HEC MONTRÉAL

Recommender Systems with Integrated Visual Representations

par
Arash Hamidian

Laurent Charlin
HEC Montréal
Directeur de recherche

Sciences de la gestion
(Spécialisation Intelligence d'affaires)

Mémoire présenté en vue de l'obtention
du grade de maîtrise ès sciences en gestion
(M. Sc.)

December 2022

Résumé

L'économie de la longue traîne (long-tail) repose sur la diversité, et l'idée est de réaliser un profit en vendant différents produits à un grand nombre de personnes. Les magasins physiques ont un espace limité pour exposer les produits aux clients et la stratégie réussite serait de stocker et de vendre uniquement les articles les plus populaires que la plupart des clients sont prêts à acheter. En revanche, les magasins en ligne (*Amazon.com* par exemple) ne rencontrent pas de restrictions de stockage de contenu et de produits numériques en raison du faible coût de location des sites web et du peu d'espace de stockage dont les produits numériques ont besoin. Ils peuvent ainsi diversifier leurs ventes et s'adapter facilement à l'économie de la longue traîne.

Cependant, un défi est de savoir comment le propriétaire du site web peut comprendre ce qu'il faut offrir aux utilisateurs. C'est là que les systèmes de recommandation deviennent importants car ils permettent de gérer les offres de produits. Il s'agit d'une forme de système de filtrage d'informations qui peut modifier les préférences des utilisateurs. Ces systèmes suggèrent des articles en découvrant le modèle des facteurs qui sont en corrélation avec les interactions entre les utilisateurs et les produits vendus.

Il existe plusieurs techniques mathématiques qui soutiennent divers concepts de recommandation. Les techniques avancées combinent souvent plusieurs facteurs, parmi lesquels (mais sans s'y limiter) le profil de l'utilisateur, le texte des commentaires, les caractéristiques visuelles des items, les scores de similarité, les modèles et les séquences des commandes. Dans cette recherche, nous explorons comment les concepts de vision artificielle et de systèmes de recommandation peuvent être utilisés conjointement. Nous

aimerions répondre à quelques questions concernant la façon dont la vision artificielle pourrait améliorer l'efficacité et enrichir la qualité des recommandations. Nous aimerions également comprendre si une telle idée pourrait nous aider à mieux concevoir les systèmes de recommandation.

Notre hypothèse de cette recherche est que les utilisateurs peuvent être totalement ou partiellement attirés par les images des produits, et que celles-ci peuvent exercer une grande influence sur leurs décisions. Nous pensons que les utilisateurs sont plus susceptibles de faire des transactions s'ils sont attirés au moins partiellement par les images des produits, et que l'apparence visuelle des items peut influencer leurs décisions. En intégrant des signaux visuels dans les modèles de recommandation, nous pouvons apprendre les facteurs de décision visuels des utilisateurs, ce qui peut finalement améliorer les systèmes de recommandation.

Dans cette étude, nous construisons deux modèles de recommandation par factorisation matricielle : 1) un modèle de recommandation non-visuel et 2) un système de recommandation visuel. Nous comparons ces modèles pour comprendre comment une représentation visuelle des éléments pourrait améliorer le système. Les modèles utilisent les rétroactions explicites des utilisateurs (explicit feedback) et sont entrainés pour apprendre les paramètres visuels et non visuels. Pour résoudre le modèle de factorisation matricielle, l'approche d'optimisation Alternate Least Square (ALS) est utilisée [9] et une hypothèse importante de ce travail est que la tendance et l’attraction de l'utilisateur envers les caractéristiques visuelles sont stables et statiques [19]. Cette hypothèse nous permet de construire des modèles statiques en apprenant un nombre limité de paramètres.

Cette thèse a pour contribution principale de dériver de nouvelles équations de factorisation matricielle pour une rétroaction explicite en fusionnant les facteurs visuels des utilisateurs et des éléments dans le modèle de base. Nous avons utilisé ces nouvelles équations pour entraîner le modèle alternatif de cette recherche afin de prédire les classements (rating) et d'étudier l'effet des signaux visuels.

Les résultats expérimentaux montrent que les variations de classement (rating) sont un peu mieux expliquées en incluant la représentation visuelle des items, et les recommandations sont plus précises quand les paramètres sont optimisés. En fusionnant les signaux visuels, on s'aperçoit que la variabilité des classements est mieux expliquée par des facteurs non-visuels plus visuels plutôt que par des facteurs non-visuels uniquement. Il est démontré que la personnalisation par factorisation matricielle dans cette étude est toujours supérieure à toute personnalisation aléatoire, et la performance de classement des modèles (ranking performance) avec des signaux visuels est supérieure à celle du modèle non-visuel. Cela indique que l'intégration de facteurs de décision visuels peut améliorer la personnalisation et enfoncer les décisions des utilisateurs en ce qui concerne l'achat.

En outre, l'analyse de popularité montre que les modèles visuels font des recommandations en faveur d'items impopulaires alors que les modèles non-visuels recommandent des items assez populaires et très populaires. Cela signifie que le modèle visuel peut absorber différents types de signaux, ce qui le rend très utile pour des applications comme le 'cold-start'.

Mots clés : systèmes de recommandation, factorisation matricielle, modèle de rétroaction explicite, classement personnalisé, systèmes de recommandation visuels

Méthodes de recherche : Factorisation matricielle avec des signaux visuels intégrés en utilisant une rétroaction explicite

Abstract

The long-tail economy is all about diversity and the idea behind it is to make a profit by selling a few different products to many people. Physical stores have limited space to expose products to the clients and the successful strategy would be to store and sell only popular items that most clients are willing to buy. In contrast, online stores (e.g. Amazon.com) do not encounter restrictions in storing digital content and products because of cheap web rental and little storage that digital products need, hence they can diversify their sales and adapt to the long-tail economy conveniently.

However, one challenge is how the web owner understands what to offer users. That is where recommender systems become important. Recommender systems are great tools to manage product offerings. They are a form of information filtering system that can alter the preferences of the users. Recommenders suggest items by discovering the pattern of the factors that correlate with the interactions between users and selling products.

Several mathematical techniques are available to support diverse recommendation concepts. Advanced techniques often combine several factors including (but not limited to) user profile, the text of the reviews, visual features of the items, similarity scores, patterns, and sequences of the orders. In this research, we explore the concept of computer vision in conjunction with recommender systems. We would like to answer some questions about how computer vision could be involved in such systems to improve efficiency and enrich the quality of the recommendations. We would also like to understand whether such an idea could help us to make a better recommender design.

As a hypothesis of this research, we believe that users can be fully or partly attracted by product images, and the images can have a great influence on their decisions. Our thought is that users decide to make a transaction if they are at least partly attracted to the products' images, and the visual appearance of the items can have stimulating effects on their decisions. Therefore, incorporating visual signals into the recommender models enables us to learn users' visual decision factors which can ultimately improve the recommender systems.

In this work we build two recommendation models by matrix factorization: 1) a non-visual recommender model and 2) a visually-aware recommender system. We compare these models to understand how the visual representation of items might improve the recommender system. The models use explicit feedback from users, and they are trained to learn both visual and non-visual parameters. For solving the matrix factorization model, Alternate Least Square (ALS) optimization approach is used [9], and one important assumption of this research is that the user's tendency and attraction toward visual features are static [19]. Such an assumption allows us to build static models by learning a modest number of parameters.

The contribution of the thesis is to derive new matrix factorization equations for explicit feedback by incorporating the visual factors of the users and items into the baseline model. We used these new equations to train the alternative model of this research to predict the ratings and study the effect of the visual signals.

Experiment results show that the rating variations are explained slightly better by including the visual representation of the items, and recommendations are more precise

once the parameters are optimized. By incorporating the visual signals, the variability of the ratings is better explained through non-visual plus visual factors rather than by non-visual factors only. It is shown that personalization through matrix factorization in this study is always superior to any random personalization, and the ranking performance of the models with visual signals is higher than the non-visual model, which indicates that involving visual decision factors can enhance personalization and impacts user decisions positively for buying.

Besides, popularity analysis shows that visual models make recommendations in favor of unpopular items whereas non-visual model recommends fairly popular and very popular items. It signifies that the visual model absorbs different signals than non-visual model, so it can be very useful for applications such as cold-start.

Keywords : Recommender systems, Matrix Factorization, Explicit Feedback Models, Personalized Ranking, visually-aware recommender systems

Research methods : Matrix Factorization with integrated visual signals using explicit feedback

Table of contents

List of tables and figures

LIST OF TABLES

LIST OF FIGURES

List of abbreviations and acronyms

ALS: Alternating Least Square

CB: Content-Based

CF: Collaborative Filtering

CNN: Convolutional Neural Network

LFM: Latent Factor Model

MF: Matrix Factorization

RS: Recommender System

DCG: Discounted Cumulative Gain

NDCG: Normalized Discounted Cumulative Gain

SME: Subject Matter Expert

RMSE: Root Mean Squared Error

MAP: Mean Average Precision

RND: Random

TMP: Temporal

Dedications

To my loved ones, Farnoush, my beautiful wife, and my sons Nivan & Lian

Acknowledgments

Words cannot express my gratitude to my professor Laurent Charlin for his invaluable patience and feedback. I could not have undertaken this journey without his instructions, advice, and generous knowledge-sharing.

I am also grateful to my current and former managers and leads at Unity Technologies, Michael Furlong, Catherine Pang, Babak Khosravifar, and Sundesh Shetty for their inspiration, patience, and mental support.

Lastly, I would be remiss in not mentioning my family, especially my spouse and children. Their patience was praiseworthy and their belief in me kept my motivation high during this research.

1. Introduction

Shop owners have limited space to expose their products to clients. They also have limited customers stepping into the shop at a time because the shop is a physical store. The successful strategy would then be to store and sell only popular items, otherwise, the owners would have made the risk of inventory cost rise for the product that could never sell. Owners of web stores on the other hand do not encounter restrictions in storing digital content and products because the rental web is cheap and digital products do not take up much space. Storing a lot of content costs them little or nothing. The long-tail economy is all about diversity and the idea is to store too many items but to make a profit by selling a few of each to many people [1].

Amazon.com, *Netflix.com*, and *Barnesandnoble.com* are a few examples of digital stores that sell products and content to their users. The former sells a wide range of products from groceries to movies, the second is a movie service, and the latter recommends and sells books. However, the challenge is how can users find the items they need and how to understand what to offer them [1][2]. That is where recommender systems come into play.

Recommender systems are designed to suggest items by discovering the pattern of the factors which correlate with the interactions between users and selling products. They are a form of information filtering system with the capacity to alter the preferences of the users. Unveiling the pattern or understanding the relationship between users and items is the primary goal of such systems for generating recommendations. User-item patterns are often discovered by employing user feedback, their page activities, or generally by understanding their interests. Recommender systems are essential tools that empower us to manage information overload in the current data expansion era. Systems that generate a music playlist for each user, systems that suggest advertisements or suggest a connection on social media, and systems that rank newsfeeds (or contents) or recommend books are just a few examples of the application of recommendation engines.

Over the last decade, recommender systems have evolved enormously, and several mathematical techniques and various algorithms have been offered to support diverse recommendation concepts, address the sparsity problem, deal with cold start problems, and generally meet consumer demands.

The simplest recommendation systems work based on popularity where the algorithms identify popular items and recommend them to users who had not seen them. This method however is not aligned with the idea and goal of the long-tail economy. Hence, instead of such primitive systems, most of the time sophisticated designs are used to infer user preferences. Advanced techniques often combine several factors including (but not limited to) user profile, the text of the item reviews, visual features of the items, similarity scores, patterns, and sequences of the orders.

Due to the recent progress in computer vision technologies and techniques, in this work, we explore such concepts in conjunction with recommender systems. We are specifically interested in pushing the boundaries in the realm of recommender systems and answering some questions about how computer vision could be involved in such systems to improve the efficiency and enrich the quality of the recommendations, and whether such an idea could help us to make a better design which eventually leads to better personalization.

2. Hypothesis and Research Questions

Our hypothesis is the driving force in conducting this research. We deeply believe that users can be fully or partly attracted by product images, and the images can have a great influence on their decisions. We think that if users need a product, it is still possible that they make the transaction even without any product images, but the product images boost the users' decision process, and they can have a stronger convincing role to make users purchase. Our thought is that users decide to make a transaction if they are attracted (or at least partly attracted) to the items' images. The visual appearance of the items can have stimulating effects on their decisions. We believe that users are more willing to purchase if they are exposed to the images than if not. Therefore, incorporating visual signals into the recommendation models enables us to learn users' visual decision factors which can ultimately improve the systems. Among our several thoughts, in this project we try to answer the following research questions:

- How do recommender systems with embedded visual representations perform compared to conventional systems? Does the visual representation of the items have any positive impact on users' personalized recommendations?
- The matrix factorization model is composed of user and item factors plus a few bias terms. How should the equations be tweaked for proper inclusion of the visual signals? The answer to this question should cover the novelty of this thesis where MF equations are adapted for using the visual signals for recommending the items.

3. Objectives

In this work, we build a visually aware recommender system to understand how visual representation of items might improve the recommender system. The system learns from the users' feedback. We focus on food and grocery items. We encode their associated product image and integrate the resulting visual features into the recommendation objective.

Packer et. al. [19] and He et. al. [20] have proposed a matrix factorization approach that uses implicit feedback for clothing recommendations. With the novel approach of this work that uses explicit feedback for grocery items, we model user preferences in relation to the visual signals to better explain the variation of user-item interactions. We compare the resulting system to a baseline that does not model visual representations. People's taste changes over time and user preferences shift constantly, therefore an important assumption of this work is that user preferences remain constant over time (no temporal dynamics are involved). Briefly, we:

- Develop a ranking function by incorporating visual representations of the *grocery and food* items. We add visual signals to the model to obtain visually-aware recommendations
- Study the impact of the visual signals and show that such features can be leveraged to build a more interactive system for a better personalization
- Demonstrate the effectiveness of the visual features and reveal the superior performance of the system with visual representations integrated, through offline experiments

In the dataset of the study (*grocery and gourmet food* data – see section 6.4) ratings are available and user feedback is provided explicitly. Therefore, the task is to predict ratings of the unseen items and rank them to build a personalized list for each user, from which the recommendations are made. We mainly seek to: design a system that is scalable, personalized for users, and interpretable in terms of visual interests of the users. We seek to design a better system with better personalization capacity where the recommended items have a closer match with the user preferences.

4. Notation and Mathematical Symbols

We represent the user and item sets with U and W, and we reserve $i \in U$ and $j \in W$ indices for each element of the sets, i.e. for every single user and every single item, respectively. So, each user i rates a couple of items and hence is associated with a set of items W_i. Also, for each item $j \in W_i$, user i provides explicit feedback in the form of a rating which we represent with r_{ij}. Besides, for all the items $j \in W$ images are available, and their feature vectors are denoted by f_j. Throughout this work, the following notations are used to denote parameters and models mathematically.

Table 1. Symbols and notation

Notations	***Descriptions***
$\boldsymbol{K}$	Latent dimensions
$\boldsymbol{p}$	K-dimensional user factor
$\boldsymbol{q}$	K-dimensional item factor
$\boldsymbol{b}$	K-dimensional user bias
$\boldsymbol{c}$	K-dimensional item bias
$\boldsymbol{\mu}$	Global bias
$\boldsymbol{i}$	User index
$\boldsymbol{j}$	Item index

$\boldsymbol{\theta_i}$	Users' visual factor
$\boldsymbol{\theta_j}$	Items' visual factor
$\boldsymbol{\lambda}$	Regularization parameter
$\boldsymbol{E_j}$	Items' embedding matrix
$\boldsymbol{f_j}$	Image features from deep CNN
$\boldsymbol{\Theta}$	Parameters set of the models
$\boldsymbol{r_{ij}}$	Rating of the user i to item j
$\boldsymbol{\hat{r}_{ij}}$	Estimated rating of user i to item j
$\boldsymbol{\beta}$	Image bias
$\boldsymbol{\hat{x}_{ij}}$	Probability of item j being purchased by user i
$\boldsymbol{\mathcal{C}}$	Cost function
$\boldsymbol{\mathcal{L}}$	Loss function
$\boldsymbol{S}$	Set of (i, j) pair where the rating is available

5. Literature review

5.1. Recommender Systems and Matrix Factorization

A recommender system (RS) can be content-based, collaborative filtering based, or a hybrid of both. We begin by highlighting the preferences and types of feedback and then discussing content-based and then discuss collaborative filtering models.

User preferences are gathered through the feedback users provide explicitly (or implicitly) in the system. In explicit feedback, users can show their interests through a rating mechanism where they can directly express their opinion toward an item by assigning it a rating [6]. Figure 1 shows a few typical explicit feedback systems used by online services and social media platforms.

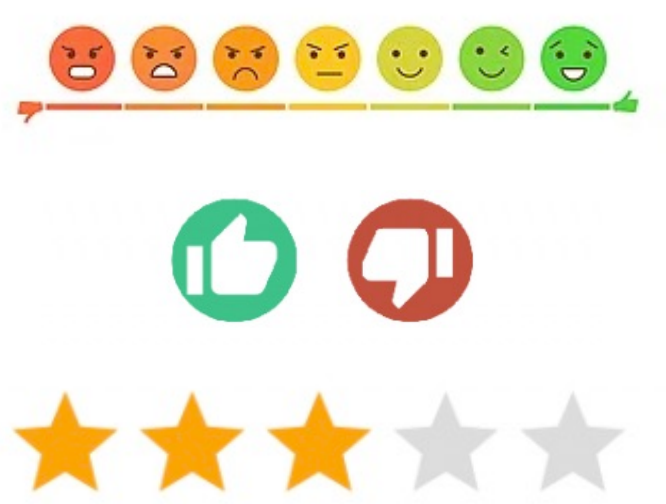


Figure 1. Typical explicit feedback systems

In implicit feedback, on the other hand, user interactions are assessed based on user behavior and their past actions, without direct intervention or feedback from the users. In this scenario, the label ranges from user clicks, purchase history, and search patterns including available user transactions with the system such as browsing history or even mouse movements [6][7]. Explicit feedback is harder to collect, and it can be biased because of personal judgment of different users on a product, whereas implicit feedback is easier to collect but it can sometimes be very noisy.

In content-based systems, features of the items are used to make recommendations. Those features are the item descriptions that are often assigned by the subject matter experts (SMEs) in the database or descriptions are provided upon signing up, for example. Such systems depend on the content features where automatic feature extraction is not widely available. The advantage of content-based systems is their capacity to handle cold-start problems, which are the challenge of recommending items to novel users, or recommending new items (whose contents are similar to the already liked items) to the corresponding existing users. However, their main drawback is their high dependency and reliance on having user or item features that are predictive of preferences. For instance, in the movie domain, scoring the visual effects, tone, rhythm, genre, montage, and emotional effects of the movies could be quite complex and need screenwriters or movie / TV experts to analyze and rate them. Those experts ought to formulate and feed the content features to the system manually, from time to time [3][4].

Collaborative filtering (CF), on the other hand, is a technique that involves collaboration between user opinions, viewpoints, or data sources for filtering the information for recommendation [5]. It is based on the assumption that users that have similar preferences in the past will have similar preferences in the future. For example, suppose that user A likes five movies and user B likes 6 movies including those five that user A likes, then it is very likely that user A likes the other (sixth) movie that user B has liked [17].

Collaborative filtering can be memory-based or model-based. Memory-based CF is called neighborhood-based systems where the concept of "similarity" is the key element. In the memory-based models, items are recommended because of being the neighbor of (similar to) another item(s), and the success of the system highly depends on the effectiveness of the implemented similarity function. Cosine, Jaccard, and Pearson similarity as well as *k*-Nearest Neighbor are the typical techniques in that category. Model-based methods, on the other hand, rely on the parameters of the models to recommend items where user and item inputs are fed into the models, the parameters are optimized, and interaction labels are estimated.

In the model-based recommendations, the problem is seen as a regression or classification task. Machine learning techniques are leveraged to discover the factors that explain the variability of the user-item interactions. However, regression and classification tasks for recommender systems are quite different from more traditional ones. In recommenders, we ought to tweak the regression and classification techniques to capture the dynamics of the user and item interactions. Instead of having parameters associated with the features, we would rather parametrize only the user or item features to optimize, which leads us to the notion of personalization. That way the system can be tuned to make personalized recommendations [8].

Among several model-based collaborative filtering techniques, matrix factorization (MF) is one of the most frequently used techniques for a recommendation which is also the area of focus in this project. MF is recognized as the most popular method which belongs to the Latent Factor Model (LFM) family. The popularity of matrix factorization is due to its flexibility and scalability. In general, LFMs try to explain the user ratings by characterizing the users and items through a few factors inferred from the feedback [9]. The basic form of the MF model looks like the following equation (Eq. 1) where p_i and q_j represent the user and item factor vectors, respectively, and interactions are modeled as their inner product [10].

$$\hat{r}_{ij} = {p_i}^T q_j \qquad \text{Eq. 1}$$

where $\hat{r}$ denotes the predicted ratings, and i and j denote their corresponding user or item indices.

Matrix factorization uses the users' explicit feedback for generating latent spaces of the user-item interactions. Once trained and optimized, the latent spaces form users and item attributes which are employed for the reconstruction of the rating scores. In such a setting, recommendation engines will be able to rank the items and personalize them for each individual user.

To estimate the parameters (latent factors), the model can be trained and optimized on regularized squared error of the observed ratings, which is called the cost function [10][11]. The regularization term helps prevent overfitting and it is useful in improving the performance of the model [10].

$$C = min\left(\sum_{i,j\in S}\left(r_{ij} - \hat{r}_{ij}\right)^2 + \lambda\left(\|p_i\|^2 + \left\|q_j\right\|^2\right)\right) \qquad \text{Eq. 2}$$

Here, S denotes the set of (i, j) pairs where user i has rated item j, and λ is the regularization parameter. This loss function ought to be minimized during the training process for designing an optimized model.

5.2. Matrix Factorization with Bias Terms

In recommender systems, user and item latent factors are just one of the components to form the rating variations. Other important components which also play a role are the user bias, the item bias, and the rating scale which is known as scaling bias. Users, items, and rating scales are independent of user and item latent factors, and they can generally get biased.

A simple explanation would be that some users tend to rate higher than others systematically and some items receive higher ratings than other items systematically. For the former (user bias), we could imagine the inherent social/moral characteristics of the users, i.e. how pessimist or optimist users are while rating the items. Suppose that user A is grumpy and rates the items with 2 stars on average, whereas user B is cheery and rates the items with 4 stars on average. Therefore, 3 stars rating of user A for an item has a different meaning than 3 stars rating of user B for that item. It means that user A liked the item, but user B did not. This is called user bias. A similar analogy is valid for the latter (item bias). Suppose that item C receives a 1-star rating on average whereas item D receives 5 stars on average. Similarly, 4 stars rating of item C is interpreted differently than 4 stars rating of item D[1] [12].

---

[1] https://github.com/wkirwin

Also, the rating scale and the fact that users are allowed to rate only between 0-5 stars impose another bias on the system called global (or rating) bias. Generally, keeping track of all these biases and involving them in the model make the recommender systems more performant [12][13].

Since biases are meant to capture a part of the observed interaction signals, precise inclusion of them in the model is essential. Therefore, a modified version of the MF model in Eq. 1 will have user, item, and global (rating) biases included, which looks like the following equation (Eq. 3) where $b_i \in \mathbb{R}$ and $c_j \in \mathbb{R}$ represent the user and item bias respectively, and $\mu \in \mathbb{R}$ represents global (rating) bias. Interactions are then expressed as the inner product of the user and item vectors plus bias terms [10][14].

$$\hat{r}_{ij} = {p_i}^T q_j \; + b_i + c_j + \mu \qquad \text{Eq. 3}$$

Here, the predicted rating is a combination of user factors, item factors, user bias, item bias, and rating bias. To optimize the model, the cost (regularized squared error) function of Eq. 2 is tweaked to include all the bias terms [15]:

$$\mathcal{C}^{modified} = min\left(\sum_{i,j\in S}\left(r_{ij} - \hat{r}_{ij}\right)^2 + \lambda\left(\|p_i\|^2 + \left\|q_j\right\|^2 + {b_i}^2 + {c_j}^2\right)\right) \qquad \text{Eq. 4}$$

5.3. Alternating Least Squares (ALS) Optimization

The most widely adopted optimization approach for solving matrix factorization is ALS [16]. Recommender systems are often designed to manage a huge number of users and an extensive number of items, which easily soars the number of user-item interactions up to a few billion. In such a setting, the Alternating Least Squares technique is an option for efficient optimization and for covering the heavy computation challenge [17].

ALS technique deals with updating and optimizing the user and item factors and biases intermittently over a loop, by keeping either of the parameters fixed, until the squared error over the observed target (cost function) is minimized. The algorithm alternately keeps one factor fixed and optimizes the other, hence it rotates between fixing user parameter(s) and recomputing the item parameter(s), and fixing item parameter(s) and recomputing the user parameter(s). Such a strategy is proven to converge and reach the optimal point i.e. reaches the minimum loss [8][10][16]. Once user factors or item factors are assumed fixed, the cost function turns into a quadratic equation that holds a global minimum, and it can easily be computed. Such an approach guarantees reaching the minimum point [17]. The mathematics of ALS and its equations have been addressed later in section 6.

Alternating Least Squares optimization is used in both explicit and implicit feedback systems. In explicit feedback, the algorithm often covers a sparse objective function since most of the items are usually unrated, whereas, in implicit feedback, the algorithm handles a dense objective function since user-item interactions are always inferred [18].

5.4. Modeling the Non-Visual Decision Factors

Standard matrix factorization with non-visual user-item interactions and bias terms, highlighted in Eq. 3 is used for modeling the non-visual factors. The parameter set to be learned from data is $\Theta = \{p_i, q_j, b_i, c_j\}$, and the ratings are modeled as interaction terms between user and item factors plus the user bias, item bias, and global bias (constant), all optimized through the ALS approach. Details and mathematics of the ALS technique as well as how the equations are derived for the user and item optimization are found in Appendix 1. Parameters are calculated through the following equations.

$$p_i = \left(\sum_{j \in W_i} q_j q_j^T + \lambda I\right)^{-1} \left(\sum_{j \in W_i} \left(r_{ij} - b_i - c_j - \mu\right) \left(q_j\right)\right) \quad \text{Eq. 5}$$

$$q_j = \left(\sum_{i \in U_j} p_i p_i^T + \lambda I\right)^{-1} \left(\sum_{i \in U_j} \left(r_{ij} - b_i - c_j - \mu\right) \left(p_i\right)\right) \quad \text{Eq. 6}$$

$$b_i = \left(\frac{1}{|\mathrm{W}| + \lambda}\right) \left(\sum_{j \in W_i} \left(r_{ij} - p_i^T q_j - c_j - \mu\right)\right) \quad \text{Eq. 7}$$

$$c_j = \left(\frac{1}{|\mathrm{U}| + \lambda}\right) \left(\sum_{i \in U_j} \left(r_{ij} - p_i^T q_j - b_i - \mu\right)\right) \quad \text{Eq. 8}$$

The optimized parameters are then used to estimate the unseen ratings according to $Eq. 3$.

5.5. Visual Signals in the Recommender Systems

Visual data has traditionally been used for personalizing image search and retrieval, but in different settings, they can also be incorporated into recommendation models for better efficiency. Visual data come in different modalities than interaction signals. They are often dense and high dimensional and cannot be handled by MF directly, so the way they should be incorporated into the model is a challenge [8].

By incorporating the image features, we model visual factors that might motivate the users. In recommender systems, models learn to discover important factors which drive the user decisions in making transactions. In some domains, visual features of the items are among the important factors that impact user decisions. An example is fashion products where the visual characteristics of the items affect the user decisions greatly (in general, most people wouldn't be comfortable buying a clothing item without seeing it first). Here, visual decision factors play an essential role in user decision-making.

In visually aware recommender systems, one possibility is to extract the visual signals directly from product images and incorporate them into the recommendation objectives for increasing the accuracy of the system [19]. However, there are a few challenges associated with modeling the visual decision factors. The first challenge belongs to the complexity of the factors involved. Extracting meaningful visual features as a visual portion of the decision information is very complex. The second challenge is the variety of visual tastes, i.e. individuals have distinct visual preferences which are completely personal. To provide good personalized recommendations, models should be able to capture and distinguish the unique visual tastes of their users. The third challenge is the evolving visual tastes over time. User preferences often change over time and as time progresses, visual decision factors evolve which make visual decisions continuously tied up to the temporal dynamics. The fourth challenge involves scaling the models to large-scale image data. The high dimensionality of the image data as well as the massive number of images involved in the modeling requires significant computational resources for model training. Finally, the last challenge to name here belongs to the difficulty in interpreting user decisions concerning visual and non-visual factors [20].

To obtain visually aware recommender systems, researchers augment their implicit feedback models. They suggest adding user-item visual interactions and visual bias terms to the standard matrix factorization equation to capture visual factors in the model [21][22]. In such a layout, recommender systems can predict the decisions based on both non-visual and visual factors:

$$\hat{x}_{ij} = \theta_i{}^T\theta_j + p_i{}^T q_j \ + b_i + c_j + \beta f_j{}^T + \mu \qquad \text{Eq. 9}$$

Here, the visual factor of the items (θ_j) is the explicit visual features, but to involve the human perception of the items' features, we must decompose it into the interaction between (latent) image features and user perception of these features. It shows how much each user is attracted to the extracted features [19][20][21].

$$\theta_j = E\ f_j \qquad \text{Eq. 10}$$

Image features f_j can be collected through a deep convolutional neural network (CNN), a widely used technique in the computer vision domain that has been proven efficient in capturing image characteristics [23][24].

The final form of the proposed equation for matrix factorization (static modeling) with implicit feedback data is

$$\hat{x}_{ij} = \theta_i{}^T\ (Ef_i) + p_i{}^T q_j \ + b_i + c_j + \beta^T f_j + \mu \qquad \text{Eq. 11}$$

where the optimized model predicts whether user i purchases item j and then a list of personalized recommendations is generated based on the predictions.

5.6. Data Split Strategies for Evaluating the Recommender System Designs

Any designed recommender system needs to be evaluated for efficiency and performance assessment, and data partitioning is one of the evaluation elements [25]. In recommender systems, like any machine learning problem, data ought to be split into different subsets for training, validation, and test purposes, but currently there is no single known procedure to follow for splitting the data. Unlike Information Retrieval (IR) and other machine learning domains, there is no specific guideline for data splitting in recommender systems. In RS, data partitioning is much more challenging than in typical machine learning tasks.

There are six possible strategies to split the data and for a single design, either of these strategies would end up reporting different performance [26]. The lack of standard guidelines for evaluation procedures in recommender systems is an ongoing challenge that raises uncertainty in making real progress for new RS designs. Without a unified/standardized evaluation setting, we cannot make sure that comparisons of the designs are apple to apple, despite using the same benchmarks [27]. Therefore, it is always necessary and important to explain the data partitioning strategy clearly or to share the split data.

5.6.1. Non-temporal split

Random split

The random partition strategy splits the items for each user, randomly. This is the older variant of the leave-one-last-item strategy (explained later) where one or few items are held out for validation and one or few are held for the test, all randomly. The random strategy used to be a typical approach in the early recommender systems where one of the items was often kept out randomly for the test. Such a strategy gradually faded out and was replaced by Leave-one-last items and Leave-one-last baskets. The advantage of this strategy is that the model is trained for all users but the main drawback is that the results are not reproducible unless the subsets are provided explicitly [26][28].

User split

In the user split strategy, subsets are built by splitting the data based on users instead of transactions, i.e. some users and the corresponding transactions are held for training, some other users and their transactions for validation, and the rest are kept for the test. This strategy is less commonly used mainly because it requires the models to have the possibility of supporting the cold start problem and making recommendations for new/unseen users, which most of the models do not [26][29].

5.6.2. *Temporal split*

Leave-one-last item

Temporal partitioning methods use the timeline of the events and split the data according to the transaction timestamps. Leave-one-last item is one of them. Such a strategy isolates the last transaction per user for the test, the second last transaction per user for validation, and the rest of the data are kept for model training. The advantage of this method is that the models are trained for all users, but the drawback is that there are a few items left for evaluation. This strategy is quite popular in the item-based recommendation models [26][30].

Leave-one-last basket

In such a strategy, for each user, the transaction of the last basket (session) is held out for the test, the second last transaction is kept for validation, and the rest is held out for model training purposes. This partitioning method is convenient when users make transactions for a basket of items in every purchase, so the user-item tuples would be like: $(user, [\,item\ 1, item\ 2, \ldots, item\ N])$. For instance, grocery shopping is the area where many items are bought together and users often make a transaction for a basket of items, therefore, the leave-one-last basket strategy could be the option for evaluating the recommendation models in that context [26][31].

Temporal user

In the temporal user strategy, for each user, a portion of the last purchased items or basket of items (e.g. 15%) is held out for the test, the second portion is isolated for validation, and the rest is taken for training. Since the transaction period of users is different, split boundaries are not identical for all users. In other words, each user has his own particular period as train/validation/test split boundaries, and researchers who design models with temporal variable(s) included must pay attention to this fact while setting boundaries for building the subsets [26][32].

Temporal global

In temporal global strategy, a fixed point in time is set for building the subsets. In other words, regardless of the users, fixed points on the time axis are chosen to define the borderline between train, validation, and test sets. Such a split is the most realistic setting where all the records are treated with the same temporal criteria, but the major drawback is that it is very likely that some users do not show up in all the subsets. Only some users might coexist in both the training and test sets. Because different users register at different times or start purchasing activity in different periods, it is possible that some users show up in the test set but do not exist in the train set, and vice versa. Hence, models with a such strategy might not get trained for all users, or evaluation may not cover the complete picture [26][33].

5.7. Common Evaluation Metrics for Recommender Models

The essence of having a clear evaluation methodology for recommendation models is undebatable. Depending on the type of user feedback (implicit or explicit), different metrics can be employed to measure the ranking performance of the systems.

Precision@N, *Recall@N*, and *MAP@N* are a few examples that can be used in implicit feedback models, and *NDCG@N* is the most popular one which can also be tuned for explicit feedback models along with *RMSE@N*.

RMSE is more convenient when the problem is regression whereas the rest are appropriate for decision support and ranking. Since recommendations are often ranked at service time, it is essential and makes more sense to use decision support or rank functions. The decision support metrics judge the systems based on the relevancy, i.e. they evaluate based on the number of relevant items in the list. They assign binary weights to the recommended items without taking the order of the recommended items into account. But in reality, top items often get more attention than the items at the bottom of the list. People tend to scan a list to specify their preferred item(s), and they usually feel exhausted and give up after a few scans. Therefore, high-quality systems are those which identify the top *N* preferences of the users and list them from top to bottom accurately to motivate them to make a transaction.

It is reasonable to assign more adaptive weights to measure the recommendations based on the order and rank of the items. Recommender systems are hence better evaluated based on the top *N* recommended items for each user (on the test set), and *NDCG@N* is the metric of choice which does such a job perfectly.

5.7.1. RMSE

Root Mean Squared Error is a typical metric for measuring the quality of an algorithm's outputs when the target variable is continuous. RMSE calculates the difference between the modeled and real values by penalizing the larger errors [34]. In models with explicit feedback, ratings are the target variable, and they are estimated (predicted) for the unseen items and so the Root Mean Squared Error could be the metric of choice for optimizing the model. It is often chosen as the objective function to minimize through iterations, like the objective that is used in this project for training the models.

However, high RMSE performance would not necessarily generate high recommendation (ranking) performance, therefore it is essential to include a ranking metric, explained in the subsequent section, to measure the ranking quality of the models [35][38]. RMSE itself has also the capacity to be used as a ranking metric where it measures the difference between estimated and observed ratings in the top-*N* recommended items, which in this case is called *RMSE@N*. This metric is however less accurate than the typical ranking metrics because of the lack of items' positional factors in the equation.

5.7.2. NDCG@N for explicit feedback

In personalization and personalized recommendation, it is ideal that for every user, we have the most preferred item at the beginning of the list, the second most preferred item in the second rank of the list, and so forth. Normalized discounted cumulative gain (*NDCG*) is a metric with the potential to show such behavior that can be used for both implicit and explicit feedback models. It penalizes wrong recommendations (and rewards the correct ones) more at the beginning than the end of the list, and that's why it is among the popular metrics for ranking assessment. It shows the relevance of the recommendations or the quality of rankings in other words [34][35].

NDCG is composed of two terms: 1) discounted cumulative gain (*DCG*) of the ranked items, which is defined as the sum of scores for all the recommended items weighted by a discount function to incorporate the item positions into the scoring, and 2) ideal discounted cumulative gain (*IDCG*) which is a normalization factor; it normalizes the

DCG to lie between $[0,1]$. The normalization term represents the maximum possible value that can be retrieved from the perfect ordering of the ranked items. In the explicit feedback models, *NDCG* is defined as [36][37]:

$$NDCG@N = Z_i \sum_{j=1}^{N} \frac{I(rank_j)}{log_2(1+j)} \qquad \text{Eq. 12}$$

In this equation, $I(rank_j)$ is an indicator function where for each user i with a list of recommendations, $I(rank_j) = r_{ij}$ if j is an interacted item, and $I(rank_j) = 0$ otherwise. Here, r_{ij} is the true rating of the item and Z_i is the normalization term which is defined as the inverse of *IDCG@N*.

$$Z_i = \frac{1}{IDCG@N} = \frac{1}{\left[\sum_{j=1}^{N} \frac{I(rank_j)}{log_2(1+j)}\right]_p} \qquad \text{Eq. 13}$$

$$NDCG@N = \frac{\sum_{j=1}^{N} \frac{I(rank_j)}{log_2(1+j)}}{\left[\sum_{j=1}^{N} \frac{I(rank_j)}{log_2(1+j)}\right]_p} \qquad \text{Eq. 14}$$

In the implicit feedback models, the evaluation term is similar except that the indicator function becomes $I = 1$ if the item is the interacted one, and $I = 0$ otherwise [36].

6. Methodology

In this project, we build a ranking system using a latent factor model, matrix factorization specifically. Among several model-based collaborative filtering techniques, matrix factorization is a widely used technique for recommender systems where user-item interactions are optimized into user and item factors. Those factors are harnessed for ranking and recommendations.

The ML model of this work is a regression that is trained using explicit feedback from the users. The estimated ratings in the output are consequently ranked and served for recommendations. The models are trained to learn both visual and non-visual parameters for rating estimations of those items that users haven't provided feedback on yet. We run experiments on the real-world dataset (explained later in section 6.4) to validate our hypothesis by comparing the baseline with the alternative model.

6.1. Baseline: Model with Non-Visual Factors

As the baseline of the experiments, standard matrix factorization with non-visual user-item interactions and bias terms which has been addressed in section 5.4 is used as a baseline. Ratings are modeled as an interaction between non-visual user and item factors plus a few bias terms which are optimized.

6.2. Alternative: Modeling the Visual Decision Factors

The alternative model should be able to read images effectively and learn the visual signals as well as non-visual factors for predicting user feedback or their decision patterns. A model comparable to the baseline is the matrix factorization of $Eq. 11$ which has visual factors on top of the non-visual factors and bias terms, optimized through ALS. Such a model formulates the visual signal of the items as well as the attraction of the users toward them.

As highlighted earlier, in this project we assume that the user's tendency and attraction toward visual features are static, and this assumption allows us to build static models for covering the objective of the work by learning the modest number of parameters. This section discusses the main contribution of this project. We develop a new MF model for validating the hypothesis by modeling the users' explicit feedback. We derive a new equation by incorporating the visual factors of the users and items into the baseline model. Then we train the new model and predict ratings of the unseen items.

It is worth emphasizing that we omit the image bias term of $Eq. 9$ to reduce the number of parameters as well as to accelerate the convergence of the model. Such omission is in agreement and conformity with the proposed approach by He et. al. (2016) for static modeling [20]. Here, the parameter set of $Eq. 9$ to be learned from data is $\Theta = \{\theta_i, E_j, p_i, q_j, b_i, c_j\}$, and ratings are modeled as visual and non-visual interaction terms between users and items plus user bias, item bias, and global offset (constant), optimized through ALS. Detailed mathematics of the ALS technique for modeling the non-visual factors and rating predictions plus how the equations are implemented for parameter tuning are found in Appendix 2. The parameters of the developed model are calculated through the following equations:

$$\theta_i = \left(\sum_{j\in W_i} \theta_j \theta_j^T + \lambda I\right)^{-1} \left(\sum_{j\in W_i} \left(r_{ij} - p_i^T q_j - b_i - c_j - \mu\right) \left(\theta_j\right)\right) \qquad \text{Eq. 9}$$

$$E_j = \left(f_j \sum_{i\in U_j} \theta_i \theta_i^T + \lambda I\right)^{-1} \left(\sum_{i\in U_j} \left(r_{ij} - p_i^T q_j - b_i - c_j - \mu\right) \left(\theta_i\right)\right) \qquad \text{Eq. 16}$$

$$p_i = \left(\sum_{j\in W_i} q_j q_j^T + \lambda I\right)^{-1} \left(\sum_{j\in W_i} \left(r_{ij} - \theta_i^T E_j f_j - b_i - c_j - \mu\right) \left(q_j\right)\right) \qquad \text{Eq. 10}$$

$$q_j = \left(\sum_{i\in U_j} p_i p_i^T + \lambda I\right)^{-1} \left(\sum_{i\in U_j} \left(r_{ij} - \theta_i^T E_j f_j - b_i - c_j - \mu\right) \left(p_i\right)\right) \qquad \text{Eq. 11}$$

$$b_i = \left(\frac{1}{|\mathrm{W}| + \lambda}\right) \left(\sum_{j\in W_i} \left(r_{ij} - \theta_i^T E_j f_j - p_i^T q_j - c_j - \mu\right)\right) \qquad \text{Eq. 12}$$

$$c_j = \left(\frac{1}{|\mathrm{U}| + \lambda}\right) \left(\sum_{i\in U_j} \left(r_{ij} - \theta_i^T E_j f_j - p_i^T q_j - b_i - \mu\right)\right) \qquad \text{Eq. 13}$$

The optimized parameters are then used to make new rating estimations according to the following equation:

$$\hat{r}_{ij} = {\theta_i}^T \left(E f_j\right) + {p_i}^T q_j + b_i + c_j + \mu \qquad \text{Eq. 14}$$

6.3. Personalized Ranking Procedure

The goal of this work is to create personalized lists for each user i based on the unseen items. Once the parameters are optimized through training the adopted MF models on RMSE (described in sections 5.4, 6.2, and 5.7.1), the ratings of the unseen items are estimated by employing the parameters and feature vectors associated with each user i and item j, they are subsequently scaled to lie properly between the rating intervals, i.e. $[0,5]$, and final lists are then created by sorting the estimated rating of the unseen items for each user from the highest to the lowest. The top 20 items of the lists (for each user i) are then assigned to the list of personalized recommendations, and the ranking performance is finally evaluated and reported by *NDCG@N* (explained in section 5.7.2).

6.4. Dataset

The data of this study is a real and public dataset from *Amazon.com*. It is available on the repository of recommender systems datasets[2] and the recommendation models can be built using a large crawl of product reviews where users have explicitly rated the *grocery and gourmet food* items on the Amazon website. The whole data consists of product reviews and metadata with the date range from May 1996 (beginning of Amazon's e-commerce activity[3]) to July 2014. The product review holds user-item ratings, review text, and helpfulness votes, and the metadata covers the product descriptions, category information, price, link to the product page, and image features.

A small subset of this data is also available with 40 million reviews and ratings and 10 *GB* of data size in total, called *5-cores* where original data has been downsized in a way that each user has reviewed and rated at least five items and each item has at least five users reviewed and rated it. This is a much cleaner and more flexible version of the original data for modeling the recommenders, which is used as the data for our experiments. Also, this small subset is managed to be available for download by product category (to skip downloading unnecessary or irrelevant data), which in this work is *grocery and gourmet food*. Purchase of the food items depends on the attraction of the users towards the food images. The availability of a wide range of food images plus diversified packaging design and photography styles for each particular grocery and food item were the driving force and reasons why this category was chosen for this experiment. The product review data and metadata have the following schema:

---

[2] http://jmcauley.ucsd.edu/data/amazon/links.html
[3] https://www.techtarget.com/whatis/definition/Amazon

Table 2. Schema of the raw data: product review and metadata

Product Review
reviewerID
asin
reviewerName
helpful
reviewText
overall
summary
unixReviewTime
reviewTime

Metadata
asin
title
price
imUrl
related
salesRank
brand
categories

Here *reviewerID* represents user ID, *asin* represents item ID, *reviewerName* is the name of the users, *helpful* stands for helpfulness score of the review, *reviewText* is the text of the review by users, *overall* is the item's rating, *summary* represents the review summary for the item, *unixReviewTime* is the Unix time of the review, and *review Time* is the review time in the datetime format, *title* is the name of the item, *price* is the purchase price at the time of the order, *imUrl* is the link to the item's image, *related* represents other items viewed or purchased by the user, and finally *salesRank*, *brand* and *categories* are the sales rank information, item brand name, and product category, respectively. A few examples of the product review raw data and product metadata are shown in Appendix 3 and Appendix 4.

Visual features of the images are also available for download and integration into the product data. The features are ready to serve, already extracted by deep convolutional neural networks (CNN) as described in J. McAuley et. al. (2015)[4] and R. He, J. McAuley (2016).[5]

The image features consist of a vector of 4,096 dimensions. It is stored in a binary format right beside the 10 characters' product ID. The visual features have 140 *GB* in total size, and they are ready to feed into the models, but initially, they ought to be pre-processed so that they can merge into the corresponding items in the dataset. For visibility and rapid monitoring, the image of the items can be downloaded and shown through the *imUrl* links on the product review data. A few examples of the product images are shown in Appendix 5 and section 7.2.

6.5. Data Split Strategies for Model Training & Performance Evaluations

In this project, two partitioning strategies have been followed: random split (RNDM) and temporal global (TMPG) (see section 5.6) section for details). In the former (RNDM), items were first shuffled for each user and 20% of them were randomly selected and held out for validation and 10% were kept for the test, and the 70% remaining were used for training the models. But in the latter (TMPG), for a more realistic recommendation scenario that corresponds to the market time of the items, two fixed points on the time axis were set as the split boundaries. The first boundary is the border between the train and validation sets where the left side of it belongs to the training set which is used for training the models, and the right side (up to the second boundary) belongs to the validation set. The second boundary is the border between validation and test subsets.

[4] Image-based Recommendations on Styles and Substitutes [39]

[5] Ups and Downs: Modelling the Visual Evolution of Fashion Trends with One-Class Collaborative Filtering [40]

These boundaries are chosen based on intuition from data, and often data analysis helps to better tune the boundaries. In this project, the market time of the items which is analyzed and explained on page 37, helps define the boundaries. The selected boundaries based on the analysis are fully addressed in section 7.3.

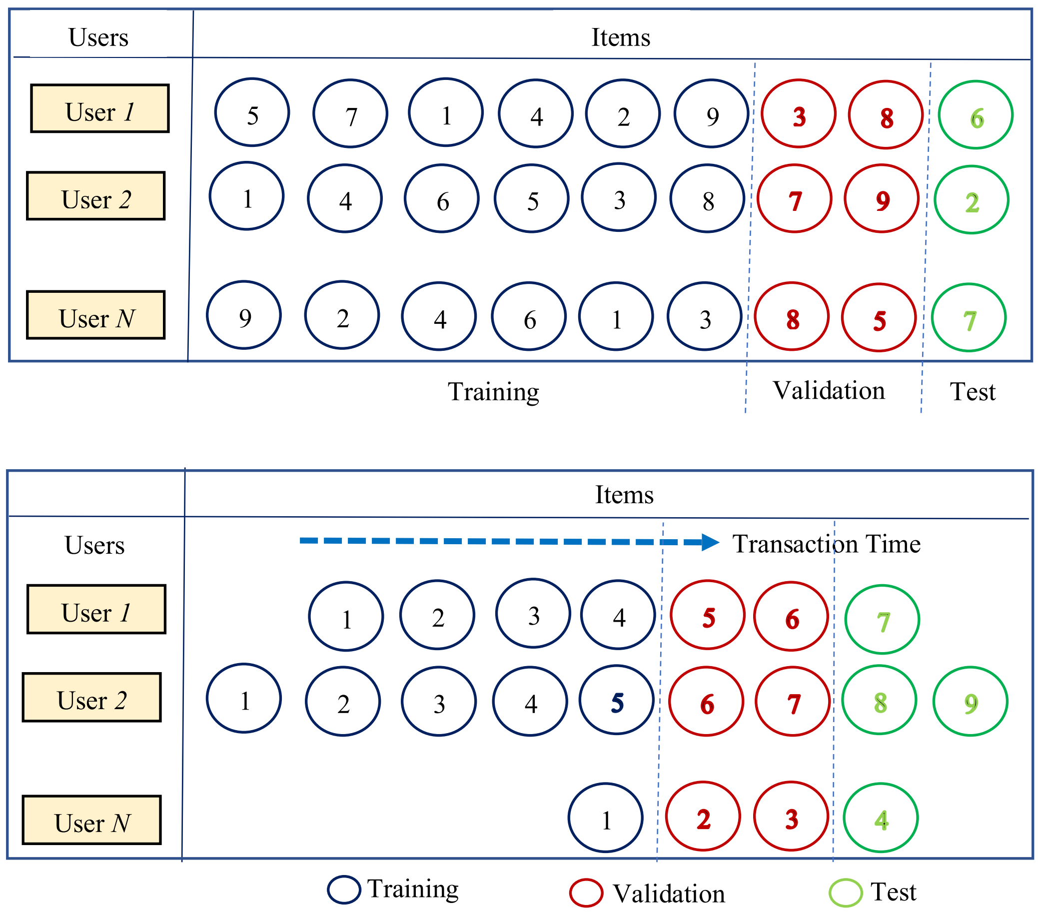


Figure 2. Data split strategies: RNDM and TMPG

7. Results and Discussions

7.1. Analysis of the Input Data

7.1.1. Statistics

Statistics of the 5-core *grocery and gourmet food* category of the Amazon data which is used for this project are addressed in Table 3. The data is a subset of the whole data explained earlier in the Dataset section (section 6.4), and for this category, the data is available from August 2003 when the gourmet food business line was added to Amazon's e-commerce platform[6] (only a very few records) until July 2014 (plenty of records).

Table 3. Statistics of the Amazon grocery and gourmet food dataset

Number of unique users	14,681
Number of unique items	8,713
Number of ratings	151,254
Median ratings per user	7
Average ratings per user	17
Sparsity of the data	0.12%
Average rating value	4.2
Ratings' date range	2003-2014

[6] https://www.cbsnews.com/news/20-years-of-amazons-expansive-evolution/

The median value of the ratings per user is a better representative of the number of ratings per user. Since most users have rated 5 items and only a few users have rated too many items, the distribution is skewed, and hence median value is a better representative. 4,194 users rated the items 5 times whereas there are only 632 users who rated the items 10 items. Moving forward, only 238 users have rated 15 items and only 104 users have rated 20 items. Therefore, the median value is a better option for reporting the average value here.

Besides, there are 722 unique items with missing visual signals. The reason why those feature lists are missing is unknown. The publisher of the data has not made any clarifying report about it. These items receive 12,590 ratings in total which represents 12% of the 5-core *grocery and gourmet food* ratings. Since the objective of the work is to compare the baseline model without image features with the alternative that has image features included, it is mandatory to drop the items without visual signals from the beginning so that the comparisons are carried out on an equal basis. Therefore, the final data which is fed into the models contains 7,991 unique items with 138,664 ratings in total.

The distribution of the rating scores skews toward the highest score. Most of the ratings in the original dataset hold very high scores which are fairly typical in the realm of recommender systems [41]. 80,147 ratings receive 5 stars ratings from the users, which represents 57% of the data. There are then 29,939 ratings that have received 4 stars rating which contributes to the second largest segment with 22% of the data, and then 16,105 have received a score of 3, and 7,213 and 5,260 have received scores of 2 and 1, respectively. Figure 3 shows the distribution of the pre-processed data which shows 79% of the data have very high ratings (4 and 5 stars). Such skew causes tough competition between the items for personalization (i.e. being selected and recommended by the model) since there are too many high-rated items to be modeled.

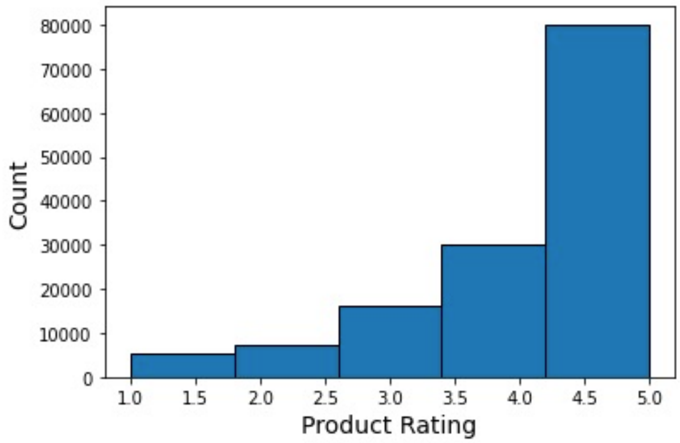


Figure 3. Distribution of the rating classes before splitting

In addition, the number of rated items is distributed unequally among the review years. There is a very limited number of items rated in the first few years where there is only one record for 2003 and less than a hundred rating records per year until 2005. The number of ratings rises gradually going forward. A breakdown of the number of ratings per year has been reported in Figure 4.

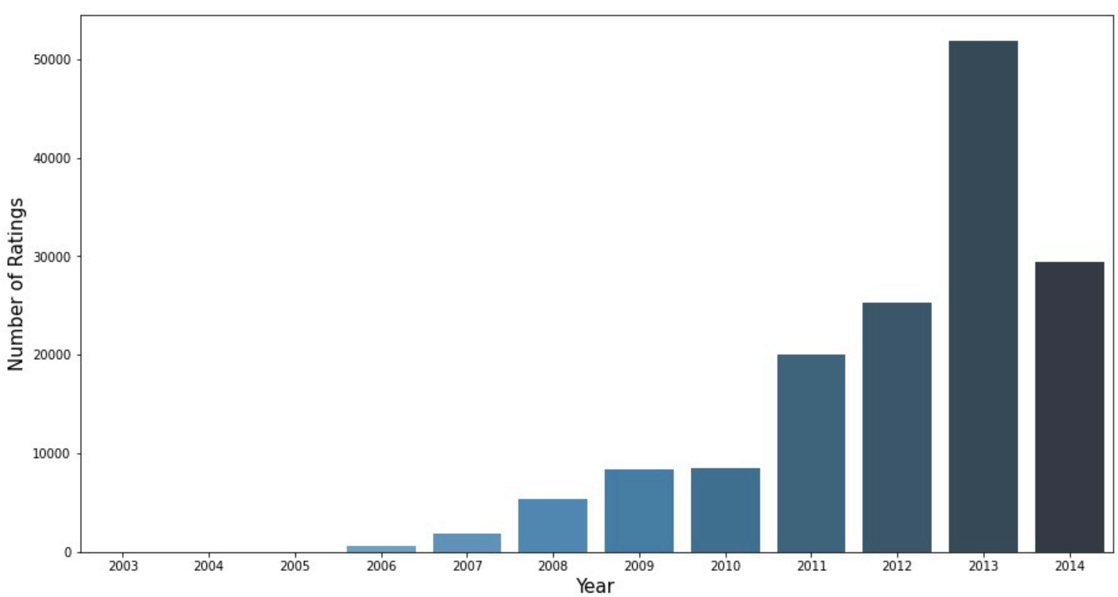


Figure 4. Distribution of the ratings during the years

The distribution of the rated items in any year follows more or less the same pattern as the original dataset (shown in Figure 3), i.e. for any year the rating distributions skew towards the highest scores like the original data. For any year, most of the ratings belong to the 4- and 5-star categories as shown in Figure 5. Therefore, competition between the items for personalization which was highlighted earlier exists regardless of the review year.

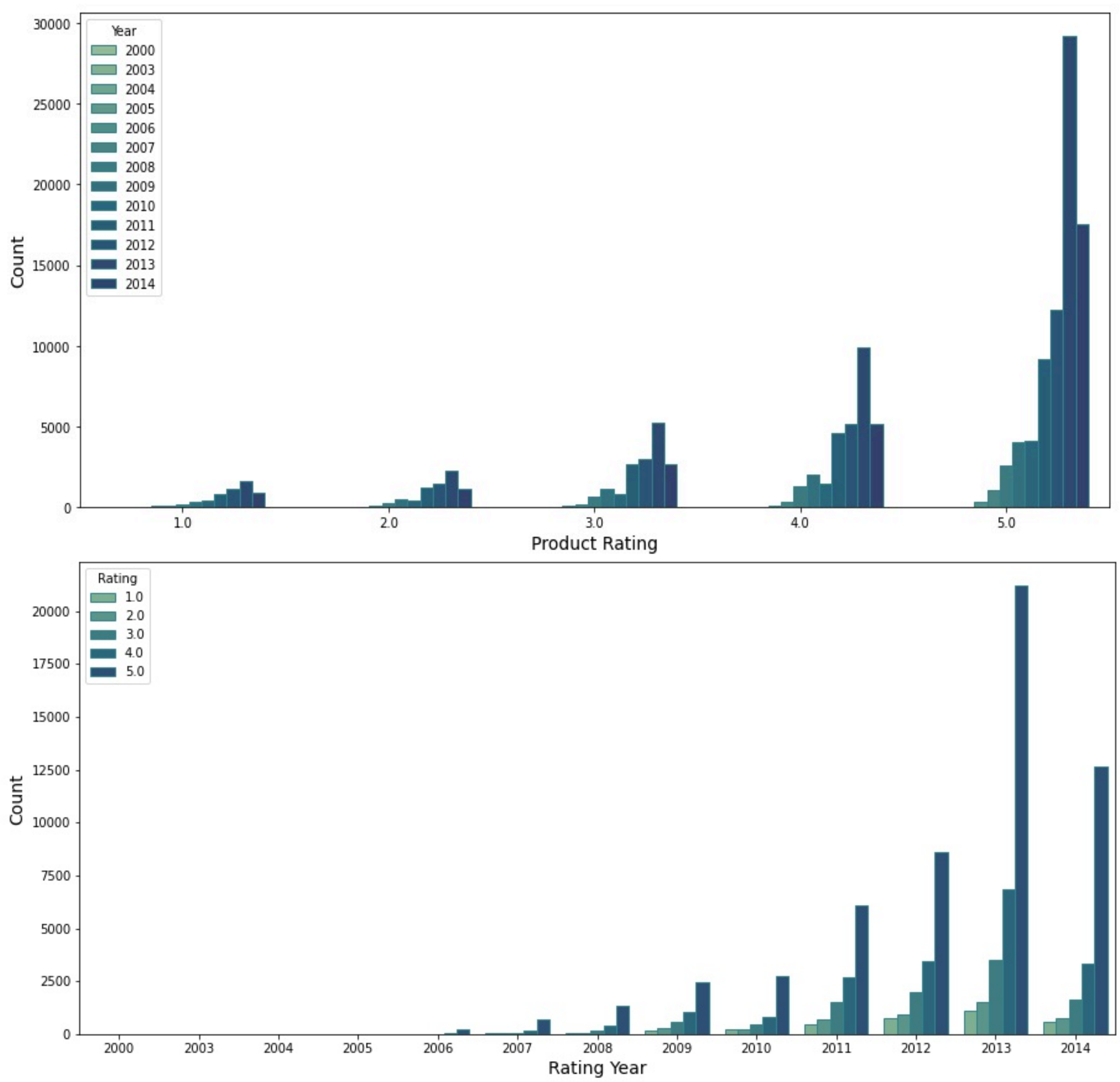


Figure 5. Distribution of the rating classes during the review years

7.1.2. Popularity Assessment: Long-Tail

Popular items are competitive products that are often purchased automatically and regularly. Popular products are sold frequently whereas unpopular items are not often seen and hence not sold frequently. Unpopular items (on the long tail) usually return higher profits than popular items and they have a high potential for being explored and discovered[42][43]. Designing recommender systems that suggest the items on the long tail is more favorable which also respects the diversity and increases the profit. It is evident that keep recommending popular items makes users fed up after some time, and encourages them to churn. Therefore, the algorithms must diversify the recommendations.

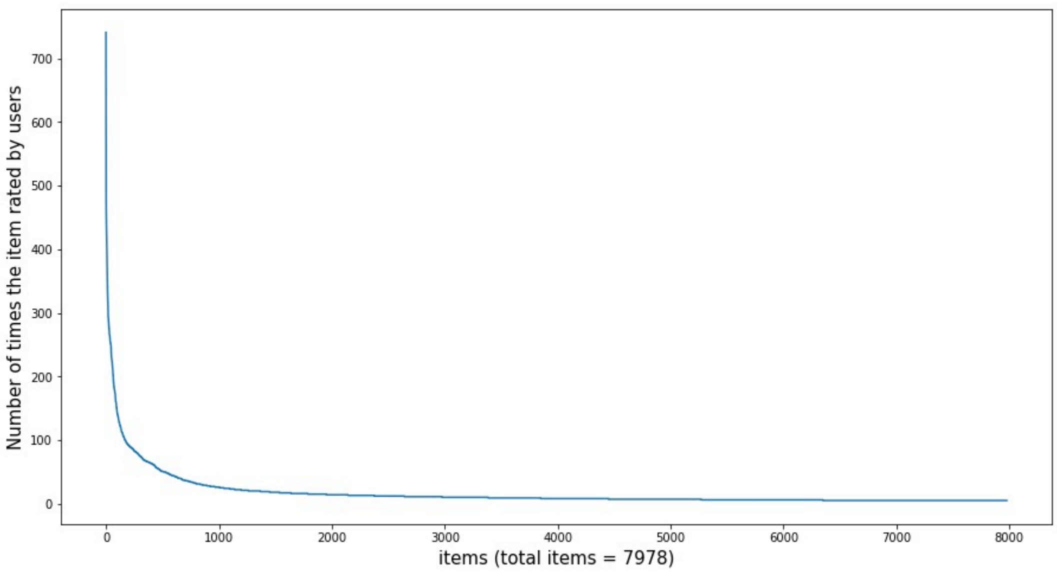


Figure 6. Long tail graph of the grocery and gourmet food items

Models that mitigate the recommendation of popular items (heavy tail) contribute to sales and sales margins. In training the recommender systems, it is preferred to have less popular items in the data so that items on the long tail to be explored and discovered. In Amazon's *grocery and gourmet food* data, most of the items are sitting on the long tail and hence they are suitable for exploring and discovering. As depicted in Figure 6, the sharp drop on the left shows that there are only a few popular items whereas the flat area on the right suggests that most of the items are seated on the long tail. In this graph, the horizontal axis represents items and the vertical axis represents the number of times the items have been rated.

If we admitted the popularity boundary to be in the middle of the curve (50 ratings) then there would only be 500 popular items. In other words, there would only be 500 items that were rated more than 50 times, which represents 6.2% of the items. The most popular item has been rated 741 times (on the top of the graph) and the least popular items have been rated 5 times which is expected (see section 7.1.1).

7.1.3. Life-cycle analysis: Availability of the items in the market

Performing analysis for the availability of the items in the market enables us to understand better the data that we are using to train and evaluate the models. The essence of life-cycle analysis is mainly because of the wide timespan for review dates. Since we have 14 years of data and most of them are supposed to be fed into the models for training and performance assessment purposes, it is essential to figure out how long the items would be available and remain in the market. It is useless to train a model for items with a short lifetime in the market because they will no longer exist to be purchased by users anyway, even if the model recommends them.

Minimum and maximum dates of each *grocery and gourmet food* item are used for life cycle assessment. Despite having 14 years of data, as shown in the boxplot of Figure 7, only a few items have very long availability and existence in the market (around 10 years and more). Here, the 4th quartile represents 55-110 months (4.5 to 10 years), the 3rd quartile is around 3 – 4.5 years, the 2nd quartile represents 1.5 – 3 years in the market, and the 1st quartile represents the market time of fewer than 1.5 years. The median and mean market times are also 33 months (2.7 years) and 38 months (3.2 years), respectively. However, a better picture of the lifecycle is given by the violin plot in Figure 7 which shows that the distribution of the market time does not follow a normal curve. Consequently, it is hard to admit the middle (median) value of the market time as a representative of the whole data.

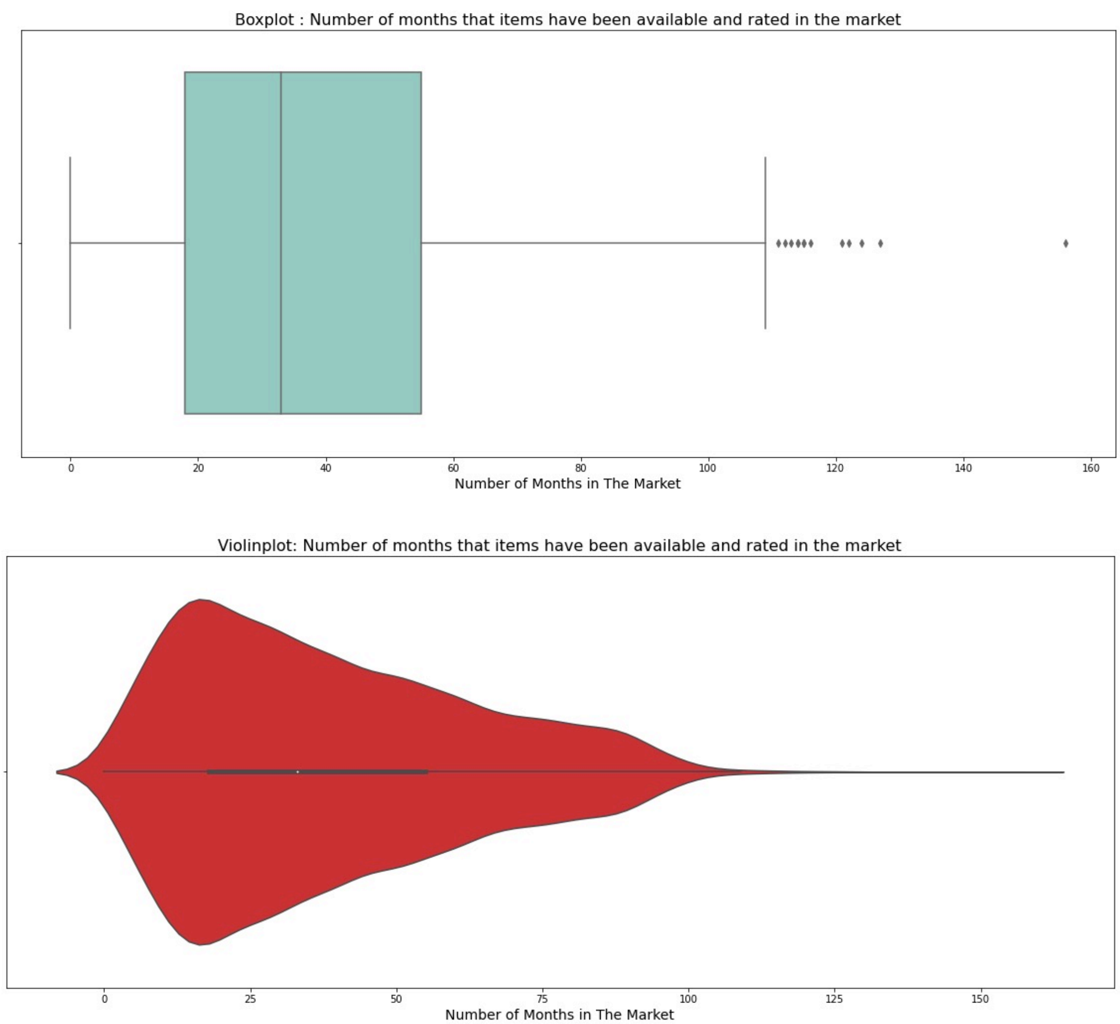


Figure 7. Market time of the grocery and gourmet food items

We observe that the highest occurrence frequency takes place near the first quartile where most items stay in the market for about 18 months (1.5 years). This number seems to be a better representative of the *grocery and gourmet food* lifecycle which shows most of the items in our research stay in the market for only 1.5 years. In other words, despite having 14 years of data, most of the items disappear within a short period and are replaced with new ones in the market.

7.1.4. Analysis of the image similarities using k-NN

Similarity analysis shows that the image attributes and features are perfectly captured by the CNN model. Analysis of five randomly selected images reveals that the 4096 dimensions of the items, given on the feature list, are tuned flawlessly to extract the image features of the *grocery and gourmet food* category. These feature vectors are representative of the images.

The k-nearest neighbor technique[7] with the K-D Tree (K-dimensional tree) algorithm[8] and the Euclidean distance metric is used for the analysis of the neighbor images. K-D Tree is a binary tree data structure that partitions the parameters recurrently and places the data into nested regions. It offers faster computation and lower runtime due to the lower complexity order of the algorithm [44].

The five nearest neighbors of the reference images are sought, and their images are retrieved to compare. Item 7091 (asin = B008269HGW) is the first randomly selected

[7] https://scikit-learn.org/stable/modules/neighbors.html
[8] https://scikit-learn.org/stable/modules/neighbors.html#nearest-neighbor-algorithms

item from the list of *grocery and gourmet food* category. It is Irish Oatmeal with vertical rectangular packaging having an elaborated design on the front (Figure 8).

Figure 8. A randomly selected (reference) item from the grocery and gourmet food category (Irish Oatmeal)

Figure 9. The five nearest neighbors to the random image selected

This item belongs to the "McCann's" company which is in the oat business since a long time ago and offers several oatmeal products in the market[9]. Other products of this brand also exist among the items of the *grocery and gourmet food* category in the Amazon

[9] https://mccanns.com/products/

dataset. The image of the reference item is designed to depict the brand name on top, the product name inside a hyperbole shape in the middle, ingredients with a smaller font size underneath the product name, and finally a commercial photo of the product at the bottom.

It is observed that four of the five most similar images to Irish Oatmeal are the other oatmeal products of the McCann's brand with similar face design; the fifth similar image is an identical product (i.e. oatmeal) of another brand with more or less the same face design. The distances of the retrieved images (to the reference image) are reported in Table 4, and the corresponding images are found in Figure 9.

Analysis of the other four randomly selected images shows that all 5 neighbors (similar) images hold more or less identical characteristics and features of the corresponding reference images. The random and the neighbor images can all be found the Appendix 9. It is evident that the 4096 dimensions of the feature lists cover the attributes and specifications of the reference image, perfectly.

Table 4. Distance of the similar images to the reference item

Item index	7092	2988	2890	7090	2989
Product asin	B008269HJY	B001EO5QW8	B001E5E3SU	B008269HC6	B000KNB0OW
Similarity rank	1	2	3	4	5
Distance	42.9	49.2	50.4	50.5	60.1

7.2. Sample Images

Several examples of the *grocery and gourmet food* images that are used in this study are shown in Appendix 5 and a few more samples are depicted in Figure 10 (numbers on the images are artifact and not parts of the images). As explained earlier in section 3, we aim to improve the model and predict the decisions more accurately by incorporating the features already extracted from the images.

For each category of grocery and food items, different products with various packaging designs and photography settings are available, which give distinct impressions to the users for buying. The image settings of the items are assumed to be persuasive for the users, which we believe to correlate with their decisions, as explained in the hypothesis. We believe that the CNN features capture the sophistication of the images and hence they can help improve the recommendations.

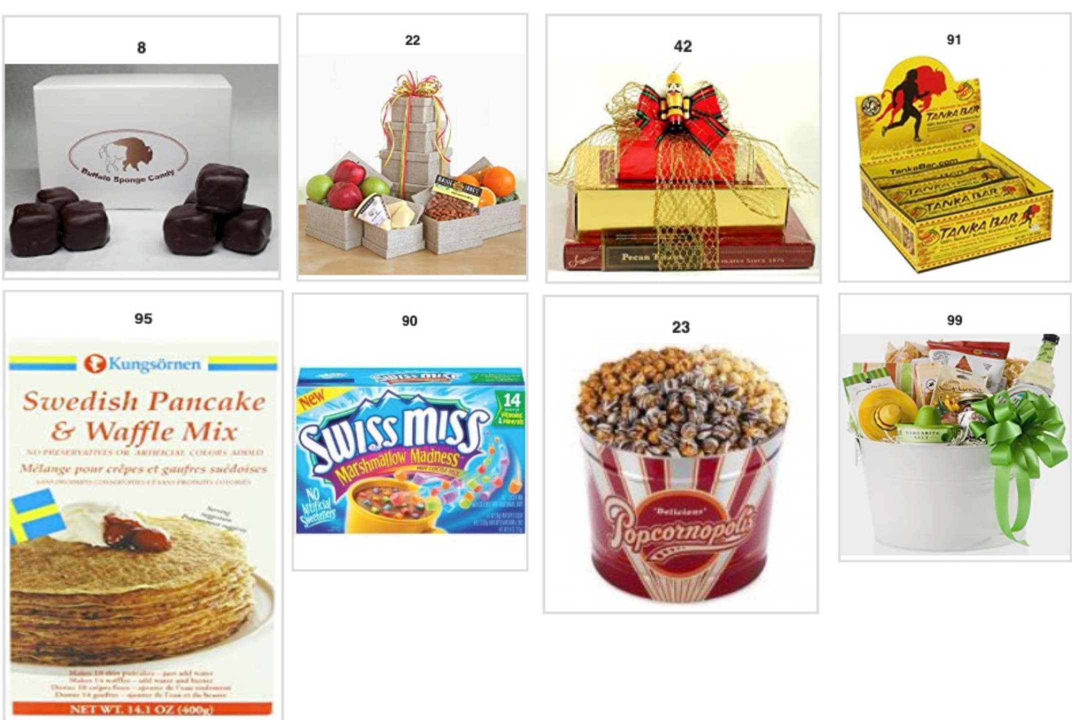


Figure 10. Sample Images of Amazon's grocery and food items

7.3. Train / Validation / Test Subsets

Random (RNDM) and temporal global (TMPG) split strategies have been used to partition the data for training and evaluations as highlighted in section 6.5. Random split ignores the datetime factor whereas temporal global employs it for separation. In RNDM, the rated items for each user are initially isolated, then shuffled, split, and finally saved into the relevant subsets. All the steps are carried out in a loop. For every single user, the number of rated items is counted and after shuffling, 10% of them are saved for the test, 70% of the remaining data is kept for training and the other 30% is dumped into the validation subset.

In the TMPG strategy, review dates are the main elements for drawing lines between the subsets. From Table 5, we observe that the rating data become quite stable after 2009, therefore this year is considered the starting point of our TMPG split for the experiments. In addition, from Figure 7 and the lifecycle-analysis section, we observe that the market time is limited, and items gradually disappear from the market after 18 months. Therefore, 18 months period from July 2009 to December 2010 were used for training the model, the next 6 months from January 2011 to June 2011 were held for validation and model selection, and the next 6 month from July to December 2011 were used as a test set for reporting.

Major drawbacks of the TMPG strategy are that users and items are distributed unevenly among the subsets plus there are unequal data points within the months and years. In other words, some users (and/or some items) might exist much more in one year and much less in another, and some of them are lost within the subsets. That can impact the evaluation step. Some users start/stop their activities at a certain time (including rating the items), so they may exist in the validation or test subsets but may not be on the training set (or vice-versa). Similarly, some items emerge at a specific date and exist on the validation or test sets but do not show up on the training subset, hence, they cannot be trained by the models to be recommended at all. To overcome such a problem, before training the models, users and items which exist on the validation and test sets but do not exist on the training set, are initially removed. Such slicing and dicing allow us to maximize the training capacity for *grocery and gourmet food* data. Statistics of the subsets have been addressed in Table 5 below.

Table 5. Statistics of the train, validation, and test subsets

	RNDM	**TMPG**
Date range covered	2003-2014	2009-2011
# Ratings - training subset	75,452	10,186
# Ratings - validation subset	43,004	9,740
# Ratings - test subset	20,208	8,548
# Unique users – training set	14,681	4,033
# Unique users – validation set	14,681	2,004
# Unique users – test set	14,681	1,878
# Unique items – training set	7,914	3,075
# Unique items – validation set	7,509	1,689
# Unique items – test set	6,040	1,551

In RNDM, there are roughly 1000 fewer unique items in the test set than in the validation set. This shortage happens mainly because users do not have an equal number of ratings, which makes the reported performance number stay lower than the performance number from the validation phase where we select the best models.

7.4. Models and Scenarios

Three recommendation scenarios are studied and addressed: 1) random recommendations 2) baseline 3) visual (VL) where the rating prediction task is modeled by latent factors plus the visual contents of the items. Random recommendation (RR) personalizes the items randomly which makes us ensure that the baseline models outperform the recommendation made randomly. Baseline (BL) is the personalization with biased matrix factorization technique where the rating prediction task is modeled by latent user and item factors and predicted ratings are computed by equations addressed in section 5.4. The baseline is adopted from the proposed approach by Pero et. al. (2013) [45] where they have used a similar baseline for their work. Finally, the visual (VL) scenario is the alternative recommendation model that uses matrix factorization with image features of the items included. Here the rating prediction task is modeled by interactions between latent user and item factors as well as the visual factors.

In the VL scenario, visual and non-visual decision factors are modeled (see section 6) where two dimensionality reduction techniques are leveraged to reduce the original dimension of the high-dimensional image space: principal component analysis (PCA) which is a linear technique, and t-distributed stochastic neighbor embedding (t-SNE) which is a non-linear technique that aims at keeping neighbors close in the lower space. Original image features extracted from the CNN model had 4096 dimensions which need to be reduced for meeting the objective of the visual (VL) scenario that employs Eq. 15 – Eq. 21.

Parameters of the models are optimized and tuned through Eq. 5- Eq. 8 for the BL scenario, and through Eq. 15 – Eq. 21 for the VL scenario, by trying different latent dimensionalities (K) and regularizations (λ). Zhang Sh. designed their experiments for latent factors between 10 and 130 [13], Rendle et. al. proposed the same range for the latent dimensionality for their experiments [31], and Liang et. al. set the dimension of the latent representation K to $\{100, 200\}$ in their experiments [32]. We use the latent dimensionalities reported in the literature and extend the range to conduct our experiments for $K = \{5,\ 10,\ 20,\ 50,\ 100,\ 200, and\ 300\}$. Also, the regularization parameter of the experiments λ is tuned to be among $\{10,25,50\}$.

The models are trained for 8 epochs using training subsets where parameters were gradually optimized throughout the iterations. Ranking performances are then evaluated on the validation subsets. The best models are nominated and selected according to the ranking performance on the validation sets, but they are reported based on the test subsets. To make sure that the higher performances are not occurred by chance, each experiment setting is repeated five times and the reported numbers are the average of the five repeated experiments.

For random recommendations (RR), the estimated rating matrix ($\hat{R}$) is constructed randomly i.e. the unseen items are randomly rated between 0 – 5, and the list of personalized recommendations (top 20 items) is generated for each user based on that, and finally, the ranking quality is evaluated and reported by the corresponding validation and test subsets, respectively.

7.5. Results

7.5.1. Ranking performance of the scenarios

In this project, we try to personalize the items for each user. We create a list of the top 20 Amazon *grocery and gourmet food* for each user. Personalization takes place by sorting the users' predicted preferences estimated by the matrix factorization models. we use *RMSE* to optimize the baseline (BL) and visual (VL) models and to evaluate the ranking quality of the scenarios we use *NDCG@20* which is explained in section 5.7 earlier.

Ranking performances plus the improvements and the corresponding hyper-parameter of the models in which the performances have occurred are reported in Table 6 and Table 7. Each reported *NDCG@20* is the average of the same experiment repeated 5 times (including random scenario - RR), to make sure that the observed improvements have not occurred by chance.

Table 6. Ranking performance of the temporal global strategy (TMPG) – Test subset

	RR	**TMPG-BL**	**VL-PCA**	**VL-tSNE**
$NDCG@20$	0.97e-3	1.04e-3 (6.6%)	1.61e-3 (54.8%)	1.26e-3 (21.2%)
K	-	10	20	200
λ	-	10	10	10

Table 7. Ranking performance of the random split strategy (RNDM) – Test subset

	RR	**RNDM-BL**	**VL-PCA**	**VL-tSNE**
$NDCG@20$	1.55e-3	11.21e-3 (621%)	8.61e-3 (-23.2%)	9.42e-3 (-16%)
K	-	20	10	100
λ	-	10	10	10

As we observe in the tables, all the baseline and alternative models outperform the random recommendations. Their ranking performances are systematically higher which means that personalization through matrix factorization models of this project is always superior to random personalization, without any exception. Ranking performance for the TMPG-VL-PCA model is 54.8% higher than its baseline (TMPG-BL) which has occurred at $K = 10$ and $\lambda = 10$, and the TMPG-VL-tSNE model has shown 21.2% improvements to the baseline which occurred at $K = 200$ and $\lambda = 10$. The results in Table 6 show that incorporating visual signals into MF models and involving visual decision factors in a correct setting can enhance personalization greatly, and it ultimately impacts user decisions positively to make purchases.

Table 6 shows that although implementation of both PCA and t-SNE techniques in the visual scenario has improved the recommendations, employing PCA performs best and improves the ranking quality more than t-SNE. The t-SNE method uses a higher latent dimensionality (K) to outperform the baseline compared to the using PCA. Besides, for the data used in these experiments, $\lambda = 10$ is the most appropriate regularization where all the best performances have occurred at that point.

Popularity analysis of the recommended items in the TMPG strategy reveals that in the base model (TMPG-BL), 53.6% of the recommendations belong to the "very popular" category whereas 34% of the recommended items are unpopular items. In the visual models, these percentages change in favor of the unpopular items. In the VL-PCA model, only 17% of the recommendations belong to the "very popular" items whereas the percentage of the unpopular recommended items increased to 67%. In the VL-tSNE model, the percentage of the unpopular recommended items is 99.5% and the rest belongs to the "fairly popular" group. The popularity boundary and percentages of the recommended items in each category are reported in Table 8, which shows visual models (with higher performance) make recommendations in favor of the unpopular items. It shows that the visual models stick to a different signal than the base model so it is very useful for applications such as cold start.

Table 8. Popularity analysis of the recommended items in the TMPG strategy

	Boundary [10]	**TMPG-BL**	**VL-PCA**	**VL-tSNE**
Unpopular	r <= 50	32.9%	65.8%	99.5%
Fairly popular	50 < r <= 250	13.4%	17.3%	0.5%
Very popular	r > 250	53.5%	16.8%	0%

Table 7 shows the deterioration of the performance by the visual models, which is mainly because most of the items disappear from the catalog within a short period, i.e. after roughly 1.5 years (Figure 7); but visual models are neither designed to absorb and understand it during the training nor the data split supports it. Generally, recommendations must be made within a reasonable period so that the items still exist in the market. This not only can end up with a more realistic personalization, but also gives a better outcome. However, the RNDM strategy splits the data randomly and does not

[10] See the Popularity Assessment: Long-Tail section on page 36 for the details

take the date order or lifecycle of the items into account, which leads to performance degradation at the end[11].

7.5.2. Image explorations

Exploration of the images for both improved and deteriorated cases reveal that certain characteristics of the image influence the efficiency of the recommendations. For the user with the highest performance improvement where NDCG is more than doubled (from 0.33 in the non-visual TMPG-BL model to 0.82 in the VL-PCA model), the color theme and shape of the product packaging improved the recommendations enormously. The images in Figure 11 belong to the items recommended to this user by the VL-PCA model that caused the highest performance (NDCG) jump on the test sets. The base model however did not recommend them to this user.

Figure 11. Image of the items with a positive impact on the recommendation for user 1628

[11] Data in RNDM and TMPG strategies are different which is already explained in page 44. Also, the RR model is a random model unrelated to MF which is addressed in the Models and Scenarios section.

As depicted in Figure 11, this user is attracted by the items with rectangular packaging shapes plus the brown and green color theme. Such specifications cover the user's taste and offering the items in Figure 11 to this user instead of the items in Figure 12 (which are recommended by the base model) improves the recommendations for this user considerably. Items with the mentioned characteristics can make the user perform more transactions, and such image characteristics have successfully been traced and captured by the VL_PCA model for making a better-personalized recommendation for this user.

Figure 12. Items recommended to user 1628 whose removal from the list of offers improved the performance

On the other hand, for several users with high NDCG deterioration occurred by the visual models (from as high as 0.9 in the base model down to around 0.4 in the VL-PCA model), the items shown in Figure 13 played a significant role in imposing negative impacts on the recommendations. It is observed that darker color theme, product type (mostly coffee), and lack of aesthetic factors in the packaging design are involved in such negative impacts.

Figure 13. Images with a negative impact on the recommendations

7.5.3. Improved Recommendations and average dollars spent

The visual VL-PCA model has improved the performance of the recommendations for 134 users which are listed in Appendix 6. Such improvements represent 3.32% of the users in the system. The highest improvement occurs for user 1628 where NDCG is improved by 147% (from 0.33 in the base model up to 0.82 in the VL-PCA model), and the lowest improvement is observed for user 3202 with only a 0.08% increase in the NDCG (from 0.2471 in the base model up to 0.2473 in the visual model).

Exploration of the output data shows that users with the improved recommendations purchase between 1 to 26 items with the average being 5 items purchased, and the average price of the items that improved the recommendations is $25. By assuming that the recommendation models can scale, it is estimated that implementing such a model for a business with one million users has the potential to increase the annual sales by 1M (users) × 3.32% (users with improved recommendations) × 5 (items) × \$25 (price) = \$4’150’000 in average, which is a considerable contribution.

7.5.4. Analysis of the sales rank of the recommended items

The sales rank of the items represents their sales position in the *grocery and gourmet food* category. In the data of the TMPG strategy (used for the TMPG-BL, VL-PCA, and VL-tSNE models), the highest sales rank belongs to item 2311 (asin = B0029XDZIK) where the rank is equal to 6, and the lowest sales rank belongs to item 755 (asin = B001FA1DGY) with the rank equal to 267,474. The median sales rank of the available items is 10,628. Exploration of the recommended items in the TMPG-BL, VL-PCA, and VL-tSNE models, on the other hand, shows that the items purchased by the users whom their recommendations were improved by the visual models hold the median sales rank of 18,090, with the highest sales rank being 6 (belonging to item 2311) and the lowest being 180,323 (belonging to item 2141 – asin B000HDKZZK).

A comparison of the ranks reveals that the users whom their recommendations were improved by the visual models tend to purchase the items with lower sales rank, which means less popular items are discovered by the visual models and offered to the right users that prefer to buy them. Such discovery improves the capacity of the system in matching the items that are on the lower edge of the sales, with the user preferences. Such discovery covers the goal of the long tail economy already explained in the Introduction section of this thesis (section 1).

7.5.5. Brand Analysis

There are 1683 different brands in total available in the input data. Bob's Red Mill, Green Mountain Coffee, Frontier, Celestial Seasoning, Kirkland Signature, YOGI, Nature Path, Simply Organic, and Quaker are among the top ten brands in terms of the number of unique items they have to offer in the market (purchased by the users). Analysis of the outputs shows that except for Bob's Red Mill, none of the top brands show up on the list of items that contributed to the performance improvements of the recommendations, i.e. on the list that was described earlier in section 7.5.3.

It is evident that popular brands do not play role in the model improvements, and instead, the users' visual interests seem to play a much stronger role. Bariani Olive Oil Company, Skippy, Nature's Path, Lindt, Hot Kid, Tinkyada, PG Tips, Bob's Red Mill, Stauffer's, Stevita Stevia, and Good Earth are the top ten brands recommended [by the visual model] to the users with the highest performance improvement, hence they have higher contributions to the improvements on the recommendations. They appear more frequently than other brands to boost performance.

A list of the top 20 brands with higher contributions in improving the models, along with the frequency of their appearances can be found in Table 9. An exhaustive list of the brands for both the improved recommendations and the items available in the market is available in Appendix 7 and Appendix 8.

Table 9. List of the top 20 brands with the highest contribution to the performance improvements

Row	Brand Name	Count
1	Bariani Olive Oil	47
2	Company	41
3	Skippy	29
4	Nature's Path	19
5	Lindt	17
6	Hot Kid	17
7	Tinkyada	15
8	PG Tips	14
9	Bob's Red Mill	9
10	Stauffer's	9

Row	Brand Name	Count
11	Stevita Stevia	9
12	Good Earth	8
13	Uncle Lee's Tea	8
14	Nature Valley	7
15	Haribo	7
16	Hormel	6
17	Planters	5
18	Coffee People	5
19	Mars	4
20	Libby's	4

7.6. Discussions

Figure 7 reveals that in Amazon's *grocery and gourmet food* data, new items emerge regularly over time and substitute the old products. Hence, training the models during the product lifecycle is crucial because the trained items should exist in the market once recommended and they need to still be a choice for users. In addition to the ratings, having content features (here are visual signals) as additional information about the items is essential. Content features play an important role in recommending new items since most of the older items (based on which the user decisions are modeled) disappear after some time and no longer exist in the catalog. Such a strategy adds flexibility to cover the cold-start problem which enhances the quality of the design.

One of the many challenges in this work is dealing with a dataset that holds a couple of limitations compared to the benchmark data. In the Movielens dataset, for instance, there are 106 (in the *100K* version) – 165 (in the *1M* version) items on average rated by each user [35] whereas in our project only 7 items on average have been rated by each user, which is $1/15 - 1/24$ of the benchmark. This leaves a lower number of ratings in the training set and leaves insufficient instances on the validation and test subsets consequently lowering the reported ranking performances.

The other challenge is the rating class balance and distributions. As shown in Figure 3 and Figure 5, rating classes are highly unbalanced and the distribution skews toward higher scores. In the *grocery and gourmet food* dataset, the majority of the reviews hold very high rating scores; 57.8% of them have the highest rating score ($r = 5$), and 21.6% of them have one score below the highest ($r = 4$). Such skew makes the top *N* recommended items of each user come out of very competitive rating estimations, which increases the sensitivity of the items recommended, and it can also deteriorate the performance of the models. Although the objectives of this project are met, it would have been ideal to work with a dataset with a more balanced rating.

Amazon's *grocery and gourmet food* data used in this study is the only publicly available explicit feedback data at the time of this essay where image features of most of the items are available. The data has 14 years of user-item ratings. However, the first few years hold only a few interactions, and in the rest of the years most items disappear within a short period, i.e. after roughly 1.5 years, and they no longer exist in the catalog regardless of whether the models recommend them or not. Thus, it is crucial that the model training, as well as the recommendation of the items, are made within a reasonable period so that the items still exist in the market if recommended. This would lead to better discovery and personalization and gives higher business outcome. Therefore, the TMPG strategy is the preferred data split approach that takes the lifecycle of the items into account for recommendations.

Since the RNDM and TMPG subsets are different, a comparison of the ranking performance makes sense only within the strategies not between them. Therefore, comparisons between Table 6 and Table 7 are prohibited. Data chunks in the former and latter strategies are totally different and consequently, the number of unique users and unique items, as well as the total number of rated items (per user and total) in the validation and test sets, are entirely different which makes them incomparable.

7.7. Future Work

As highlighted in the objectives section, in this work we assumed that user decision factors do not evolve over time, and they remain constant. Since new items emerge over time and replace older ones on the catalog consistently, additional setting on the model seems essential for better recommendations. We expect that adding a temporal component improves the recommendation accuracy in grocery and food products. It is therefore suggested that new models with temporal dynamics of the decision factors are developed and further experiments with time dependency of the user and item factors, involved in Eq. 9, are conducted.

The addition of the image bias term to the equation likely improves the model. As emphasized in section 6.2, in this work, the image bias term is omitted to reduce the number of parameters for faster convergence. However, we believe that adding it back to the equation increases the performance, so it is suggested that the model be revised by adding the bias term and the rating estimation model be extended to $\hat{r}_{ij} = {\theta_i}^T \theta_j + {p_i}^T q_j + b_i + c_j + \beta^T f_j + \mu$, and finally a new round of experiments to be carried out to see what level of improvement is reachable.

In this work, we modeled a recommender using explicit feedback. An alternative would be implicit feedback modeling conditional to having additional buying information on the data. In case there is access to further purchasing data, e.g. those which users have bought but did not leave any review or did not rate, it is likely that recommendation through implicit feedback outperforms the current models and leads to better personalization. It is therefore suggested that the recommendations are remodeled by developing a state-of-the-art model for implicit feedback.

It is likely that optimizing the relevance of the items and ranking performances in the objective function instead of using root mean squared error, improves the ranking accuracy and personalized recommendation. Cremonesi et. al. [38] demonstrated that an optimized RMSE would not necessarily translate into a perfect ranking performance, so it is strongly suggested that the MF of this work be re-modeled by replacing the objective function with the relevance of the ranking lists, i.e. optimizing the *NDCG@20* instead of *RMSE*. We believe that this approach improves the reported NDCG@20 numbers greatly.

It is also recommended to repeat the same experiments (or conduct a series of new experiments with a similar modeling approach) on the datasets where images are more influential on user decisions. Jewelry[12], movie, restaurant - online order (Uber eat for instance), etc. are a few examples where we believe that working with a similar objective can lead to better personalization when image features are incorporated. These categories are recommended conditional to availability of the image features.

[12] There is a *clothing-shoes-jewelry* category, available on the repository of recommender systems datasets (http://jmcauley.ucsd.edu/data/amazon/links.html) which have already been explored. However, the author of this report found that only 12% of the items in this category have image features available (88% are missing the image features). Also, statistics of the 5-core data of this category reveal that the median and average of the ratings per user are 6 and 7, respectively, which is inferior to the current [studied] *grocery and gourmet food* category. Therefore, the author concluded that the *clothing-shoes-jewelry* category in current shape is not suitable for further investigations.

8. Conclusion

In the context of recommender systems, variability of the user-item interactions is typically explained by user and item decision factors known as non-visual factors. In this project, we showed that the variation of the interactions (ratings) is explained slightly better by incorporating the visual representation of the items. In such a setting, personalized recommendations are more precise once the parameters are optimized. We demonstrated that by incorporating the visual signals, the variability of the ratings is better explained through non-visual plus visual factors rather than by non-visual factors only. Personalized ranking and recommendations are consequently improved.

In this work, we generate a list of the top 20 Amazon *grocery and gourmet food* items for each user, they are sorted by predicted preferences, and a list of personalized recommendations is then generated based on them. We finally evaluate different scenarios by *NCDG@20* to cover the objective of the project and validate the hypothesis.

Analysis of the market time shows that most of the items disappear within a short period (after roughly 1.5 years) and they no longer exist in the catalog. Therefore, it is crucial that modeling and recommendations are carried out within a reasonable period so that the items that still exist in the market get a higher chance to be recommended. This tactic leads to better discovery and personalization, and it also returns higher business outcomes. In this project, the concept of temporal global is adopted for data splitting, and such a strategy is applied to take the lifecycle of the items into account for recommending the items.

Experiment results show that the baseline and alternative (visual) models outperform random recommendations, which means that personalization through matrix factorization of this project is always superior to any random personalization, without any exception. Furthermore, the ranking performance of the models having visual signals incorporated is 54.8% and 21.2% higher than their baseline, which signifies that involving visual decision factors in the models can enhance personalization greatly and impacts user decisions positively for buying.

Also, popularity analysis of the recommended items shows that visual models make recommendations in favor of the unpopular items whereas non-visual model recommends fairly popular and very popular items. The results show that the visual models absorb different signal than the base model, so it is very useful for applications such as cold start.

One important challenge of this study is the unbalanced rating classes that make the rating estimations and ranking of the unseen items too competitive. Despite the objective being met, it would have been ideal to use a dataset with more balanced rating classes. Besides, it is suggested that for similar objectives, datasets of more relevant domains like jewelry, restaurants (Uber eat for example), or movies where images have a stronger role in convincing the users to make a transaction, are used to repeat the current experiments or to conduct new ones. Because images of jewelry or movie posters have a higher impact on user decisions than grocery images, we believe that such data can be a better input for increasing the ranking performances if their image features are incorporated into the models.

Bibliography

[1] Falk, K. Practical Recommender Systems; Manning Publications: Shelter Island, NY, 2019.

[2] Ansari, A. Internet Recommendation Systems; *SAGE Publications Inc*, 2000; 37(3): 363-375. doi: 10.1509/jmkr.37.3.363.18779.

[3] Pazzani, M. J. and D. Billsus. 2007. “Content-Based Recommendation Systems.” Lecture Notes in Computer Science 4321(4321):325–41.

[4] Amatriain, Xavier. 2013. “Mining Large Streams of User Data for Personalized Recommendations.” Acm Sigkdd Explorations Newsletter 14(2):37–48.

[5] Bartolini, Ilaria, Zhenjie Zhang, and Dimitris Papadias. 2011. “Collaborative Filtering with Personalized Skylines.” Ieee Transactions on Knowledge & Data Engineering 23(2):190–203.

[6] Núñez-Valdéz Edward Rolando et al. 2012. “Implicit Feedback Techniques on Recommender Systems Applied to Electronic Books.” Computers in Human Behavior 28(4):1186–93.

[7] Pasi G and 6th International Conference on Intelligent Human Computer Interaction, IHCI 2014 6 2014 12 08 - 2014 12 10. 2014. “Implicit Feedback through User-System Interactions for Defining User Models in Personalized Search.” Procedia Computer Science 39:8–11.

[8] McAuley, Julian. 2022. Personalized Machine Learning. Cambridge, United Kingdom: Cambridge University Press. Retrieved August 1, 2022 (INSERT-MISSING-URL).

[9] Koren, Y., R. Bell, and C. Volinsky. 2009. “Matrix Factorization Techniques for Recommender Systems.” Computer -Ieee Computer Society- 42(8):30–37.

[10] Xue H.-J et al. 2017. “Deep Matrix Factorization Models for Recommender Systems.” Ijcai International Joint Conference on Artificial Intelligence 3203-3209:3203–9.

[11] Zhong S.-T et al. 2020. “A Model-Bias Matrix Factorization Approach for Course Score Prediction.” Neural Processing Letters (2020).

[12] Website: activisiongamescience.github.io/2016/01/11/Implicit-Recommender-Systems-Biased-Matrix-Factorization/

[13] Zhang Sh. (2019). *Personalized Recommendation: Neural Architecture and Beyond*. University of New South Wales, Australia.

[14] Paterek, A. (2007, August). Improving regularized singular value decomposition for collaborative filtering. In Proceedings of KDD cup and workshop (Vol. 2007, pp. 5-8)

[15] Koren Y. 2010. “Collaborative Filtering with Temporal Dynamics.” Communications of the Acm 53(4):89–97.

[16] Isinkaye, F. O., Y. O. Folajimi, and B. A. Ojokoh. 2015. “Recommendation Systems: Principles, Methods and Evaluation.” Egyptian Informatics Journal 16(3):261–73.

[17] Hu Y, Koren Y, Volinsky C, and 8th IEEE International Conference on Data Mining, ICDM 2008 Pisa, ITA 2008 12 15 - 2008 12 19. 2008. “Collaborative Filtering for Implicit Feedback Datasets.” Proceedings - Ieee International Conference on Data Mining, Icdm 263-272:263–72.

[18] Bell, Robert M., Yehuda Koren, and Seventh IEEE International Conference on Data Mining (ICDM 2007) Omaha, NE 28 October 2007 - 31 October 2007. 2007. “Seventh Ieee International Conference on Data Mining (Icdm 2007).” Pp. 43–52 in Scalable Collaborative Filtering with Jointly Derived Neighborhood Interpolation Weights.

[19] Packer C, McAuley J, and Ramisa A. 2018. “Visually-Aware Personalized Recommendation Using Interpretable Image Representations.” Arxiv (2018 06 26).

[20] He, Ruining and Julian McAuley. 2016. “Proceedings of the 25th International Conference on World Wide Web.” Pp. 507–17 in Ups and Downs: Modeling the Visual Evolution of Fashion Trends with One-Class Collaborative Filtering.

[21] He, Ruining and Julian McAuley. 2016. “Vbpr: Visual Bayesian Personalized Ranking from Implicit Feedback.” Proceedings of the Aaai Conference on Artificial Intelligence 30(1).

[22] Messina P et al. 2020. “Curatornet: Visually-Aware Recommendation of Art Images.” Ceur Workshop Proceedings 2697.

[23] Bluche T, Ney H, Kermorvant C, and 12th International Conference on Document Analysis and Recognition, ICDAR 2013 Washington, DC, USA 2013 08 25 - 2013 08 28. 2013. “Feature Extraction with Convolutional Neural Networks for Handwritten Word Recognition.” Proceedings of the International Conference on Document Analysis and Recognition, Icdar 285-289:285–89.

[24] Naranjo-Torres José et al. 2020. “A Review of Convolutional Neural Network Applied to Fruit Image Processing.” Applied Sciences 10(10):3443–43.

[25] Said, Alan and Bellogín Alejandro. 2014. “Proceedings of the 8th Acm Conference on Recommender Systems.” Pp. 129–36 in Comparative Recommender System Evaluation: Benchmarking Recommendation Frameworks.

[26] Meng Z, McCreadie R, Macdonald C, and Ounis I. 2020. “Exploring Data Splitting Strategies for the Evaluation of Recommendation Models.” Arxiv (2020 07 26).

[27] Dacrema M.F, Cremonesi P, and Jannach D. 2019. “Are We Really Making Much Progress? A Worrying Analysis of Recent Neural Recommendation Approaches.” Arxiv (2019 07 16).

[28] Rendle, S., Freudenthaler, C., Gantner, Z., & Schmidt-Thieme, L. (2012). BPR: Bayesian personalized ranking from implicit feedback. arXiv preprint arXiv:1205.2618.

[29] Quadrana, Massimo, Paolo Cremonesi, and Dietmar Jannach. 2019. “Sequence-Aware Recommender Systems.” Acm Computing Surveys 51(4):1–36.

[30] Zamani, Hamed and W. Bruce Croft. 2020. “Proceedings of the 13th International Conference on Web Search and Data Mining.” Pp. 717–25 in Learning a Joint Search and Recommendation Model from User-Item Interactions.

[31] Rendle, Steffen, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. “Proceedings of the 19th International Conference on World Wide Web.” Pp. 811–20 in Factorizing Personalized Markov Chains for Next-Basket Recommendation.

[32] Liang D, Jebara T, Hoffman M.D, and Krishnan R.G. 2018. “Variational Autoencoders for Collaborative Filtering.” Arxiv (2018 02 15).

[33] Meng Z, McCreadie R, Macdonald C, and Ounis I. 2019. “Variational Bayesian Context-Aware Representation for Grocery Recommendation.” Arxiv (2019 09 17).

[34] Zhang S et al. 2020. “Unraveling Metric Vector Spaces with Factorization for Recommendation.” Ieee Transactions on Industrial Informatics 16(2):732–42.

[35] Zhang, Richong, Han Bao, Hailong Sun, Yanghao Wang, and Xudong Liu. 2016. “Recommender Systems Based on Ranking Performance Optimization.” Frontiers of Computer Science 10(2):270–80.

[36] Sun, Zhongchuan, Bin Wu, Yunpeng Wu, and Yangdong Ye. 2019. “Apl: Adversarial Pairwise Learning for Recommender Systems.” Expert Systems with Applications 118:573–84.

[37] Balakrishnan, Suhrid and Sumit Chopra. 2012. “Proceedings of the Fifth Acm International Conference on Web Search and Data Mining.” Pp. 143–52 in Collaborative Ranking.

[38] Cremonesi, Paolo, Yehuda Koren, and Roberto Turrin. 2010. “Proceedings of the Fourth Acm Conference on Recommender Systems.” Pp. 39–46 in Performance of Recommender Algorithms on Top-N Recommendation Tasks.

[39] McAuley, Julian, Christopher Targett, Qinfeng Shi, and Anton van den Hengel. 2015. “Proceedings of the 38th International Acm Sigir Conference on Research and Development in Information Retrieval.” Pp. 43–52 in Image-Based Recommendations on Styles and Substitutes.

[40] He, Ruining and Julian McAuley. 2016. “Proceedings of the 25th International Conference on World Wide Web.” Pp. 507–17 in Ups and Downs: Modeling the Visual Evolution of Fashion Trends with One-Class Collaborative Filtering.

[41] Marlin, Benjamin M. and Richard S. Zemel. 2009. “Proceedings of the Third Acm Conference on Recommender Systems.” Pp. 5–12 in Collaborative Prediction and Ranking with Non-Random Missing Data.

[42] Wimble, Matt, John Tripp, Brandis Phillips, and Nash Milic. 2016. “On Search Cost and the Long Tail: The Moderating Role of Search Cost.” Information Systems and E-Business Management 14(3):507–31.

[43] Anderson, Chris. 2006. The Long Tail: Why the Future of Business Is Selling Less of More. New York: Hyperion.

[44] B. Gawri, A. Kasturi, L. B. M. Neti and C. Hota. 2022. “14th International Conference on COMmunication Systems & NETworkS (COMSNETS)” Pp. 734-738 in An efficient approach to kNN algorithm for IoT devices doi: 10.1109/COMSNETS53615.2022.9668540.

[45] Pero, S., Horvath, T., & 21st International Conference on User Modeling, Adaptation and Personalization, UMAP 2013 Rome, ITA 2013 06 10 - 2013 06 14. (2013). Opinion-driven matrix factorization for rating prediction. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 7899 Lncs, 1–13. https://doi.org/10.1007/978-3-642-38844-6_1

Appendix

Appendix 1. Details of the Equations for Computing User and Item Factors through ALS Optimization Technique

$$\mathcal{L} = min\left(\sum_{i,j\in S}\left(r_{ij} - \hat{r}_{ij}\right)^2 + \lambda\left(\|p_i\|_F^2 + \left\|q_j\right\|_F^2 + {b_i}^2 + {c_j}^2\right)\right)$$

To optimize the system:

$$\frac{\partial\mathcal{L}}{\partial p_i} = 0 \rightarrow solve\ for\ p_i \qquad (1)$$

$$\frac{\partial\mathcal{L}}{\partial q_j} = 0 \rightarrow solve\ for\ q_j \qquad (2)$$

$$\frac{\partial\mathcal{L}}{\partial b_i} = 0 \rightarrow solve\ for\ b_i \qquad (3)$$

$$\frac{\partial\mathcal{L}}{\partial c_j} = 0 \rightarrow solve\ for\ c_j \qquad (4)$$

$$(1) \rightarrow 2\sum_{j\in W_i}\left(r_{ij} - p_i^T q_j - b_i - c_j - \mu\right)\left(-q_j\right) + 2\lambda p_i = 0$$

$$\sum_{j\in W_i} q_j q_j^T p_i + \lambda p_i = \sum_{j\in W_i}\left(r_{ij} - b_i - c_j - \mu\right)\left(q_j\right)$$

$$\left(\sum_{j\in W_i} q_j q_j^T + \lambda I\right)(p_i) = \sum_{j\in W_i}\left(r_{ij} - b_i - c_j - \mu\right)\left(q_j\right)$$

$$p_i = \left(\sum_{j\in W_i} q_j q_j^T + \lambda I\right)^{-1}\left(\sum_{j\in W_i}\left(r_{ij} - b_i - c_j - \mu\right)\left(q_j\right)\right)$$

$where$: $\quad W_i = \text{set of items rated by user i}$

$$(2) \rightarrow 2\sum_{i\in U_j}\left(r_{ij} - p_i^T q_j - b_i - c_j - \mu\right)(-p_i) + 2\lambda q_j = 0$$

$$\sum_{i\in U_j} p_i p_i^T q_j + \lambda q_j = \sum_{i\in U_j}\left(r_{ij} - b_i - c_j - \mu\right)(p_i)$$

$$\left(\sum_{i\in U_j} p_i p_i^T + \lambda I\right)\left(q_j\right) = \sum_{i\in U_j}\left(r_{ij} - b_i - c_j - \mu\right)(p_i)$$

$$q_j = \left(\sum_{i\in U_j} p_i p_i^T + \lambda I\right)^{-1}\left(\sum_{i\in U_j}\left(r_{ij} - b_i - c_j - \mu\right)(p_i)\right)$$

$where$: $U_j =$ set of users rated item j

$$(3) \rightarrow 2\sum_{j\in W_i}\left(r_{ij} - p_i^T q_j - b_i - c_j - \mu\right)(-1) + 2\lambda b_i = 0$$

$$\sum_{j\in W_i} b_i + \lambda b_i = \sum_{j\in W_i}\left(r_{ij} - p_i^T q_j - c_j - \mu\right)$$

$$\sum_{j\in W_i} 1 \times b_i + \lambda b_i = \sum_{j\in W_i}\left(r_{ij} - p_i^T q_j - c_j - \mu\right)$$

$$b_i \sum_{j\in W_i} 1 + \lambda = \sum_{j\in W_i}\left(r_{ij} - p_i^T q_j - c_j - \mu\right)$$

$$b_i = \left(\frac{1}{|\mathrm{W}| + \lambda}\right)\left(\sum_{j\in W_i}\left(r_{ij} - p_i^T q_j - c_j - \mu\right)\right)$$

$where$: $W_i =$ set of items rated by user i

$|W| = size\ of\ W = number\ of\ items\ in\ the\ set$ W

$$(4) \rightarrow 2 \sum_{i \in U_j} \left(r_{ij} - p_i^T q_j - b_i - c_j - \mu \right)(-1) + 2\lambda c_j = 0$$

$$\sum_{i \in U_j} c_j + \lambda c_j = \sum_{i \in U_j} \left(r_{ij} - p_i^T q_j - b_i - \mu \right)$$

$$\sum_{i \in U_j} 1 \times c_j + \lambda c_j = \sum_{i \in U_j} \left(r_{ij} - p_i^T q_j - b_i - \mu \right)$$

$$c_j \sum_{i \in U_j} 1 + \lambda = \sum_{i \in U_j} \left(r_{ij} - p_i^T q_j - b_i - \mu \right)$$

$$c_j = \left(\frac{1}{|\mathrm{U}| + \lambda} \right) \left(\sum_{i \in U_j} \left(r_{ij} - p_i^T q_j - b_i - \mu \right) \right)$$

$where$: $U_j =$ set of users rated item j

$|\mathrm{U}| = size\ of\ \mathrm{U} = number\ of\ users\ in\ the\ set\ \mathrm{U}$

Appendix 2. Details of the Equations for Computing User and Item Visual and Non-visual Dimensions through ALS Optimization Technique in a Regularized System

$$\mathcal{L} = min\left(\sum_{i,j\in S}\left(r_{ij} - \hat{r}_{ij}\right)^2 + \lambda\left(\|\theta_i\|_F^2 + \left\|\theta_j\right\|_F^2 + \|p_i\|_F^2 + \left\|q_j\right\|_F^2 + b_i{}^2 + c_j{}^2\right)\right)$$

$$\hat{r}_{ij} = \theta_i.\theta_j{}^T + p_i.q_j{}^T + b_i + c_j + \mu$$

$$\theta_j = E_j f_j$$

To optimize the regularized system:

$$\frac{\partial\mathcal{L}}{\partial\theta_i} = 0 \rightarrow solve\ for\ \theta_i \qquad (5)$$

$$\frac{\partial\mathcal{L}}{\partial\theta_j} = 0 \rightarrow solve\ for\ \theta_j \qquad (6)$$

$$\frac{\partial\mathcal{L}}{\partial p_i} = 0 \rightarrow solve\ for\ p_i \qquad (3)$$

$$\frac{\partial\mathcal{L}}{\partial q_j} = 0 \rightarrow solve\ for\ q_j \qquad (4)$$

$$\frac{\partial\mathcal{L}}{\partial b_i} = 0 \rightarrow solve\ for\ b_i \qquad (5)$$

$$\frac{\partial\mathcal{L}}{\partial c_j} = 0 \rightarrow solve\ for\ c_j \qquad (6)$$

$$(1) \rightarrow 2 \sum_{j \in W_i} \left(r_{ij} - \theta_i^T \theta_j - p_i^T q_j - b_i - c_j - \mu\right)\left(-\theta_j\right) + 2\lambda\theta_i = 0$$

$$\sum_{j \in W_i} \theta_j \theta_j^T \theta_i + \lambda\theta_i = \sum_{j \in W_i} \left(r_{ij} - p_i^T q_j - b_i - c_j - \mu\right)\left(\theta_j\right)$$

$$\left(\sum_{j \in W_i} \theta_j \theta_j^T + \lambda I\right)(\theta_i) = \sum_{j \in W_i} \left(r_{ij} - p_i^T q_j - b_i - c_j - \mu\right)\left(\theta_j\right)$$

$$\theta_i = \left(\sum_{j \in W_i} \theta_j \theta_j^T + \lambda I\right)^{-1} \left(\sum_{j \in W_i} \left(r_{ij} - p_i^T q_j - b_i - c_j - \mu\right)\left(\theta_j\right)\right)$$

where: W_i = set of items rated by user i

$$(2) \rightarrow 2 \sum_{i \in U_j} \left(r_{ij} - \theta_i^T \theta_j - p_i^T q_j - b_i - c_j - \mu\right)(-\theta_i) + 2\lambda\theta_j = 0$$

$$\sum_{i \in U_j} \theta_i \theta_i^T \theta_j + \lambda\theta_j = \sum_{i \in U_j} \left(r_{ij} - p_i^T q_j - b_i - c_j - \mu\right)(\theta_i)$$

$$\left(\sum_{i \in U_j} \theta_i \theta_i^T + \lambda I\right)\left(\theta_j\right) = \sum_{i \in U_j} \left(r_{ij} - p_i^T q_j - b_i - c_j - \mu\right)(\theta_i)$$

$$\theta_j = E_j f_j$$

$$E_j = \left(f_j \sum_{i \in U_j} \theta_i \theta_i^T + \lambda I\right)^{-1} \left(\sum_{i \in U_j} \left(r_{ij} - p_i^T q_j - b_i - c_j - \mu\right)(\theta_i)\right)$$

where: U_j = set of users rated item j

$$(3) \rightarrow 2\sum_{j\in W_i}\left(r_{ij}-\theta_i^T\theta_j-p_i^Tq_j-b_i-c_j-\mu\right)\left(-q_j\right)+2\lambda p_i=0$$

$$\sum_{j\in W_i} q_jq_j^Tp_i+\lambda p_i=\sum_{j\in W_i}\left(r_{ij}-\theta_i^T\theta_j-b_i-c_j-\mu\right)\left(q_j\right)$$

$$\left(\sum_{j\in W_i} q_jq_j^T+\lambda I\right)(p_i)=\sum_{j\in W_i}\left(r_{ij}-\theta_i^T\theta_j-b_i-c_j-\mu\right)\left(q_j\right)$$

$$\theta_j=E_jf_j$$

$$p_i=\left(\sum_{j\in W_i} q_jq_j^T+\lambda I\right)^{-1}\left(\sum_{j\in W_i}\left(r_{ij}-\theta_i^TE_jf_j-b_i-c_j-\mu\right)\left(q_j\right)\right)$$

$where$: $W_i =$ set of items rated by user i

$$(4) \rightarrow 2\sum_{i\in U_j}\left(r_{ij}-\theta_i^T\theta_j-p_i^Tq_j-b_i-c_j-\mu\right)(-p_i)+2\lambda q_j=0$$

$$\sum_{i\in U_j} p_ip_i^Tq_j+\lambda q_j=\sum_{i\in U_j}\left(r_{ij}-\theta_i^T\theta_j-b_i-c_j-\mu\right)(p_i)$$

$$\left(\sum_{i\in U_j} p_ip_i^T+\lambda I\right)\left(q_j\right)=\sum_{i\in U_j}\left(r_{ij}-\theta_i^T\theta_j-b_i-c_j-\mu\right)(p_i)$$

$$\theta_j=E_jf_j$$

$$q_j=\left(\sum_{i\in U_j} p_ip_i^T+\lambda I\right)^{-1}\left(\sum_{i\in U_j}\left(r_{ij}-\theta_i^TE_jf_j-b_i-c_j-\mu\right)(p_i)\right)$$

$where$: $U_j =$ set of users rated item j

$$(5) \rightarrow 2 \sum_{j \in W_i} \left(r_{ij} - \theta_i^T \theta_j - p_i^T q_j - b_i - c_j - \mu \right) (-1) + 2\lambda b_i = 0$$

$$\sum_{j \in W_i} b_i + \lambda b_i = \sum_{j \in W_i} \left(r_{ij} - \theta_i^T \theta_j - p_i^T q_j - c_j - \mu \right)$$

$$\sum_{j \in W_i} 1 \times b_i + \lambda b_i = \sum_{j \in W_i} \left(r_{ij} - \theta_i^T \theta_j - p_i^T q_j - c_j - \mu \right)$$

$$b_i \sum_{j \in W_i} 1 + \lambda = \sum_{j \in W_i} \left(r_{ij} - \theta_i^T \theta_j - p_i^T q_j - c_j - \mu \right)$$

$$\theta_j = E_j f_j$$

$$b_i = \left(\frac{1}{|\mathrm{W}| + \lambda} \right) \left(\sum_{j \in W_i} \left(r_{ij} - \theta_i^T E_j f_j - p_i^T q_j - c_j - \mu \right) \right)$$

$where$: W_i = set of items rated by user i

$|W| = size\ of\ W = number\ of\ items\ in\ the\ set$ W

$$(6) \rightarrow 2 \sum_{i \in U_j} \left(r_{ij} - \theta_i^T \theta_j - p_i^T q_j - b_i - c_j - \mu \right) (-1) + 2\lambda c_j = 0$$

$$\sum_{i \in U_j} c_j + \lambda c_j = \sum_{i \in U_j} \left(r_{ij} - \theta_i^T \theta_j - p_i^T q_j - b_i - \mu \right)$$

$$\sum_{i \in U_j} 1 \times c_j + \lambda c_j = \sum_{i \in U_j} \left(r_{ij} - \theta_i^T \theta_j - p_i^T q_j - b_i - \mu \right)$$

$$c_j \sum_{i \in U_j} 1 + \lambda = \sum_{i \in U_j} \left(r_{ij} - \theta_i^T \theta_j - p_i^T q_j - b_i - \mu \right)$$

$$\theta_j = E_j f_j$$

$$c_j = \left(\frac{1}{|\mathrm{U}| + \lambda}\right)\left(\sum_{i \in U_j}\left(r_{ij} - \theta_i^T E_j f_j - p_i^T q_j - b_i - \mu\right)\right)$$

$where$: U_j = set of users rated item j

$|\mathrm{U}| = size\ of\ \mathrm{U} = number\ of\ users\ in\ the\ set\ \mathrm{U}$

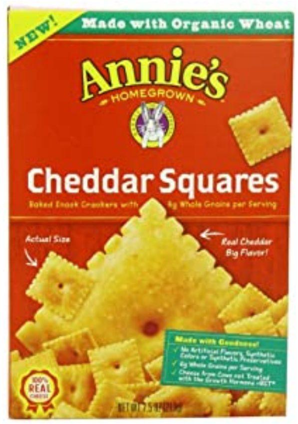
Appendix 3. Few Examples of the Product Review Raw Data

	reviewerID	asin	reviewerName	helpful	reviewText	overall	summary	unixReviewTime	reviewTime
0	A1VEELTKS8NLZB	616719923X	Amazon Customer	[0, 0]	Just another flavor of Kit Kat but the taste i...	4.0	Good Taste	1370044800	06 1, 2013
1	A14R9XMZVJ6INB	616719923X	amf0001	[0, 1]	I bought this on impulse and it comes from Jap...	3.0	3.5 stars, sadly not as wonderful as I had hoped	1400457600	05 19, 2014
2	A27IQHDZFQFNGG	616719923X	Caitlin	[3, 4]	Really good. Great gift for any fan of green t...	4.0	Yum!	1381190400	10 8, 2013
3	A31QY5TASILE89	616719923X	DebraDownSth	[0, 0]	I had never had it before, was curious to see ...	5.0	Unexpected flavor meld	1369008000	05 20, 2013
4	A2LWK003FFMCI5	616719923X	Diana X.	[1, 2]	I've been looking forward to trying these afte...	4.0	Not a very strong tea flavor, but still yummy ...	1369526400	05 26, 2013
5	A1NZJTY0BAA2SK	616719923X	Elizabeth	[0, 1]	These Kit-kats are very good, but if you're lo...	4.0	Subtle	1378339200	09 5, 2013
6	AA95FYFIP38RM	616719923X	Emily Veinglory "Book Reviewer"	[1, 2]	I found these in a Mitsuwa Marketplace in Illi...	3.0	Available in some US stores	1382054400	10 18, 2013
7	A3FIVHUOGMUMPK	616719923X	greenlife	[2, 3]	Creamy white chocolate infused with Matcha gre...	5.0	So Delicious!!	1372982400	07 5, 2013
8	A27FSPAMTQF1J8	616719923X	Japhyl	[0, 0]	After hearing mixed opinions about these Kit K...	5.0	These are my favorite candies ever!	1371168000	06 14, 2013
9	A33NXNZ79H5K51	616719923X	Jean M "JM"	[0, 10]	I love green tea, I love Kit Kats, but the two...	1.0	Not a fan	1348012800	09 19, 2012
10	A220GN2X2R47JE	616719923X	Jeremy	[6, 8]	I ordered these in Summer so they of course ar...	5.0	Amazing!	1380672000	10 2, 2013
11	A3C5Z05IKSSFB9	616719923X	M. Magpoc "maliasuperstar"	[2, 3]	These are definitely THE BEST candy bar out th...	5.0	I wish I could find these in a store instead o...	1369526400	05 26, 2013
12	AHA6G4IMEMAJR	616719923X	M. Zinn "mczinn"	[0, 0]	Yes - this is one of the most expensive candie...	5.0	Thank goodness they are expensive	1373068800	07 6, 2013
13	A1Q2E3W9PRG313	616719923X	Sabrina	[0, 0]	I love the green tea kitkat, taste so good, no...	5.0	it is good	1370649600	06 8, 2013
14	A1P3PLYYMURAV1	616719923X	Sunny	[0, 0]	I love Kit Kat & green tea....together they ar...	3.0	Meh	1380153600	09 26, 2013
15	A38IEZF0P3ZUQJ	616719923X	The Fallen Angel with a broken Wing "I Cry Ou...	[0, 0]	I tried this for the first time today and it i...	4.0	Not enough Matcha tea flavor for me ... (=_=)	1374192000	07 19, 2013
16	A23RYWDS884TUL	9742356831	Another Freak	[0, 0]	This curry paste makes a delicious curry. I j...	5.0	Delicious!	1369699200	05 28, 2013
17	A945RBQWGZXCK	9742356831	Cheryl	[1, 2]	I've purchased different curries in the grocer...	5.0	Great flavor	1347840000	09 17, 2012
18	A1TCSC0YWT82Q0	9742356831	GinSing	[2, 2]	I love ethnic foods and to cook them. I recent...	5.0	OMG! What a treasure find!	1375488000	08 3, 2013
19	A3AMNY44OP8AOU	9742356831	Jennifer Lee	[1, 1]	I started a new diet restricting all added sug...	4.0	Tastes great!	1390435200	01 23, 2014
20	A3IB4CQ2QEJLJ8	9742356831	J. Zack	[0, 0]	So many flavors. I can't begin to tell you how...	5.0	Thai curry is the way to go to add spice to an...	1398556800	04 27, 2014
21	AQA5DF3RWKETQ	9742356831	LindaE	[1, 2]	I've used this a lot recently in some of my ch...	5.0	Thai Green Curry	1353974400	11 27, 2012
22	AOHQ17LGZHTI5	9742356831	Matthew S.	[1, 2]	This is a huge step up in quality from the kin...	5.0	Make green curry quite easy	1359072000	01 25, 2013
23	AZKRFNQ8EFO4T	9742356831	Mike Williams	[4, 4]	I have two things in mind when I make a Thai c...	5.0	Authentic, easy, and delicious!	1383609600	11 5, 2013
24	A1Z7Y2GMAP9SRY	9742356831	M. Thompson "Dyson Diva"	[0, 0]	I like to make my own curry but this is a tast...	5.0	Yum!	1403827200	06 27, 2014

Appendix 4. Few Examples of the Product Metadata

	asin	description	title	imUrl	related	salesRank	categories	price	brand
0	0657745316	This is real vanilla extract made with only 3 ...	100 Percent All Natural Vanilla Extract	http://ecx.images-amazon.com/images/I/41gFi5h0...	{'also_viewed': ['B001GE8N4Y']}	{'Grocery & Gourmet Food': 374004}	[[Grocery & Gourmet Food]]	NaN	NaN
1	0700026444	Silverpot Tea, Pure Darjeeling, is an exquisit...	Pure Darjeeling Tea: Loose Leaf	http://ecx.images-amazon.com/images/I/51hs8sox...	NaN	{'Grocery & Gourmet Food': 620307}	[[Grocery & Gourmet Food]]	NaN	NaN
2	1403796890	Must have for any WWE Fan\n \n \n \nFeaturing ...	WWE Kids Todler Velvet Slippers featuring John...	http://ecx.images-amazon.com/images/I/518SEST5...	NaN	NaN	[[Grocery & Gourmet Food]]	3.99	NaN
3	141278509X	Infused with Vitamins and Electrolytes Good So...	Archer Farms Strawberry Dragonfruit Drink Mix ...	http://ecx.images-amazon.com/images/I/51CFQlis...	{'also_viewed': ['B0051IETTY']}	{'Grocery & Gourmet Food': 620322}	[[Grocery & Gourmet Food]]	NaN	NaN
4	1453060375	MiO Energy is your portable energy source givi...	Mio Energy Liquid Water Enhancer Black Cherry ...	http://ecx.images-amazon.com/images/I/51EUsMcn...	{'also_viewed': ['B006MSEOJ2', 'B005VOOQLO', '...	{'Grocery & Gourmet Food': 268754}	[[Grocery & Gourmet Food]]	11.99	Mio
5	1453060782	With these Splash water flavor enhancers you a...	Splash Energy Liquid Water Enhancer 24 Serving...	http://ecx.images-amazon.com/images/I/51-y9QTW...	{'also_viewed': ['B00C1LXBFC', 'B006MSEOJ2', '...	NaN	[[Grocery & Gourmet Food]]	6.39	NaN
6	1453060464	With these Splash water flavor enhancers you a...	Splash Energy Liquid Water Enhancer 24 Serving...	http://ecx.images-amazon.com/images/I/51-y9QTW...	{'also_viewed': ['B00C1LXBFC']}	{'Grocery & Gourmet Food': 221057}	[[Grocery & Gourmet Food]]	NaN	NaN
7	1603112251	Become a cocktail king with these unique bitte...	Cocktail Kingdom Wormwood Bitters - 5 oz	http://ecx.images-amazon.com/images/I/41wokwTg...	{'also_viewed': ['B001CDTO6U', '1580083595', '...	{'Grocery & Gourmet Food': 43972}	[[Grocery & Gourmet Food]]	17.95	NaN
8	1613170416	Twelve dice inspired by the Dresden Files nove...	Evil Hat Productions Fate Dice: Winter Knight	http://ecx.images-amazon.com/images/I/41Worg0y...	{'also_bought': ['0977153479', '0977153487', '...	{'Toys & Games': 26935}	[[Grocery & Gourmet Food, Beverages, Coffee, C...	14.05	Evil Hat Productions
9	1837994021	A glass and a half full of joy! Imagine little...	Cadbury Dairy Milk Daim 120g	http://ecx.images-amazon.com/images/I/41a47ZTu...	{'also_viewed': ['B000JZHDWE', 'B004G9204K', '...	{'Grocery & Gourmet Food': 457148}	[[Grocery & Gourmet Food]]	11.95	Cadbury
10	3301261876	A tub of 600 Haribo fruit flavoured Jelly Babies	Haribo Jelly Babies Gummy Sweets	http://ecx.images-amazon.com/images/I/51Hh-%2B...	{'also_bought': ['B005E915PG', 'B000KCXKOQ', '...	{'Grocery & Gourmet Food': 107168}	[[Grocery & Gourmet Food]]	25.00	NaN
11	3295000018	Kiva Gourmet Ghost Chili Pepper Powder:Made fr...	Kiva Gourmet Smoked, Ghost Chili Pepper Powder...	http://ecx.images-amazon.com/images/I/41LsMz6S...	{'also_bought': ['B004C9PTCE', 'B001PQTYN2', '...	{'Grocery & Gourmet Food': 35703}	[[Grocery & Gourmet Food]]	14.80	NaN
12	3621813330	The larger size of the Starburst Sour Chews ba...	Starburst Sour Chews 240g Bag	http://ecx.images-amazon.com/images/I/41ou31A-...	{'also_viewed': ['B00BHFB3G2', 'B008RYX8T6', '...	{'Grocery & Gourmet Food': 235044}	[[Grocery & Gourmet Food]]	12.50	NaN
13	5628754218	Do not look for cheap if you are looking the o...	Iaso Tea Original By Total Life Changes and Dr...	http://ecx.images-amazon.com/images/I/61KIZFA3...	{'also_viewed': ['B00BD4CLYU', 'B005DN3DC6', '...	{'Grocery & Gourmet Food': 27229}	[[Grocery & Gourmet Food]]	NaN	NaN
14	5901002482	NaN	Scripture Tea Apricot Decaf Box of 20	http://ecx.images-amazon.com/images/I/31xEqgGv...	{'also_viewed': ['B0031MK6NK', 'B004J0HRWG', '...	{'Grocery & Gourmet Food': 216840}	[[Grocery & Gourmet Food]]	NaN	NaN
15	6065555568	Overview\nIncrease energy and improve mental c...	HERBALIFE ENERGY SUPPLEMENT LIFTOFF	http://ecx.images-amazon.com/images/I/41qRmdMR...	{'also_viewed': ['B000JQFZJG', 'B000A39454', '...	{'Health & Personal Care': 509663}	[[Grocery & Gourmet Food, Beverages, Energy Dr...	59.95	NaN

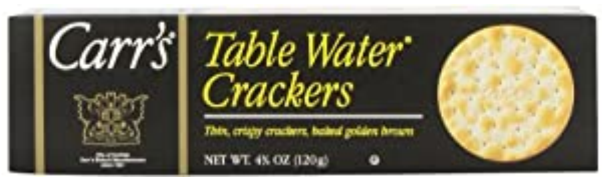
Appendix 5. Few Examples of the Product Images




QUAKER
Chewy
Variety Pack
30 Chocolate Chip
60
Granola Bars

Worthington
Loma Linda
VEGAN
Tender Bits
Made with Grain Protein

Kellogg's
Cracklin'
Oat Bran

Low Calorie
Tofu
Shirataki
NOODLE SHAPED TOFU
NET WT. 8 oz. (226g)

Carr's
Table Water
Crackers

Keebler
Vanilla
Wafers
Baked with Care!
Be a GOOD COOKIE
NET WT 12 OZ (340g)

KIRKLAND
Rice
Crackers
with Nuts

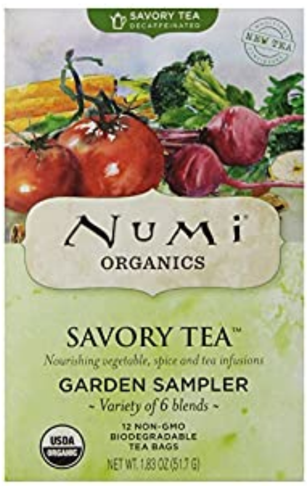

Philadelphia Candies
MILK CHOCOLATE PECANETTES

Kellogg's
Nutri-Grain
fruit
crunch
NEW
MADE WITH REAL FRUIT
apple cobbler
10 granola bars


Hoosier Hill Farm
gourmet White
Cheddar Cheese
powder


TAHINI
Pure Ground Sesame


A Taste Of Italy
MILANO'S
100% Grated
IMPORTED
PARMESAN
Cheese
Aged Over 10 Months
Gluten Free
NET WT 8 OZ (227g)

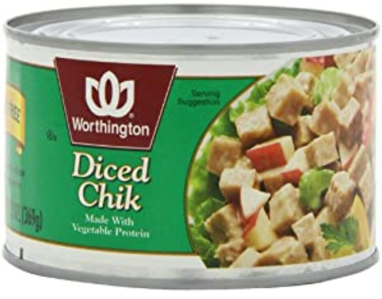

SAVORY TEA
NUMI
ORGANICS
SAVORY TEA™
Nourishing vegetable, spice and tea infusions
GARDEN SAMPLER
~ Variety of 6 blends ~
12 NON-GMO
BIODEGRADABLE
TEA BAGS
NET WT. 1.83 OZ (51.7 G)


Lipton
green tea
made with real fruit pieces
Mandarin Orange
FLAVOR WITH OTHER NATURAL FLAVOR
20 PYRAMID TEA BAGS
NET WT 1.2 OZ (34g)


CELESTIAL
SEASONINGS
Free
Red Dress
Pin
Perfectly Pear
WHITE TEA
CONTAINS CAFFEINE • 100% NATURAL
20 TEA BAGS • NET WT 1.0 OZ (29g)


GLUTEN FREE
NON-GMO
KOSHER
LARABAR
über
CHERRY COBBLER
Sweet and Salty Fruit & Nut Bar
NET WT 1.42 OZ (40g)


Carr's
Assorted Biscuits
for Cheese


Worthington
Diced
Chik
Made With
Vegetable Protein


Worthington
Loma Linda
Linketts
Vegetable and Grain Protein Links


Worthington
Loma Linda
Swiss
Stake
with Gravy
Made with Vegetable Protein


Annie Chun's
ALL NATURAL ASIAN CUISINE
NOODLE BOWL
TERIYAKI
Fresh cooked Hokkien noodles and sauce with tofu, carrot and spinach toppings

Annie Chun's
CHOW MEIN
Classic Chinese Noodles

Annie Chun's
NOODLE EXPRESS
TERIYAKI
Microwavable

NEW IMPROVED FLAVOR
STEVIA
Select
All natural
Alcohol-Free
STEVIA
Select
Raging Raspberry

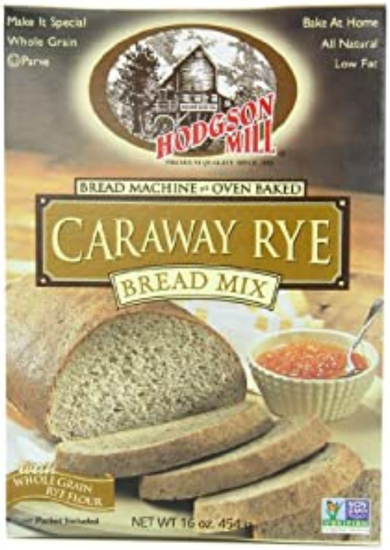
SweetLeaf
LIQUID STEVIA
Vanilla Crème

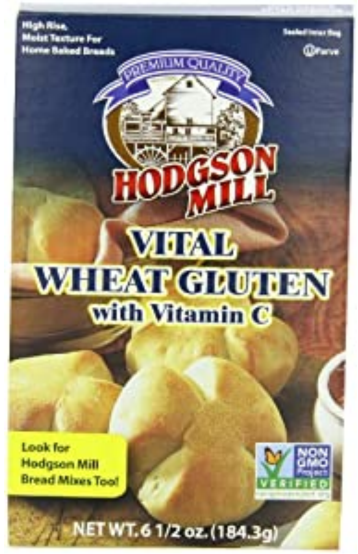
SweetLeaf
Sweet Drops
Sweetener
GRAPE

Perfect for Snacking and Baking
NATIVE FOREST
USDA ORGANIC
100% ORGANIC
PINEAPPLE
Chunks
In Organic Pineapple Juice
NET WT 14 OZ 400g

FruitaBü

BARE FRUIT
100% Organic
Bake-Dried Fuji
APPLE CHIPS


Make It Special
Whole Grain
Bake At Home
All Natural
Low Fat
HODGSON MILL
BREAD MACHINE or OVEN BAKED
CARAWAY RYE
BREAD MIX
NET WT 16 oz.


High Rise,
Moist Texture For
Home Baked Breads
PREMIUM QUALITY
HODGSON MILL
VITAL
WHEAT GLUTEN
with Vitamin C
Look for
Hodgson Mill
Bread Mixes Too!
NET WT. 6 1/2 oz. (184.3g)


PALEO
BREAD


B115 BEST BY
vigo
CALDO DE JAMON
HAM
FLAVORED
SOUP BASE
NET WT. 10.5 OZ. (298 g)


RIGHT FOODS
DR. McDOUGALL'S
ORGANIC MAPLE
made with organic gluten free oatmeal


Baxters
LUXURY
French
Onion
soup


Coconut Bar
Almond


Mr
KIPLING
Exceedingly good cakes
5 Mini
Battenbergs


curlywurly
curlywurly
curlywurly
curlywurly
curlywurly
curlywurly


KIRKLAND
WALNUTS
3 LBS.


Wright's
Natural
LIQUID SMOKE


CELTIC
SEA SALT
FINE GROUND


DAVIDSON'S
Herbal
Mulling Spices


AZAR
MACADAMIAS
NET WT. 32 OZ (2 LB) 907g

Appendix 6. List of users for whom recommendations are improved

	User index	NDCG TMPG-BL	NDCG VL-PCA	improvement
1	1628	0.3333	0.8246	147.38%
2	3853	0.2612	0.6223	138.21%
3	4027	0.2724	0.6389	134.57%
4	478	0.2712	0.6358	134.42%
5	3382	0.2708	0.6341	134.14%
6	2134	0.2711	0.6344	134.03%
7	1102	0.2733	0.6330	131.58%
8	1599	0.2756	0.6302	128.63%
9	2305	0.2812	0.6349	125.76%
10	1746	0.2801	0.6322	125.67%
11	2166	0.2755	0.6209	125.35%
12	434	0.2277	0.5000	119.62%
13	1928	0.2789	0.6121	119.43%
14	2898	0.2891	0.6256	116.42%
15	1734	0.2932	0.6329	115.83%
16	903	0.3010	0.6368	111.54%
17	916	0.2988	0.6317	111.40%
18	3150	0.3123	0.6289	101.36%
19	909	0.2789	0.5089	82.44%
20	1189	0.3562	0.6351	78.30%
21	305	0.3869	0.6781	75.29%
22	3066	0.3734	0.6461	73.04%
23	396	0.2560	0.4307	68.26%
24	1530	0.3895	0.6309	62.00%
25	166	0.4307	0.6934	61.01%
26	1123	0.3010	0.4821	60.16%
27	65	0.4980	0.7887	58.37%
28	2338	0.4005	0.6309	57.53%

29	3513	0.4307	0.6781	57.45%
30	1069	0.3333	0.5089	52.67%
31	1642	0.3442	0.5073	52.19%
32	1150	0.3353	0.5000	49.11%
33	251	0.4281	0.6331	47.90%
34	2478	0.4307	0.6345	47.32%
35	3291	0.4317	0.6309	46.16%
36	828	0.5275	0.7684	45.65%
37	2886	0.6137	0.8935	45.60%
38	2500	0.2354	0.3422	45.38%
39	3792	0.4307	0.6216	44.34%
40	1723	0.4771	0.6826	43.07%
41	3385	0.3562	0.5000	40.37%
42	991	0.5000	0.6934	38.69%
43	260	0.6392	0.8744	36.80%
44	2730	0.2626	0.3562	35.62%
45	1431	0.3728	0.5042	35.27%
46	609	0.2891	0.3869	33.83%
47	3956	0.4582	0.6131	33.83%
48	2908	0.2500	0.3333	33.33%
49	2103	0.5655	0.7500	32.63%
50	1522	0.4057	0.5283	30.23%
51	1213	0.5785	0.7523	30.05%
52	2014	0.2789	0.3562	27.70%
53	3209	0.5965	0.7545	26.49%
54	39	0.5000	0.6309	26.19%
55	1802	0.6131	0.7717	25.86%
56	3338	0.2314	0.2891	24.93%
57	3871	0.5091	0.6309	23.93%
58	1577	0.5022	0.6211	23.67%

59	2365	0.2891	0.3562	23.23%
60	1952	0.2277	0.2789	22.52%
61	451	0.4217	0.5166	22.51%
62	2523	0.2702	0.3213	18.91%
63	467	0.4281	0.5089	18.87%
64	3071	0.4057	0.4771	17.61%
65	2952	0.7470	0.8702	16.49%
66	3870	0.8179	0.9509	16.27%
67	225	0.4307	0.5005	16.21%
68	3278	0.6826	0.7925	16.10%
69	91	0.3491	0.4018	15.09%
70	784	0.4434	0.5089	14.76%
71	173	0.3015	0.3453	14.53%
72	2924	0.7153	0.8155	14.00%
73	853	0.4307	0.4861	12.86%
74	267	0.6177	0.6965	12.76%
75	1081	0.6395	0.7153	11.86%
76	3284	0.7193	0.8038	11.74%
77	3380	0.3155	0.3512	11.31%
78	2548	0.6781	0.7500	10.60%
79	1101	0.7487	0.8255	10.25%
80	558	0.6931	0.7599	9.64%
81	2637	0.4653	0.5089	9.36%
82	734	0.6928	0.7574	9.32%
83	3954	0.8377	0.9111	8.77%
84	3929	0.6934	0.7530	8.59%
85	2504	0.6309	0.6851	8.59%
86	587	0.7511	0.8155	8.57%
87	1999	0.6548	0.7103	8.48%
88	3607	0.6977	0.7545	8.14%

89	639	0.7598	0.8198	7.90%
90	2081	0.6983	0.7522	7.71%
91	3191	0.6624	0.7103	7.24%
92	1582	0.7501	0.8035	7.13%
93	3210	0.6436	0.6872	6.78%
94	2958	0.5889	0.6284	6.71%
95	2151	0.5584	0.5930	6.18%
96	193	0.8246	0.8752	6.14%
97	596	0.8231	0.8733	6.10%
98	2119	0.7505	0.7935	5.72%
99	3669	0.4771	0.5039	5.61%
100	237	0.2560	0.2702	5.58%
101	940	0.3171	0.3333	5.12%
102	1526	0.7153	0.7519	5.11%
103	3873	0.5056	0.5308	4.99%
104	2136	0.6934	0.7253	4.60%
105	157	0.8323	0.8702	4.56%
106	1811	0.8182	0.8539	4.36%
107	44	0.6934	0.7235	4.34%
108	647	0.4936	0.5144	4.22%
109	1240	0.6313	0.6577	4.18%
110	528	0.7855	0.8182	4.16%
111	2562	0.2891	0.3010	4.14%
112	3654	0.8935	0.9304	4.12%
113	2120	0.7921	0.8246	4.11%
114	1772	0.6250	0.6505	4.08%
115	2645	0.4682	0.4871	4.05%
116	1015	0.6351	0.6577	3.56%
117	1939	0.9304	0.9624	3.44%
118	1209	0.6934	0.7165	3.33%

119	3246	0.7232	0.7466	3.24%
120	1887	0.4593	0.4733	3.05%
121	919	0.6934	0.7144	3.03%
122	1827	0.2626	0.2702	2.89%
123	199	0.6208	0.6366	2.55%
124	1750	0.6505	0.6667	2.48%
125	602	0.6351	0.6505	2.42%
126	395	0.6290	0.6436	2.32%
127	1210	0.8539	0.8702	1.91%
128	3920	0.6199	0.6313	1.84%
129	2643	0.2314	0.2354	1.74%
130	3715	0.3961	0.4022	1.54%
131	1674	0.7500	0.7606	1.42%
132	3007	0.6270	0.6337	1.08%
133	4004	0.6505	0.6517	0.18%
134	3202	0.2471	0.2473	0.08%

Appendix 7. Brands and frequency of the appearance for the improved recommendations

Brand	Freq.	Brand	Freq.	Brand	Freq.
Bariani Olive Oil Company	47	Frontier	3	Vogue Cuisine	1
Skippy	41	Pamela's Products	2	Vigo	1
Nature's Path	29	Nile Spice	2	Newman's Own	1
Lindt	19	Lavazza	2	Vegemite	1
Hot Kid	17	Albanese	2	Nestle	1
Tinkyada	17	Hain	2	Near East	1
PG Tips	15	Red Vines	2	Truvia	1
Bob's Red Mill	14	Kashi	2	Stevita	1
Stauffer's	9	Lakewood	2	Trident	1
Stevita Stevia	9	Kitchens Of India	2	Stash Tea Company	1
Good Earth	9	Lotus Foods	2	Stonewall Kitchen	1
Uncle Lee's Tea	8	Splenda	2	Southeastern Mills	1
Nature Valley	8	Wholesome Sweeteners	2	Snyder's of Hanover	1
Haribo	7	Torani	2	Smucker's	1
Hormel	7	Bigelow Tea	2	Sun Maid	1
Planters	6	Taylors of Harrogate	2	Swad	1
Coffee People	5	Daelia's	2	Sezme	1
Mars	5	Walkers	2	Selina Naturally	1
Libby's	4	Annie's Homegrown	2	Swanson	1
Teeccino	4	YOGI	2	Seitenbacher	1
DeBoles	4	Traditional Medicinals	2	San Pellegrino	1
Quaker	3	Nonni's	1	S&amp'B	1
Celestial Seasonings	3	Xlear	1	SweetLeaf	1
Stephen's Gourmet	3	Numi	1	Rolo	1
Betty Crocker Baking	3	Nong Shim	1	TABASCO brand	1
Eden	3	Pop-Tarts	1	Rice Select	1
Twinings	3	Nielsen-Massey	1	Teas' Tea	1
Let's Do Organic	3	Walden Farms	1	Red Star	1

Appendix 8. Brands and the number of items available in the market

Brand	# items	Brand	# items	Brand	# items
Bob's Red Mill	177	Navitas Naturals	26	Harmony House Foods	16
Green Mountain Coffee	62	Melitta	25	Tone's	16
Frontier	56	Hodgson Mill	25	Crown Prince	16
Celestial Seasonings	56	Nong Shim	24	Folgers	15
Kirkland Signature	50	Libby's	24	V8	15
YOGI	48	General Mills Cereals	23	Bigelow Tea	15
Nature's Path	48	Nutiva	23	Pacific Natural Foods	15
Lipton	46	Now Foods	23	Newman's Own	15
Unknown	44	Harney &amp' Sons	23	Tasty Bite	15
Simply Organic	42	Wholesome Sweeteners	23	Numi	15
Quaker	36	Trader Joe's	22	YummyEarth	14
McCormick	34	Betty Crocker Baking	22	Campbell's	14
Dr. McDougall's Right Foods	34	Traditional Medicinals	22	Muir Glen	14
Starbucks	33	Kraft	20	Nestle	14
Twinings	33	Lindt	20	Schar	14
Knorr	32	SweetLeaf	20	Tinkyada	14
Pamela's Products	31	Crystal Light	19	Skippy	14
Keurig	31	Lundberg	18	Lavazza	14
Kashi	31	Annie's Homegrown	18	Coffee-mate	14
Eden	31	Earth's Best	18	Sun Maid	14
Maruchan	30	Cadbury	18	Thai Kitchen	14
Amy's Organic	29	Eight O'Clock Coffee	18	Walkers	13
Haribo	29	Blue Diamond Almonds	17	Mother Earth Products	13
Hershey's	28	Walden Farms	17	Go Raw	13
Hormel	28	Barry Farm	17	Kitchens Of India	13
Barilla	27	Enjoy Life Foods	17	Stash Tea Company	13
Special K	27	Back to Nature	16	Kellogg's	13
Ghirardelli	27	Keebler	16	Starwest Botanicals	13

Appendix 9. Brands and the number of items available in the market

2nd random image: id = 7762, asin = B00D2IHRFS

D = 50.8

D = 67.2

D = 69.3

D = 69.4

D = 70.2

3rd random image: id = 3216, asin = B001EQ5AVI

D = 42.5 D = 43.7 D = 43.9 D = 57.2 D = 57.9

4th random image: id = 6569, asin = B005Q8IXXM

D = 60.4 D = 60.4 D = 62.3 D = 63.4 D = 63.9

5th random image: id = 5336, asin = B0042ROCYM

D = 59.0

D = 59.1

D = 59.2

D = 59.3

D = 59.4

Appendix 10. Ranking performance of the best models on the train and validation sets

- Performance on the training set

	RR	TMPG-BL	VL-PCA	VL-tSNE
$NDCG@20$	1.05e-2	14.904e-2 (1382%)	16.720e-2 (12.19%)	16.239e-2 (8.96%)
K	-	10	20	200
λ	-	10	10	10

- Performance on the validation set

	RR	TMPG-BL	VL-PCA	VL-tSNE
$NDCG@20$	1.93938e-3	8.0896e-3 (317%)	8.6822e-3 (7.32%)	8.35927e-3 (3.33%)
K	-	10	20	200
λ	-	10	10	10