25	74A16F37550B9E9377F57587DCC9A437	2
22	6C3C23D736FFF2861AEAD0ADA24FD9AA	2
20	A43D81146AFB76CAE3F5C29C7EAA95D6	2
19	6C5796C06FA53D73D78CA73D2F3E014D	2
15	12411A3B5796341B6B1CE8291EDA99D3	2
1	338EF7294888FD8950F8FEDEBE71AE97	2

To further visualize results of our engine, please consider the following Products.

Headsets Alternatives

Product ID:
6955DEF805149A6C15E7A67D17BDDFBC
Product name: USB HEADSET H390
Vendor: Logitech

Product ID:
CCA5D8A412C7530EB7400D5AAEF6B085
Product Name: TRUST USB V2.0 HEADSET
Vendor: TRUST COMPUTER

Speakers and Sound Bars

Product ID:
68F5500C34BA6584E1A926910D092D82
Product Name: Z200 SPEAKERS MIDNIGHT
Vendor: Logitech

Product ID:
12411A3B5796341B6B1CE8291EDA99D3
Product Name: REMO 2.0 SPEAKER SET
Vendor: Trust Computer

Product Bundling Ideas

Product ID:
A43D81146AFB76CAE3F5C29C7EAA95D6
Product Name: TRINO HD VIDEO WEBCAM
Vendor: Trust Computer

Product ID:
338EF7294888FD8950F8FEDEBE71AE97
Product Name: ASTO SOUND BAR PC
Vendor: Trust Computer

It is easy to observe that our results provide far more variety and consider multiple scenarios in terms of what purpose the customer was purchasing this particular product for. Our recommendation engine not only suggests close alternatives, but it also suggests items that are paired well together. As shown in the example, if the customer is purchasing headsets for multimedia purposes, our engine would also suggest speakers and sound bars as either alternatives or additional equipment that might be needed. For a scenario where the customer is purchasing headsets for working from home and online meetings, our engine also suggested bundling headsets with a webcam which will also be an essential part of the practice.

VIII. Conclusions and Business Implications

After evaluating the results of each model, we found that the network model provides the best recommendation results for the dataset we were given. What this tells us is that saving user information is beneficial to giving accurate product recommendations. We can also use our network model for product bundling where we can look at all products that were recommended and analyze the most common pairs. Good recommendations help improve Ingram Micro's customer loyalty from resellers and help improve the sales volume of new products and introduce new brands. More importantly, a good recommendation can lead to increased average carted value from resellers to Ingram Micro.

IX. Appendix

Works Cited

1. Sabanoglu, T. (2021, March 26). *Global retail e-commerce market size 2014-2023*. Statista. https://www.statista.com/statistics/379046/worldwide-retail-e-commerce-sales/#:~:text=In%202020%2C%20retail%20e%2Dcommerce,trillion%20US%20dollars%20in%202022.
2. *Home*. Ingrammicro. (n.d.). https://www.ingrammicro.com/.
3. Real Python. (2021, June 5). *Build a Recommendation Engine With Collaborative Filtering*. Real Python. https://realpython.com/build-recommendation-engine-collaborative-filtering/.
4. Yuefeng Zhang, P. D. (2020, August 12). *Machine Learning for Building Recommender System in Python*. Medium. https://towardsdatascience.com/machine-learning-for-building-recommender-system-in-python-9e4922dd7e97.