User Modeling and User-Adapted Interaction (2021) 31:867–905
https://doi.org/10.1007/s11257-021-09293-9

Improving cold-start recommendations using item-based stereotypes

Nourah AlRossais[1] · Daniel Kudenko[2] · Tommy Yuan[1]

Received: 7 November 2019 / Accepted in revised form: 23 May 2021 / Published online: 21 September 2021

Abstract

Recommender systems (RSs) have become key components driving the success of e-commerce and other platforms where revenue and customer satisfaction is dependent on the user's ability to discover desirable items in large catalogues. As the number of users and items on a platform grows, the computational complexity and the sparsity problem constitute important challenges for any recommendation algorithm. In addition, the most widely studied filtering-based RSs, while effective in providing suggestions for established users and items, are known for their poor performance for the new user and new item (cold-start) problems. Stereotypical modelling of users and items is a promising approach to solving these problems. A stereotype represents an aggregation of the characteristics of the items or users which can be used to create general user or item classes. We propose a set of methodologies for the automatic generation of stereotypes to address the cold-start problem. The novelty of the proposed approach rests on the findings that stereotypes built independently of the user-to-item ratings improve both recommendation metrics and computational performance during cold-start phases. The resulting RS can be used with any machine learning algorithm as a solver, and the improved performance gains due to rate-agnostic stereotypes are orthogonal to the gains obtained using more sophisticated solvers. The paper describes how such item-based stereotypes can be evaluated via a series of statistical tests prior to being used for recommendation. The proposed approach improves recommendation quality under a variety of metrics and significantly reduces the dimension of the recommendation model.

✉ Nourah AlRossais
nar537@york.ac.uk

Daniel Kudenko
kudenko@l3s.de

Tommy Yuan
tommy.yuan@york.ac.uk

[1] University of York, York, UK

[2] L3S Research Center, Leibniz University Hannover, Hannover, Germany

Keywords Recommender systems · Stereotypes · New item · New user · Cold start

1 Introduction

The taxonomy and applications of recommender systems (RSs) have been widely studied, ranging from user-based collaborative filtering (CF)(Billsus and Pazzani 1998; Breese et al. 1998; Goldberg et al. 1992, 2001; Herlocker et al. 1999, 2000, 2002, 2004), to the alternative approach of content-based filtering (CBF) (Deshpande and Karypis 2004; Linden et al. 2003; Lops et al. 2011; Sarwar et al. 2001, 2002). While user-driven CF methods rely on the opinions of similarly minded users to predict a rating, CBF systems look at preferences given to similar items. The main drawbacks of filtering-based recommendations include poor computational efficiency caused by the sparsity of the data, overspecialisation leading to a lack of novelty and serendipity, and the cold-start treatment (i.e. new user and new item problems).

Cold-start phase is defined as the situation in which the RS needs to cope with a new user first approaching the platform or a novel item being launched. While the majority of the literature focuses on user CF and CBF as well as hybrid approaches of the two, the review work of Jannach et al. (2012) has shown that less than 5% of the existing research addresses the new user and new item problems (see Schein et al. 2002).

Recently, the cold-start problem has been an active research subject, with several works addressing techniques tailored to handle either the new user or the new item problem (Felício et al. 2017; Frolov and Oseledets 2019; Kluver and Konstan 2014; Mirbakhsh and Ling 2015). For example, Fernández-Tobías et al. (2019) propose to solve the new user problem via latent rating patterns discovered using item metadata in a factorisation-based method. Deldjoo et al. (2019) focus instead on the new item problem, using innovative audio and video metadata in the movie recommendation domain. The emerging pattern is characterised by heavier use of the available metadata context of both items and users, coupled often with factorisation-based methods. The general findings suggest that during extreme cold starts it is difficult for any of the researched systems to significantly improve over basic baseline models. Moving away from pure cold start, the researched models improve over the baseline once the first few ratings have been collected. The present work seeks to quantify the potential for improvement in extreme cold start deriving from stereotypes.

During the cold-start phases, when there is little or no feedback for an item or a user, one can resort to finding similarities in the metadata of the new user (item) and the existing users (items). Thus, stereotypes can be built on the idea that users with similar features may also share similar broad-level preferences and that items with similar features may be preferred by certain types of users. Therefore, a stereotype can be viewed as an aggregation of the characteristics of the items or users that allow one to group items and users in general classes. The search for similarities between the new user or new item and the rest of the users and items population, when little is known about a user or an item, rests on the correct categorisation via

metadata. It would be unfeasible, and likely error-prone, to rely on expert human knowledge to correctly classify a new user or a new item to the platform.

In this research we study the possibility of improving RS performance during cold start by adopting a different point of view from those of previous works, that of rating agnostic stereotypes. We wish to demonstrate:

- how stereotypes can be built automatically for the most common types of features, and most importantly in a way that is *independent of the user-to-item preferences*.
- the benefits of building stereotypes independently of the user-to-item matrix, which result in a basis that improves the recommendation quality of a range of RS.
- the better recommendation quality during cold start which reaches beyond the simple improved accuracy, and it instead embraces several aspects that are deemed to determine positive recommendation characteristics.
- how a series of statistical tests can be formulated with the objective to evaluate the stereotypes as a base for a RS. In particular the stereotypes stability and their ability to capture user-to-item preference traits. This last characteristic is deemed important but it is often overlooked in techniques that are viewed as black boxes and RS driven by deep learning. We wish to gather whether the stereotypes learned have the ability to represent user preferences, in a way that is independent from assessing recommendations.

The paper is organised as follows: Section 2 reviews work related to addressing the cold-start problems. Section 3 presents the underlying ideas on how to generate rating and preference-independent item-based stereotypes. Section 4 shows the results of the automatic procedures for assembling stereotypes for the two datasets: the integrated MovieLens/IMDb and the Amazon. Section 5 discusses a statistically driven approach to the evaluation of the stereotypes. The application of the stereotypes in the context of recommendation for new user and new item is given in Sect. 6. Section 7 provides an assessment of the proposed systems against recommendations driven by standard factorisation methods as well as factorisation methods with the embedded item and user metadata. Finally, in Sect. 8, we draw our conclusions and identify future work.

2 Related work and contribution

Elahi et al. (2018) provides a comprehensive review of the recent developments in addressing the cold-start problems. Historically, cold-start phases have been addressed by implementing hybrid recommendation techniques, combining collaborative and content-based filtering (Adomavicius and Tuzhilin 2005; Barkan et al. 2019; Burke 2002; Cella et al. 2017; Cohen et al. 2017; Frolov and Oseledets 2019; Ricci et al. 2015). Deshpande and Karypis (2004) argue that the new user and new item problems can be related to the sparsity of the rating matrices. Users can be

grouped based on the available information about them. For example, see the use of demographics information by Pazzani (1999) and Krulwich (1997).

Other works suggest extracting information about the new user from social media—for example, see Sedhain et al. (2014), Alahmadi and Zeng (2015) and Du et al. (2017)—or linking across domains—for example, see Enrich et al. (2013), Fernández-Tobías et al. (2019) and Mirbakhsh and Ling (2015)—by using the knowledge of ratings and tags assigned by the users to items in an auxiliary domain (e.g. movie ratings) to model preferences in a target domain (e.g. book purchases). Fernández-Tobías et al. (2016) proposed three strategies of user personality information and applied them to CF to solve the new user problem, while Nasery et al. (2016) and Kalloori and Ricci (2017) incorporated feature-based preferences between items to alleviate the cold-start problem.

Special approaches for handling the new user and new item problems consist of requiring a first compulsory training period of the RS on every new user and new item before performing recommendations (see Elahi et al. 2014; Linden et al. 2003). Such works demonstrate the inherent difficulties of handling pure cold starts; they usually improve recommendations over simpler baseline models once the users and items become 'known' via a series of directly expressed preferences or, as Nasery et al. (2016) suggests, as indirect preferences expressed to features.

A range of techniques that increase efficiency by reducing the cardinality and sparsity of the rating and consumption matrix include those built upon the idea of factorisation of the user-to-item rating matrix (Braunhofer et al. 2015; Frolov and Oseledets 2019; Koren 2008; Sarwar et al. 2000). These techniques, among which the singular value decomposition (SVD) is probably the most popular thanks to the success obtained in the Netflix grand prize (Koren et al. 2009; Koren 2009), aim to reduce the dimensionality of the rating matrix by projecting the ratings over a latent factor space. This process enables researchers to determine how users rate items. Most of the studies referenced in this work, when reaching the prediction stage, rely on factorisation techniques to reduce the dimensionality of the user-to-item matrix or to provide a latent space where clustering methods are applied (for example, see Braunhofer et al. 2015).

In addition to the above-mentioned methods, it is also preferable to use classes like stereotypes as a mechanism for generating recommendations for users in the cold-start scenario. A range of studies (Brajnik and Tasso 1994; Kay 1994a, b) followed the ideas of user-based stereotyping presented by Rich (1979). Up until the late 90s, the construction of stereotypes had been almost exclusively manual and driven by expert knowledge. Lamche et al. (2014) conducted an evaluation of the effectiveness of a user-based stereotypes recommender system for the mobile fashion domain. However, the stereotypes were identified by the author beforehand. Kamitsios et al. (2018) presented a stereotype-based user model in an educational game to offer personalisation according to a player's skill. Likewise, the stereotypes (i.e. mode of the game) were identified by the author. The effectiveness of stereotype-based RS in digital library 'Sowiport' was measured by Beel et al. (2017). The results were not encouraging as the authors assumed one class of stereotypes only (i.e. students and researchers). Thus, all Sowiport visitors were receiving similar recommendations related to specific topics. The work by ALRossais and Kudenko

(2018a) provides a first attempt at evaluating stereotype-based and non-stereotype-based RS. Nevertheless, in such work, stereotypes were still built using expert knowledge.

The work of Paliouras et al. (1999) provides one of the first attempts to 'learn' the user and item classes via supervised learning techniques. Grouping of features, or clustering, was soon introduced as a way to address the sparsity of rating matrices, especially in the context of classifiers and probabilistic-based systems (Eskandanian et al. 2017; Khalaji et al. 2012; Ungar and Foster 1998). A wealth of research has focused on the application of classification and grouping methodologies to CF and CBF for clustering—see O'Connor and Herlocker (1999)—and for forests of trees—see Koprinska et al. (2007). However, this research does not address the cold-start phases.

Adomavicius and Tuzhilin (2005) and Braunhofer et al. (2015) attempted to apply grouping methodologies to the cold-start phase and, in particular, to the new user case. In the extreme cold-start scenario, if no data is available, the system may recommend popular items or items with the highest average ratings, as discussed by Fernández-Tobías et al. (2016). Recent work on clustering for RS indicates its popularity as a method for enhancing recommendation quality (Rimaz et al. 2019). It is important to note that the majority of the clustering-, similarity- and dimensionality-reduction approaches developed for filtering-based systems or to solve cold-start problems all operate on the user-to-item preferences (or ratings) matrix (Du et al. 2017; Felício et al. 2016, 2017; Kluver and Konstan 2014; Mauro and Ardissono 2019; Mirbakhsh and Ling 2018; O'Connor and Herlocker 1999; Sacharidis 2017; Sollenborn and Funk 2002; Shani et al. 2007; Wibowo et al. 2018). Recently, groupings of users and items have been performed via neural networks-driven text embedding, like word2vec doc2vec, leading to an algorithm capable of grouping users and items via their metadata. These approaches have been tested for cold-start scenarios in (Misztal-Radecka et al. 2020).

The present work approaches the problem differently by investigating the possibility of obtaining a viable RS that uses stereotypes generated directly via the feature's metadata similarities instead of ratings and preferences. The concepts of rating's agnostic stereotype had been preliminary introduced in (ALRossais and Kudenko 2019), the present work builds on such concepts to derive a more formal definition and evaluation of stereotypes in the context of cold-start recommendations. Ratings- and preferences-agnostic stereotypes lead to significant dimensionality reduction when the RS is trained but, at the same time, retain sufficient flexibility for capturing general preference traits in a population of users.

The main contribution of the present work is to highlight the benefit to the RS community of adopting stereotypes during cold start, especially those that have been built using item and user metadata relationships, not embedding past user-to-item preferences. Every result presented from Sect. 6 onward is data that is shown to corroborate the ability of stereotypes to provide an alternative way to get better recommendations (under several metrics) for the new user and new item phases. In examining the results the second most important finding arises, as a side product of the research, namely that the improvement in cold-start recommendations appears to be independent of the recommendation technique used, providing RS researchers with

an extra dimension for improvement. Other secondary novelty contributions are presented in the work, a formal test framework to evaluate stereotypes before using them in a RS, a metric to evaluate serendipity for complex categorical features.

3 Constructing item-based stereotypes

While stereotypes have been loosely introduced earlier in the paper, the objective here is to define associations between metadata features for both users and items. Such associations prove helpful to an RS in categorising both new users and new items to generate recommendations when few reviews are available. This section provides a general explanation of how such metadata-driven relationships may be discovered.

When considering a dataset that is complex and rich in item and user metadata, such as the combined dataset of MovieLens and IMDb (ALRossais and Kudenko 2018b), one must consider a range of features, from simple numerical to categorical and complex categorical. A categorical variable is defined as complex when (1) it cannot be easily translated into a numerical variable via encoding, (2) when the semantics of the categories play an important role in the optimal determination of stereotypes and (3) when it is multi-choice (e.g. there is no predefined minimum or maximum number of labels that describe the item or user). These variables can be viewed as multiple-choice answers on a questionnaire, with the underlying idea being 'pick all that apply'. In the movie domain, typical examples of complex categorical features include the 'genre' and 'keywords' used for labelling movies. For instance, for one item the genre may be categorised as 'drama', whereas for another item might be 'drama' in addition to 'romance' and 'historic'.

Clustering-based algorithms applied to item metadata provide a direct representation of stereotypes along with valuable insights into which features drive class separations. The main challenge in the application of a clustering algorithm resides in the standardisation of the data. Most clustering algorithms (e.g. k-means and its variations) work with Euclidean distances. For categorical features, the concepts of distance and order may be difficult to define and, when significant, may introduce unexpected false relationships.

The k-mode algorithm (Huang 1998) was introduced to deal with categorical data. The clustering cost function is minimised using a frequency-based method to update the modes. Several marginal improvements have been introduced (Aranganayagi and Thangavel 2009; Sangam and Om 2015). The k-modes clustering algorithm can be initialised in different ways. According to Huang (1998), the artefacts (i.e. the centroids) are placed randomly across the feature space, and according to Cao et al. (2009), they should be placed based on their initial estimated density. This work demonstrates that k-modes may not be the preferred choice for stereotyping, and we introduce an alternative approach.

Recently, Cao et al. (2017) suggested an algorithm for clustering categorical set-valued data. However, the algorithm fails to consider the effect of correlation between labels. For providing recommendations, our proposed method rests on the effects of multi-label correlation.

3.1 Stereotypes for complex categorical features

For a complex categorical feature, there exist several entries where multiple labels are assigned to the same item. One can compute the correlation matrix between categories $R_{i,j}$ (see Tsokos (2009) for the definition). Ad-hoc correlation groups, following Zimek (2008), can convey information about the similarities and dissimilarities of the labels involved. However, to formalise further, the clustering of the correlation matrix can be performed. Hence, it is necessary to introduce both a metric defining distances between a pair of observations and a 'linkage' criterion to define the similarities among groups (clusters) of observations. Suitable metrics from the correlation entries can be obtained in several ways (see Podani (2000) for a range of dissimilarity metrics examples). In the context of this study, a simple linear metric (also referred to as penalty P) is adopted.

$$P_{i,j} = 1 - |R_{i,j}| \tag{1}$$

In the hierarchical clustering literature, many alternative linkages have been proposed (Friedman et al. 2001), with the ***single, complete*** and ***Ward*** linkages among the most widely used. The suggested method for automatically creating stereotypes for complex categorical features rests on the systematic truncation of the dendrogram of the hierarchical clustering procedure. One must choose which penalty function to adopt. A quadratic penalty tends to compress excessively toward 1.0 entries that have low correlations (e.g. less than 0.4 in absolute value). The resulting dendrograms would appear too compressed when using correlation matrices that have low correlations in magnitude. Instead, the linear penalty (1) is better suited for exploring scenarios where the correlations are low on average.

Dendrogram truncation criteria can be implemented by examining how the linkage merge iterations shape the clusters discovered, moving up the dendrogram branches from stronger links toward weaker ones. From a certain point forward, the discovered structures begin to merge toward a single cluster. This dynamic can be summarised by monitoring the average cluster size and the number of clusters formed up to a given iteration. Therefore, the cut-off procedure can be implemented via a dual criterion: (1) by looking for the last local plateau in the number of clusters as a function of the iteration and (2) by applying a reverse-elbow procedure to the average cluster size. The two criteria can also be coupled by taking the ratio, at any iteration, of the average cluster size divided by the number of clusters formed, which is referred to as the *dendrogram iteration ratio*. The cut-off procedure then reduces to finding the highest iteration exhibiting a local minimum in the iteration ratio. The only scenario in which this idea would fail is in the case of a monotonically increasing dendrogram iteration ratio, which is found when no true underlying groups exist in the data (e.g. the data represents a collection of items that do not belong together, and it is grouped into a single, ever-growing cluster). In this special case, the conclusion is that the feature cannot be split into stereotypes. The complete procedure to create stereotypes for complex categorical features is illustrated in Algorithm 1.

Algorithm 1 Algorithm for assembling stereotypes for complex categorical features

Compute the correlation matrix
Compute the average non-diagonal correlation value V

if V is low on average (for example less than 0.4) **then**
  Use linear dissimilarity
else
  Use quadratic dissimilarity
end if

Hierarchically cluster the correlation matrix

Assemble the dendrogram
Compute the dendrogram iteration ratio R
Find the highest iteration at which R displays a local minimum

if at least one local minimum **then**
  Assemble the stereotypes by cutting the dendrogram between such iteration and the successive one
else
  No stereotype
end if

3.2 Stereotypes for numerical features

When working with numerical features, we are interested in creating generalisations that may be useful for an RS in identifying patterns. For example, in the movie domain, we may discover that users in their 40s like 80s movies, while teenagers prefer high-budget movies. These basic examples show potential relationships between user age groups (a numerical feature) and other numerical features of the items (e.g. the year of production or the budget).

Numerical features can be discrete, continuous, or mixed and either single or multimodal. When a feature is multimodal, a natural method for creating numerical stereotypes is to *select the most relevant modes and intervals around them*. This definition seems operatively simple. However, the simplicity is challenged by the fact that distributions are derived via numerical approximations of the probability density functions, for example, via a kernel density estimation (KDE) (Chen 2017).

Numerical approximations often result in a 'wiggled' graph with each local maxima potentially indicative of a mode. An algorithm is required to automatically classify the peaks in a histogram (or KDE) according to their significance. This problem is not as simple as ranking local maxima in a function. Figure 1 shows an idealised fictitious probability distribution with four local modes. If the local maxima in the figure were ranked via their probability density (i.e. ranking them as A, B, C and D), the ranks would not be representative of the structures. In Fig. 1 it is notable that peak 'A' is the most significative effect, and peaks 'D' and 'C' are somewhat

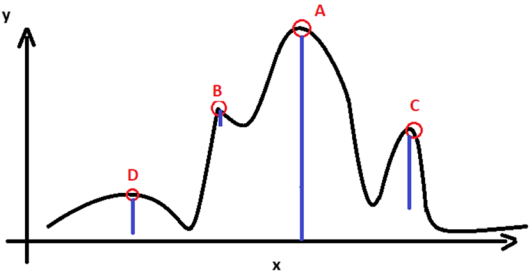

Fig. 1 Fictitious probability density approximation

lower-level effects that represent well-defined areas of the distribution. Peak 'B' likely represents noise around 'A'.

A formal solution to this problem was provided in the mathematical branch of computational topology and particularly in the field of persistent homology (Edelsbrunner and Harer 2010). The concept of significance (i.e. persistence) can be used to such a scope. Persistence is better explained with a classic topology example: the function is analogous to a submerged mountain, with an initial water level above the global maximum A. As the level drops, whenever it reaches a local maximum, a new island is born, and whenever it reaches a local minimum, two islands merge (the lower island merges into the higher). The lifespan of an island is correlated to its significance, also called persistence. In Fig. 1, the persistence of each local maxima is shown via the vertical blue bars, which allows the desired ranking of the local maxima: (A, C, D and B).

In this study, numerical features are divided into two categories. **Type I** features are such that the sample distribution has a number of significant modes greater or equal to two, and the estimated proportion of the population sample that can be attributed to such modes is relevant (i.e. greater than or equal to 60%). Features that do not respect such conditions are called **Type II** features, and the stereotypes are built using percentile-driven intervals (e.g. quartiles).

4 Stereotype creation experiment

To demonstrate and assess the proposed recommendation methodology, we performed the cold-start experiments using two datasets: the integrated set of MovieLens with added metadata from the IMDb database and the publicly available dataset of reviews for item purchases from Amazon.com. The MovieLens dataset, one of the most popular datasets for recommendation problems, continues to be widely used in the research literature (Eskandanian et al. 2019; Harper and Konstan 2016; Trattner and Jannach 2020; Wasilewski and Hurley 2019; Zheng et al. 2018). Fewer works have considered integrating the MovieLens dataset with the item-based metadata from IMDb. Two such recent works are Rana and Bridge (2018) and Barkan et al. (2019). In this research, all the item metadata features available in IMDb

are integrated into the MovieLens reviews, as discussed by ALRossais and Kudenko (2018b).

The second dataset from Amazon.com has also been the subject of intensive investigation, due to the high sparsity, both in the normal recommendation context (Musto et al. 2017; Wibowo et al. 2018) and in cold-start scenarios (Zheng et al. 2018). This research focuses on two subsets of the large Amazon dataset (He and McAuley 2016), namely: 'Sport and Outdoors' and 'Clothing, Shoes and Jewellery'.

After addressing the users and items with poor or missing metadata, the resulting sizes of the datasets are summarised in Table 1. This section explains the stereotype creation experiments, showing key results for the MovieLens/IMDb dataset. Similar results were obtained for the Amazon.com dataset, which is not shown in this section for brevity but used in later sections for recommendation purposes.

The two dataset selected with the scope of illustrating the proposed methodology both display a range of features which span the types of numerical, categorical and complex categorical. These three metadata types can be thought of representing the most widely encountered features across a range of domains. They are not by any means an exhaustive set; several modern feature types exist that are specialised for each particular domain, for example, visual features for movies as discussed in Deldjoo and Cremonesi (2018) and references within. While such specialised features may be critical for recommendations in their respective domains, they are not considered within this scope as to keep the introduction of our methodology as general and domain independent as possible during the formulation. The specialisation of the proposed methodology to domain specific fields and features is the subject of future work.

4.1 Results for complex categorical features

To illustrate the treatment of a complex categorical feature, we use the MovieLens/IMDb 'Genre' feature as an example. Figure 2 shows the correlations among the feature labels after a simple grouping is performed via a greedy search algorithm. The grouping was performed to improve the display of data, and it does not affect what follows. The average in sample correlation for genre is low in absolute value. Therefore, a linear penalty and Ward linkage are used as suggested in Sect. 3. Figure 3 shows the resulting dendrogram (left) and dendrogram iteration ratio (right), with the iteration number highlighted where the algorithm suggests cutting the dendrogram.

Table 1 Statistics of the MovieLens/IMDb and Amazon datasets

Dataset	MovieLens /IMDb	Amazon Sport & Outdoors	Amazon Clothing, Shoes & Jewellery
No. of items	3,827	478,898	1,136,004
No. of users	6,040	1,990,521	3,117,268
No. of ratings	1,000,209	3,268,695	5,748,920

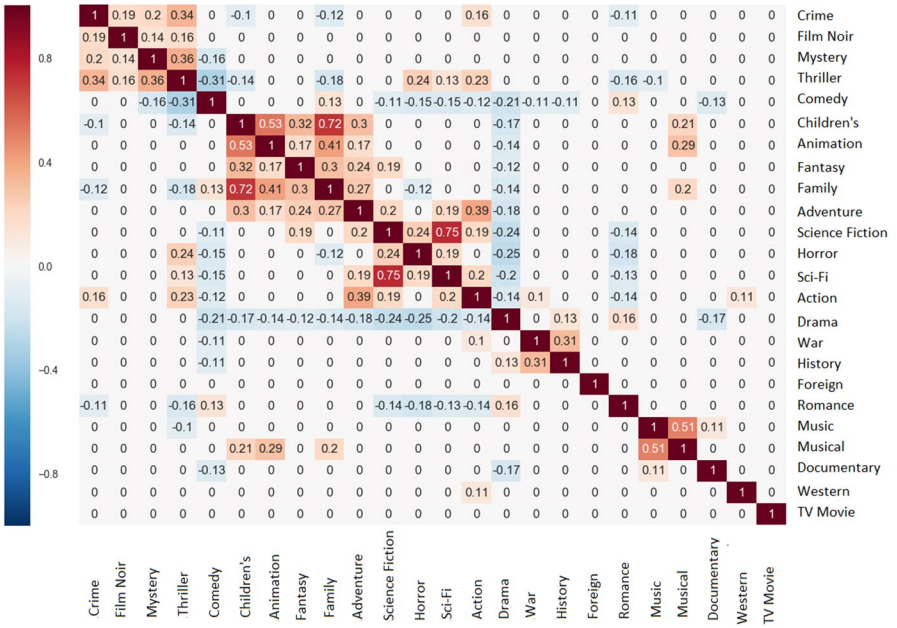

Fig. 2 Correlation matrix for the genre feature

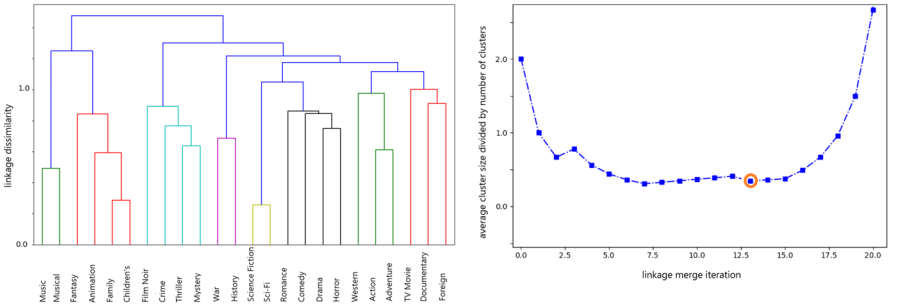

Fig. 3 Genre dendrogram using linear dissimilarity and Ward linkage and the resulting iteration ratio

The stereotypes obtained for the genre feature are shown in Table 2. For reference, and to provide a comparison with an independent methodology, the clusters obtained by applying k-modes are also reported in the same table for $k = 5$, with both the initialisation procedures proposed by Huang (1998) and that by Cao et al. (2009). In the k-mode cases, there was not a single well-identified kink in the elbow plot relating to this methodology (plot not shown), making the choice of k arbitrary. It was observed that the frequency-based concepts underneath k-modes lead to the absence of lower-frequency labels. It can be argued that these labels should indeed be retained as they may represent specific niche user preferences and are required in the recommendation items coordinates as shown later in the paper. Similar results

Table 2 Genre feature: stereotypes and k-modes resulting from centroids composition for five clusters

	Stereotypes	Centroid Composition (Huang)	Centroid Composition (Cao)
1	[Music, Musical]	[Drama, Comedy, Romance]	[Family, Children's, Animation]
2	[Fantasy, Animation, Family, Children's]	[Drama]	[Drama]
3	[Action, Adventure, Western]	[Adventure, Family, Children's, Animation]	[Comedy, Adventure, Family, Fantasy, Children's]
4	[War, History]	[Comedy]	[Comedy]
5	[TV Movie, Documentary, Foreign]	[Thriller, Action]	[Thriller, Action, Adventure, Science Fiction, Sci-Fi]
6	[Film Noir, Crime, Thriller, Mystery]		
7	[Romance, Comedy, Drama, Horror]		
8	[Science Fiction, Sci-Fi]		

were obtained for the feature keywords (not shown) and for Amazon.com's complex categorical features (not shown), providing empirical evidence that Algorithm 1 yields a better grouping approach than k-modes for the stereotype construction of complex categorical features.

4.2 Results for numerical features

The concepts of persistence and barcode suggested in Sect. 3.2 were implemented in Python for a one-dimensional real valued sequence. In the MovieLens/IMDb stereotype generation example, the numerical discretisation of the probability density function was performed using 20–40 bins. We can estimate that a jitter of +/- 2.5% is, in this case, the limit between signal and noise. Therefore, we disregard as not significative all modes associated with a population of less than 4%.

The stereotype construction procedure applied to the numerical features of the MovieLens/IMDb dataset identified seven features of Type I (stereotyped via the persistence procedure discussed in Sect. 3.2) and seven features of Type II (stereotyped via percentiles). For the features spanning several orders of magnitude (e.g. budget, revenue and vote count), the natural logarithm of the feature was used as a transformation to compress the scale. Table 3 shows two examples of stereotyped numerical features: one example of a Type I feature (budget) and one example of Type II feature (release year). For each feature, the table reports the following: (1) the feature mode (i.e. the local modes identified in the distribution of the feature, if any); (2) the barcode, expressed as a probability value that was attached to that mode if any was identified; (3) the fraction of the population that the mode is deemed to represent (or that the stereotype is deemed to represent in the case of Type II features); and (4) the lower and upper bounds associated with each stereotype. The numerical discretisation adopted in the presented case the limit between signal and noise is a jitter of +/- 2.5%. Therefore, all modes associated with a population of

Table 3 Modes of the Type I feature (if the population is less than 4%, the mode is deemed not significant) and Type II feature (using quartiles intervals)

Feature Mode	Barcode (Probability)	Fraction of Population	Stereotype	Lower Bound	Upper Bound
Type I Feature Example: log (budget + 1 [USD])					
0.000	0.4903	0.52	True	0.000	4.024
16.096	0.1481	0.42	True	4.024	18.308
4.024	0.0013	not significative	False	–	-
18.308	0.001	not significative	False	–	-
Type II Feature Example: release year					
–	–	0.25	True	1918	1980
–	–	0.25	True	1980	1993
–	–	0.25	True	1993	1998
–	–	0.25	True	1998	Present

less than 4% are deemed not significative. Budget falls into a bimodal example, leading to its categorisation as a Type I feature, with the following split in between low- and high-budget movies. The release year has no such statistical property. Thus, it is categorised as Type II and is stereotyped in the years group driven by percentile separations.

5 Preliminary evaluation of stereotypes

In this section we introduce a series of comprehensive statistical tests that has been performed to evaluate the quality of stereotypes prior to recommendations. The test framework that follows can be viewed as an extra contribution to the automation of stereotypes creation. For each stereotype created on the in-sample data, the following statistical tests have been formulated to evaluate the stability, accuracy and predictive content of the stereotypes:

- A hard test (the most severe) that checks the entity of the discrepancies between the stereotypes discovered in training and those that would be independently discovered on the 'unseen' test data.
- A soft test in which the stereotypes are generated over the aggregate dataset and used to obtain the 'true' stereotype coordinates for each metadata coordinate of every item in the test data. These coordinates can be used to benchmark the accuracy of the predicted stereotype.
- A predictive power test of the user's preference traits. In this context, ratings and preferences are used to assess how each stereotype is capable of representing the user's population preference traits.

The hard test consists of comparing the stereotypes obtained over the training data with the stereotypes obtained independently over the test data. For complex

categorical features, this comparison is performed by scoring how close two sets of labels are. This provides a measure of precision by looking at one minus the ratio of the number of labels in the set difference between the labels of the stereotype examined and the reference stereotype to the total labels of the reference group. A similar measure of precision can be obtained for the numerical stereotypes by first measuring two quantities: (1) the normalised difference of how far the centre of probability masses are located from stereotypes that represent the same part of the population ($\delta X_P^{s_j,s_i}$), where s_i, s_j are the two stereotypes under comparison, and 2) the difference in probability masses ($\delta P^{s_j,s_i}$). Next, a proxy for precision is defined as $(1 - \delta X_P^{s_j,s_i})(1 - \delta P^{s_j,s_i})$. In both cases, the maximum precision is 1.0, with a 0.0 issued in the limit case of no match between the stereotypes.

Figure 4 displays the results of the more restrictive test; for each feature, the figure displays the average as well as the maximum and minimum precision values recorded across the stereotypes generated for that feature. The results of the comparison between the 8 stereotypes for genre feature is that there is an 88% accuracy (average match), with a median match of 100%—indicating that in most of the stereotypes there is a perfect one-to-one match between the stereotype composition derived on the training data vs those derive on the test data.

For most of the remaining features, the average precision is well above 80%, which indicates a remarkable stability of the stereotypes on unseen data for the dataset under examination. The Amazon experiment (not shown) displays even higher average and minimum values of precision.

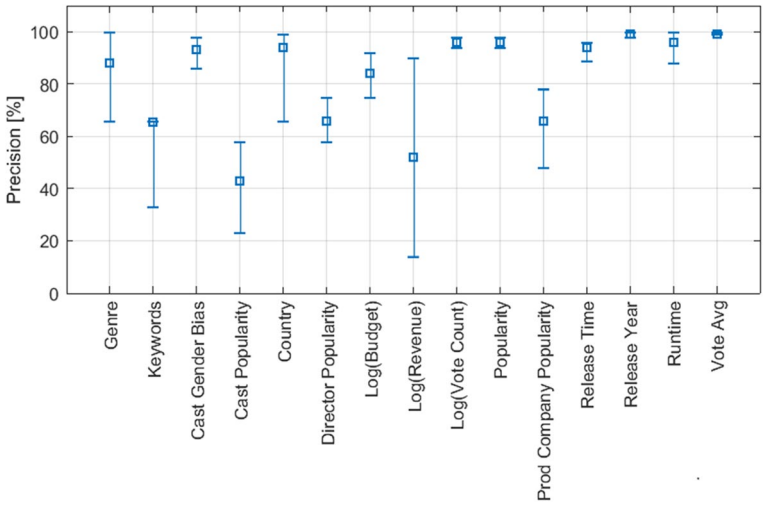

Fig. 4 Restrictive test results: maximum, minimum and average precision values (for the precision metric of the hard test as defined in the text) recorded across the stereotypes generated for that feature. For example, for the feature genre, there is an 88% accuracy (average match), with a median match of 100%—indicating that in most of the stereotypes there is a perfect one-to-one match between the stereotype composition derived on the training data vs those derived independently on the test data

Table 4 Soft test: descriptive statistics of mismatch ratio for MovieLens/IMDb complex categorical features

Feature	Average (%)	Median (%)	Number of nonzero	Average of nonzero (%)
Genre	3.5	0.0	56	73
Keywords	14	0.0	267	61

Table 5 Soft test: stereotypes evaluation for MovieLens/IMDb numerical features

Feature Name	F1-score	Accuracy (%)
Cast gender bias	1.0	100
Log (budget)	1.0	100
Cast popularity	0.77	75
Country	1.0	100
Director popularity	0.86	83
Log (revenue)	1.0	100
Log (vote count)	0.98	98
Popularity	0.99	98
Prod. comp. popularity	0.86	85
Release time of the year	1.0	100
Release year	0.98	98
Runtime	0.98	98
Vote avg.	1.0	100

The soft test is instead performed by using a standard classification approach. Complex categorical features, given their set value property, require ad-hoc metrics for scoring how much the projected stereotype differs from the 'true' stereotypes. This classification is performed by introducing a mismatch ratio, defined for each item in the test data as the ratio of the number of predicted stereotype labels not matching the view of the 'true' stereotypes to the total stereotypes labels of the pair of stereotypes under comparison. The evaluation shown in Table 4 highlights that the distribution of the results over the items of test data is composed of a significant number of perfect matches and a smaller number of considerable mismatches. In most cases, a new item would be well categorised in its stereotypical representation. However, in the few cases where the stereotypical representation is not correct, it will be substantially inaccurate. For the soft test of numerical features, the standard metrics of accuracy and F1-score (Friedman et al. 2001) are presented in Table 5. Overall, the soft test confirms the remarkable stability of the structures discovered, thus paving the road to using such stereotypes in the context of recommendation. Comparable results with the same high level of accuracy were obtained for the Amazon dataset (not shown).

The last statistical test examines the degree to which a user's preference traits can be described via the stereotypes. One can test how biased a user's selections are—i.e. does the user display a statistically significant positive or negative bias toward a stereotype compared to the item's population distribution? For example,

in a simplified scenario, suppose that all the items' metadata could be described via three stereotypes: stereotype A accounts for 40% of items, B and C for 30% of items each. Suppose a user selects 50 items, 40 of type A and 10 of type C so that the user's selections are 80% of type A, 0% of type B and 20% of type C. Based on the number of selections and numbers of items in the population, it is possible to conclude that types A and B show positive and negative preferences, respectively, while nothing can be said about type C. This type of reasoning can be applied via a statistical test whose null hypothesis is as follows: 'For the stereotype investigated, the user consumed a proportion of items that is similar to the proportion expected if the stereotype had no influence in the user's choice'. Rejecting the null means that the stereotype does indeed influence the shaping of the user's preferences.

The mechanics of the test are carried out via the calculation of confidence intervals for the difference of two proportions arising from binomial and multinomial distributions. This statistical problem was studied by Agresti and Coull (1998), and a formula for the confidence interval corresponding to a given statistical significance level was proposed in the same reference.

By examining the results of the Agresti and Coull (1998) test across the thousands of users in the test dataset and across all features and stereotypes, one can develop an understanding of whether or not stereotypes describe user's preferences. Preferences here indicate both positive and negative biases. For example, the fact that the Agresti and Coull (1998) test shows that it is statistically significant that a given user has *not* consumed a number of items falling into a particular stereotype (for example, low-budget movie items) still indicates valuable information for an RS.

Table 6 displays the results of the application of the Agresti and Coull (1998) test with 99% confidence. The table is ordered by the proxy of how significant the feature is for the user's population and can be read as follows: for the feature genre, only 12.3% of users display no significative positive or negative preferences toward at least one of the genre stereotypes (i.e. 12.3% of the users have review sets for which no genre is positively or negatively preferred). Of the users displaying at least one positive preference (87.7% of the population), only 26% display a large positive preference (LPP) toward at least one stereotype, and 30% display a large negative preference (LNP) toward at least one stereotype (the two may not be mutually exclusive). The discovered stereotypes are indeed capable of describing positive and negative preference traits for over 70% of the users. The table also shows how, across feature types, the typical number of stereotypes with the ability of engaging user preferences are between 1 and 3. Some users may be more responsive to the stereotype of certain features than other users; for example, some users may be more engaged with the cast popularity of the movie and its budget, while other users with the movie genre.

When the same tests were performed on the Amazon dataset (not shown), we found that the number of users indifferent to all of the stereotypes of a given feature fell within a similar range as that identified for the MovieLens/IMDb dataset, with the most descriptive stereotyped features leaving just 16% of the population indifferent. The stereotypes of the less descriptive features of users preferences left 58% of the population of users indifferent. Furthermore, in the Amazon dataset, we found

Table 6 Summary for all MovieLens/IMDb features for the explanatory power of stereotypes via the Agresti-Coull test with a confidence level of 99%

Feature	% Users indifferent	% Users with LPP	% Users with LNP	Avg of significant stereotypes	Total stereo-types
Log (vote count)	1.1	97	95	3.0	4
Popularity	3.0	91	92	2.6	4
Log (budget)	4.3	85	93	2.0	3
Log (revenue)	4.4	85	93	2.2	3
Genre	12.3	26	30	3.3	8
Vote avg.	12.7	66	65	2.0	4
Keywords	18.0	53	46	3.0	25
Release year	22.1	49	57	2.1	4
Director popularity	23.4	19	0	2.03	5
Cast popularity	24.7	43	47	2.2	4
Runtime	25.7	53	28	2.0	4
Prod. comp. popularity	26.7	22	43	1.7	4
Release time of the year	65.4	13	2	1.3	5
Cast gender bias	70.2	6	3	1.3	4
Country	75.6	5	5	2.0	2

that up to 70% of the users exhibited a strong positive or negative preference for at least one or more stereotypes.

This section concludes with two observations; the first is that, in the experiment under examination, the stereotypes obtained via the proposed methodology have been shown to be stable on out-of-sample data in both the hard and soft tests (i.e. they accurately describe the item population metadata with a reduced set of dimensions, and they are capable of describing users' positive and negative preferences). These tests are key to confirming that, for the problem at hand, one can proceed to embed the stereotypes as the base coordinates in an RS. The second observation is that the suite of test presented as a way to aid the preliminary evaluation of the stereotypes can be considered among the contributions of this research.

6 Stereotype-based recommendation performance

Traditionally, RS research has focused on predicting the rating that users would give to each item. For the rating-prediction task, evaluation is usually based on error metrics such as root mean squared error (RMSE) or mean absolute error (MAE). Rating prediction continues to be an important performance evaluation aspect of RS and has been adopted by recent research (Mauro and Ardissono 2019; Wibowo et al. 2018). Nevertheless, researchers have acknowledged that accuracy of rating predictions alone is not sufficient for identifying a quality RS, and the ongoing trend is to evaluate ranked lists of items, presenting users with ranked lists of items and

evaluating which RS-derived lists possess qualities such as being relevant and novel to the user.

Our presentation of results includes a wide range of metrics, from ratings predictions and metrics describing the quality of ranked lists to metrics attempting to capture the serendipity of recommendations. Specifically, we first predict and recommend which items a user is likely to consume under cold-start scenarios. For this task, we benchmark the stereotype-based model against the same RS model using the primitive metadata features and then present the results in Sect. 6.1.1. In Sect. 6.1.2, we focus on the rating predictions. Using different machine learning approaches with increasing complexity and stereotypes as features, we benchmark several RSs against the same models using the primitive metadata features. Our main goal is to demonstrate that enriching the user and item metadata via stereotypes, as described in the research, leads to better cold-start performance regardless of the RS algorithm chosen.

In Sect. 7, we benchmark the RS-driven by stereotypes against SVD-based RS with metadata. The latter is also known in the literature as a factorisation machine. Singular value decomposition with metadata remains a competitive and popular approach, especially in cold-start problems as well as when defining top-N recommendations (Frolov and Oseledets 2019; Hadash et al. 2018; Zhang et al. 2017). The recommendations of these two approaches are studied under a variety of metrics that also comprise ranking quality, including hit rate (HR), mean reciprocal ranking (MRR), mean average precision (MAP), normalised discounted cumulative gain (nDCG) and half-life utility (HLU). We also introduce an ad-hoc metric for complex categorical features that represent the variety and serendipity of recommendations.

6.1 Preliminary experimental evaluation

The preliminary evaluation aims at demonstrating the benefits of using stereotypes over standard metadata and identifying the RS that is analysed in detail in Sect. 7.

In evaluating stereotypes recommendations, in the sections that follow, for each model, two experiments are performed:

- New User Experiment. The set of users is first filtered to exclude users that have less than 10 reviews. The remaining set of users (5544 in the MovieLens/IMDb case, 82622 in the Amazon case) are split into training and test sets. Each pair of training and test sets are in the ratio of 70% to 30%. 6 such sub-experiments are performed. Each experiment is set up selecting users for the test data randomly with the only constraint that each user must be in the test set of at least one experiment and it cannot be in the test set of more than two distinct experiments. For each experiment the system first generates stereotypes based on all the items, and all the users in the training dataset. The system fits the rating provided by the users in the training dataset over the model's features (original metadata, stereotypes). For each user in the test dataset the model generates recommendations and ranked recommendation lists as if the user had not rated any item, and the resulting recommendations are compared for each of those test set users to the

ones effectively expressed. The resulting metrics (consumption/non consumption, rating value, rank) are reported in the new user experiments, according to the metric evaluated.

- New Item Experiment. The new item experiment follows a similar pattern. First the item group is filtered to exclude items that have received less than 10 reviews. The remaining set of items (3395 in the MovieLens/IMDb case, 139261 in the Amazon case) is used to generate training and test splits with 6 sub-experiments in the same manner as for the new user experiment. If an item falls in the test dataset, all users that have rated those items have their rating removed in the training process. For each item in the test dataset (whose reviews had been blanked away from all users), the RS generates recommendations for every user.

The results reported are the average over six experiments in which the dataset was split in training and test in a 70–30 ratio, as previously described. Performance can be evaluated using the average and the distributions around the averages across all runs. In the item's consumption case, the variable predicted is a binary variable expressing whether or not a user consumed an item, leading to a 'user-to-item consumption matrix'. In the rating case, the variable predicted is the rating, which is reported on a 1–5 scale for both datasets.

6.1.1 Cold-start assessment of item consumption

When performing predictions for an item's consumption, one is not only interested in the class label (0,1), but also in obtaining an estimate of the probability that a user will consume an item. For such an experiment, a simple neural network with a single layer of neurons and a softmax layer to rescale the output to a probability density was chosen as a classifier. Subsequently, this classifier will be referred to as the neural network with softmax recommender (NNSR).

Baseline Model and Stereotype-Based Models

Given the different nature of stereotypes for complex categorical features and numerical features, in this preliminary phase, the recommendations are independently evaluated for the two types of stereotypes. This demonstrates that performance improvements are intrinsic to the stereotypes approach and not due to a particular type of feature. For this evaluation, three models are examined:

- A baseline model ($NNSR_b$) which uses all features available in the item and user metadata as they are in the original data.
- A complex categorical stereotype model ($NNSR_c$) which uses the stereotypes for the complex categorical features and reverts to the standard features for the remaining features.
- A numerical stereotype model ($NNSR_n$) which uses the stereotypes for the numerical features and reverts to the standard features for the categorical features.

In this section, the baseline model serves as the reference model in terms of performance.

Recommendation Results

Table 7 shows the metrics derived from the confusion matrices for the new user and new item experiments in the MovieLens/IMDb data. For evaluating the performance of model, the area under the curve (AUC) for both the receiver operating characteristic (ROC) and the precision–recall (PRC) curves are reported. When the predicted classes are very unbalanced, as is the case in scenarios where users have consumed only few items compared to the total number of items (e.g. unbalanced presence of 0s over 1s in the data), predicting rare '1' events (true positives) becomes more important than predicting '0' (true negatives). In such cases, as prescribed by Saito and Rehmsmeier (2015), the AUC for the PRC may be more indicative of performance of model, although the latter is more difficult to interpret. Despite the use of features with lesser dimensions than the original metadata, it can be observed that the stereotype-based system provides an improvement in predicting items consumption, especially for the true positive metric and the PRC AUC. For example, by examining Table 7, the reader can notice that while the standard metrics of accuracy and precision might not reveal a substantial difference across systems, the true positive rate is improved, by using numerical stereotypes, by 4.5% compared to the base system (from 73.32% to 76.6%), and a similar improvement is recorded for both new user and new item experiments. In the PRC AUC the improvement driven by stereotypes is of the order of 5%, in the case of new item experiment, and it seems to be driven independently by both numerical and complex categorical features stereotypes. For the new user case the improvement in PRC AUC is lower and of the order of 2% (moving the metric from 41.1% to 41.9%), but in a similar fashion as the new item experiment it appears to be driven by both numerical and complex categorical features.

This first analysis demonstrates that the improvements, which might not be noticeable in general precision metrics, are indeed present in the metrics that matter most in predicting consumption in the case where the consumed class is rare compared to the catalogue, hinting that the dimension-reduction process intrinsically embeds elements of increased predictability during cold-start. This can be viewed as supporting evidence toward the use of stereotypes in cold-start phases.

Table 7 Classification-prediction metrics derived from the confusion matrices, including the area under the curve (AUC) for both the receiver operating characteristic (ROC) and the precision–recall curve (PRC) for the new-user and new-item experiments in the MovieLens/IMDb. (T.P. refers to true positive, and F.P. refers to false positive)

	New user experiment			New item experiment		
	$NNSR_b$ (%)	$NNSR_c$(%)	$NNSR_n$(%)	$NNSR_b$(%)	$NNSR_c$(%)	$NNSR_n$(%)
Accuracy	71.49	71.68	71.30	70.8	71.3	71.4
Precision	30.27	30.49	30.71	29.6	30.2	30.8
T.P. Rate	73.32	73.55	76.6	73.2	73.8	76.6
F. P Rate	28.82	28.64	29.6	29.6	29.1	29.5
ROC AUC	79.6	80.0	79.8	79.3	80.1	80.9
PRC AUC	41.1	41.8	41.9	40.2	42.2	41.9

It is widely recognised that one way to evaluate an RS is by examining the ranked lists of recommendations that are produced by truncating the list at N items, typically referred to as 'top-N'. By construction, the NNSR systems can predict the probability of each item's consumption by a given user. Later in the paper ranked lists will be examined via modern metrics of how useful the items recommended to the users are. In this context, we focus momentarily on the ability of stereotypes to better predict which items may be selected by which users.

For every new user, the top-N items ranked by probability of consumption are selected and crosschecked for actual consumption of those new users, and the precision metrics are computed for the various NNSR systems. Table 8 shows the sample statistics for precision. The table also provides the *p*-values obtained comparing the mean of the two samples. The null hypothesis is that the average precision obtained with the two methods are equal. So, rejecting the null hypothesis is equivalent to say that there is enough statistical significance that the two means are not the same. And when the mean of the stereotypes is higher than that of the base system it implies that the model with stereotypes performs better. For example, in the new user experiment for the top 100 items, the base model scores an average precision of 42.44%, the stereotype-based model $NNSR_c$ 43.29% (an improvement of 2% over base with 96% confidence) and the $NNSR_n$ 44.55% (an improvement of 4.9% over the base with over 99% confidence). While in the new item experiment the use of stereotypes leads to an improved ability to predict which new items a known user would end up selecting across all lists examined compared to the base, for the new user experiment these results point to a statistically significant benefit for larger lists.

In the smallest of the top N lists, albeit the average precision is marginally higher using stereotypes, we cannot statistically conclude that they improve the predictions under exam. We can however conclude that in no case stereotypes-driven predictions perform worse than the base model predictions.

Table 8 New user and new item top-N recommendations for MovieLens/IMDb including performance metrics of stereotype models $NNSR_c$ and $NNSR_n$ versus the baseline model $NNSR_b$ along with the performance increase and *p*-value of the test on the significance of the performance increase due to stereotypes. Bold is meant to highlight large *p*-value for which we have no statistical significance

		New user experiment			New item experiment		
		Top50	Top100	Top150	Top50	Top100	Top150
$NNSR_b$	Avg Precision	25.70%	42.44%	56.53%	22.87%	41.93%	55.12%
	Precision Std	15.9%	14.2%	13.2%	17.8%	14.8%	13.3%
$NNSR_c$	Avg Precision	26.16%	43.29%	57.13%	25.49%	43.08%	56.96%
	Precision Std	16.0%	14.3%	13.3%	16.0%	14.1%	13.1%
	Precision %	1.79%	2.03%	1.07%	11.4%	2.74%	3.34%
	p-value	**0.36**	0.04	**0.21**	<0.01	0.09	0.04
$NNSR_n$	Avg Precision	25.82%	44.55%	58.33%	24.72%	44.74%	58.50%
	Precision Std	16.8%	14.9%	13.6%	16.8%	14.9%	13.4%
	Precision %	0.47%	4.85%	3.18%	8.09%	6.73%	6.13%
	p-value	**0.78**	<0.01	0.01	0.04	<0.01	<0.01

The prediction of item consumption is strongly affected by the imbalance in the class predicted (i.e. consumed) versus the majority of the observations (i.e. not consumed). In extremely unbalanced datasets, in order to minimise error, classifiers have a negative incentive in predicting the rare class, and as a result lean to always predict 'not consumed'. To deal with very unbalanced datasets (e.g. datasets where the minority class has a frequency below a few per cent, but above 0.1%) specialised techniques exist. The present research has resorted to using the Synthetic Minority Over-sampling Technique (Chawla et al. 2002), for the MovieLens/IMDb dataset. While the MovieLens/IMDb dataset has a highly unbalanced class of rated versus not rated, with the average user having rated about 1% of the catalogue, in the Amazon dataset the imbalance is even more extreme, with the typical user having just 1.7 reviews on average. The average user-to-item rating is in the region of sub-0.001% of the catalogue. This level of imbalance is also outside the scope of techniques like the one referenced. We therefore do not investigate the item-consumption experiment for the Amazon dataset.

6.1.2 Cold-start assessment of item ratings

Having demonstrated in Sect. 6.1.1 that the use of stereotypes improves the cold-start predictions for item consumption and having also demonstrated that both numerical and complex categorical stereotypes provide independent sources of improvement, this section focuses on predicting rating with the full range of stereotyped features.

Given the nature of the rating variable, generally represented as a discontinuous number with R possible values, two options are available: using a classification approach in R buckets or predicting the normalised, scaled dependent variable using a regression-like algorithm. The literature includes examples of both methods (Latif and Afzal 2016) for a classification example and (Spiegel et al. 2009) for a regression example. Section 6.1.1 demonstrates how stereotypes can be used in a classification approach. In this part of the evaluation, the potential benefit of using stereotypes versus original metadata is investigated using regression-like approaches.

Generally, user-to-item ratings exhibit biases (Bell and Koren 2007). Several techniques have been proposed in the literature to account for such biases, see, for example, (Bell and Koren 2007; Spiegel et al. 2009). In this study, ratings are normalised per user by converting them to standard scores.

For each of the experiments—new user and new item—several machine learning algorithms capable of predicting a numerical target variable are tested, where the only difference between the setups evaluated consists of how the predictor features are treated. In the baseline model, all features are treated as they are in the original dataset. In the stereotype model, all features are treated via their stereotype representation. The algorithms tested and presented for this evaluation cover the full spectrum of algorithm complexity:

- A naive approach where a system is metadata unaware and involves either (a) predicting a rating that equals the average rating for the item considered (new user) with no regard to the specific user or (b) predicting a rating that equals the

average rating of the user considered, with no regard to the specific item (new item).
- A simple linear regression approach where the regression model is a standard least-square linear regressor.
- A neural network regression approach which is based on a single-layer neural network with a softmax layer.
- An XGBoost-driven regression where XGBoost stands for eXtreme Gradient Boosting and it was developed by Chen (2016) as an implementation of gradient-boosted decision tree classifiers and regressors (Friedman 2001, 2002).

Rating Predictions and Recommendation Results

Tables 9 and 10 show the key performance metrics obtained for the new user and new item experiments, respectively, for the MovieLens/IMDb and Amazon datasets. The tables display prediction–accuracy metrics for the naive system and for the RS derived via the three different regression approaches, as well as the different treatments of the metadata used by each regressor (original metadata indicated as base model versus stereotype model). As expected, using an RS capable of extracting rating relationships from metadata reduces the error in cold-start rating predictions compared to the naive approach. Furthermore, increasing the regressor's ability to use the metadata improves the rating prediction (i.e. moving from a simple linear model to a more complex neural-network-driven regression).

Contrary to what intuition might have suggested, reducing the metadata feature space via the use of stereotypes improves rating prediction. For instance, in the new user experiment, the improvement in precision metrics gained from using stereotyped features is greater than the improvement in the same metrics that arises from switching from a simple linear regression model to a more complex one, such as a neural network regression. For example, for the Amazon dataset the improvement obtained when using base coordinates vs stereotypes with a XGBoost solver is such that the RMSE is lowered from 0.612 to 0.593. That corresponds to a 3%

Table 9 Performance metrics for the new user problem

	Dataset	MovieLens/IMDb		Amazon	
	Model	Base	Stereotype	Base	Stereotype
RMSE	Naive	0.963	–	0.76	–
	Linear R.	0.940	0.938	0.616	0.604
	Neural Net R.	0.918	0.906	0.614	0.598
	XGBoost R.	0.913	0.901	0.612	0.593
MAE	Naive	0.772	–	0.48	–
	Linear R.	0.743	0.742	0.530	0.523
	Neural Net R.	0.724	0.712	0.523	0.515
	XGBoost R.	0.721	0.710	0.522	0.512
Time	Linear R.	10.7	9.1	1.534	1.147
	Neural Net R.	69.5	55.6	27.89	18.055
	XGBoost R.	90.5	73.2	12.77	5.472

Table 10 Performance metrics for the new item problem

	Dataset	MovieLens/IMDb		Amazon	
	Model	Base	Stereotype	Base	Stereotype
RMSE	Naive	1.01	–	0.80	–
	Linear R.	0.939	0.934	0.585	0.570
	Neural Net R.	0.928	0.917	0.584	0.563
	XGBoost R.	0.926	0.918	0.585	0.550
MAE	Naive	0.81	–	0.53	–
	Linear R.	0.740	0.736	0.515	0.492
	Neural Net R.	0.735	0.727	0.511	0.485
	XGBoost R.	0.738	0.729	0.510	0.474
Time	Linear R.	10.8	8.6	1.193	1.094
	Neural Net R.	56.8	34.9	26.239	18.383
	XGBoost R.	90.5	71.6	10.799	5.786

improvement. Such improvement is larger than the one deriving from improvements in the recommendation model. For example, in the same experiment replacing the solver from a linear regression to XGBoost only yields an RMSE improvement from 0.616 to 0.612, or 0.6%. Hence using stereotypes, in this particular example, allows an improvement in RMSE that is of a factor of 5 the one that can be obtained improving the sophistication of the recommendation solver. The reader can verify in a similar manner for both Tables 9 and 10 that the pattern suggested in the example persists, albeit with differing strength, across RMSE and MAE. Most importantly, the added precision obtained by using stereotypes does not seem to depend on the regression model used, suggesting that stereotypes offer an extra dimension for improving the quality of recommendations (at least in cold-start phases) that is independent of the rating-prediction algorithm used. This finding is one of the most important findings of the research justifying the use of stereotypes in the RS community as an extra 'direction' for improvement during cold-start phases.

Finally, with the adoption of stereotypes, the complexity reduction in the metadata features can be appreciated via the reduction of CPU time (in seconds) used for training a given regression approach. For the most complex regressors, the stereotyped metadata allows for more than 20% improvement in CPU time. All experiments are run on an Intel Core i7-7700K CPU @ 4.2 GHz with 64.0 GB RAM. In the experimental evaluation, the time of the Amazon experiment was less than that of the MovieLens/IMDb experiment, even though the dataset was larger overall. This difference is due to different numerical setups tested in the two experiments. For the MovieLens/IMDb experiment, the dataset (size and unbalance) was handled with standard storage of matrices. For the Amazon experiment, the vastness and sparsity of the dataset required us to introduce further improvements to the memory storage of the problem's matrices, namely using the compressed rows storage (CRS) as a sparse matrix storage and operation technique to improve the CPU time.

In the Amazon case an increased complexity in the model underlying RS is not met by improvements as for the MovieLens/IMDb case. We attribute this to the

characteristic rating distribution of the Amazon dataset, which are highly skewed toward a high rating. Approximately 60% of the ratings were equal to five (the maximum), and another 20% were equal to four, leaving only another 20% for ratings between one and three. Investigation of this particular aspect is outside the scope of this paper, and it does not affect our conclusion on the value added by stereotypes.

7 Cold-start assessment of recommendations driven by stereotype versus SVD-based RS with metadata

Section 6 shows a broad evaluation of the benefits of introducing stereotypes over the original metadata during cold start, and it allows us to select a stereotype-based RS to benchmark against another well-known methodology. Matrix-factorisation techniques, particularly SVD and SVD++ methods, have gained substantial popularity for addressing problems like sparsity and cold start (Frolov and Oseledets 2019; Hadash et al. 2018; Zhang et al. 2017). Singular value decomposition representation provides an ideal framework for dimensionality reduction. For the sake of completeness, we conduct an investigation and compare such techniques with the stereotype-driven approach. The standard, classic, formulation of the SVD algorithm does not include the user or item metadata. However, it was shown in the previous section that without such information, particularly during cold-start phases, the recommendations provided would be uninformed. Therefore, a fair comparison with SVD and SVD++ should incorporate the user and item metadata in the factorisation procedure.

In Sect. 7.1, the concepts behind the incorporation of metadata into SVD and SVD++ are revised. In Sect. 7.2, an in-depth analysis of the recommendation quality of the two approaches (stereotypes and factorisation methods) is conducted; such analysis is not limited to recommendation accuracy, but it attempts to investigate other aspects and desirable properties of recommendations, such as utility and novelty.

7.1 SVD/SVD++-based RS (with Metadata)

The intuition behind SVD and general matrix factorisation methods is that there should exist a latent space (P_f) of dimensionality f that determines how a user rates an item. User-item interactions (i.e. ratings) are modelled as inner products in that space. For example, the user u's rating of item i, which is denoted by r_{ui}, can be represented as the inner product of two arrays of length f leading to the estimate:

$$r_{ui} = q_i^T * p_u \tag{2}$$

where each item i is associated with a vector $q_i \in P_f$ and each user u is associated with a vector $p_u \in P_f$. To learn the factor vectors p_u and q_i (and therefore the latent space representations), the system minimises the regularised squared error on the set of known ratings. It is important to stress here that these are latent characteristics and do not necessarily correspond to the user and item metadata.

The SVD construct is usually expressed in terms of a simple scalar product which does not by construction allows the use of users or items metadata. Two enhancements have been proposed to such construct to improve its performance in the cold-start phase: (1) introducing user-and item-specific biases in the ratings, (2) adding user and item metadata.

Enhancements 1 and 2 lead to the decomposition of the rating of user u of item i, which is denoted by r_{ui}, as illustrated in Eq. 3:

$$r_{ui} = \mu + b_u + b_i + (q_i + \sum_{a \in A(i)} y_a)^T * (p_u + \sum_{b \in B(u)} y_b) \tag{3}$$

where the terms $\mu + b_u + b_i$ represent overall mean, user bias and item bias, respectively. The vectors q_i and p_u represent the standard SVD terms not discussed here for brevity. To include the item's metadata, a term is added to the right of q_i. The metadata is encoded to a 1 to enne encoding, giving item i a set of attributes $A(i)$ (e.g. genres, movie budget and revenue) via the vector y_a. Similarly, a term for the user metadata representation via a set of attributes $B(u)$ is added to the right of p_u.

In the seminal work on factorisation machines (FM) Rendle (2010) demonstrates that a strong similarity exists between SVD++ and FM, with an extra term provided by FM for modelling extra users and movies interactions. Inspection of such extra term, from section V.B of Rendle (2010) and Eq. 3—which we refer to as SVD with metadata—reveals such a strong similarity that it is justified to refer to our SVD with metadata formulation as a special version of a FM.

A third enhancement, independent of the two previously discussed, led to the technique SVD++. As highlighted by Koren (2008) and Koren et al. (2009), RSs can use implicit feedback to gain insight into user preferences. This step rests in the assumption that items that a user has not rated have implicit feedback content: if a user has not rated an item, the implicit feedback assumption postulates a negative preference.

7.2 SVD with metadata versus stereotypes recommendations

In this section, the results obtained via SVD and SVD++-based RS (with metadata) are compared to the stereotype-based system driven by the XGBoost regression under the lenses of classical accuracy metrics and modern measures evaluating the usefulness of ranked recommendation lists. The results presented show further corroborating evidence that stereotypes-aided recommendations are superior to the other state-of-the-art systems during extreme cold-start phases. It is important to note that in the new user experiment, for the 30% of users in the test set, we assumed that no ratings were available, hence the SVD++ technique is not applicable. It should be noted that in the current evaluation, the implicit feedback is incorporated in the MovieLens/IMDb experiment for the factorisation-driven RS, but it is not used for the stereotype-driven RS. Furthermore, given the sparsity of the Amazon dataset, where the average user has rated 1.7 items on a catalogue with over one million items, we feel that the use of implicit feedback is not justified as users are

simply unable to consume all the items they would like to consume given the time and given the financial means required.

Table 11 shows basic prediction accuracy metrics for both the new user and new item experiments. The stereotype-based model outperforms all of the SVD-driven methods in both RMSE and MAE with the only exclusion being the SVD++ with metadata in the MovieLens/IMDb case (i.e. the addition of the implicit feedback information in the new item problem). As noted previously, the stereotype-driven models do not make use of implicit feedback information.

We move next to the study of performance in ranked lists recommendations. The number of reviews is sparse for MovieLens/IMDb, with the average user having reviewed around 1% of the catalogue, and extremely sparse for Amazon, with the average user having reviewed less than 0.01% of the catalogue. Therefore, it is important to note that the disparity between the number of reviews available for the average user compared to the number of items in the catalogue naturally leads to rank-accuracy metrics that are not comparable across datasets. In what follows we show how stereotypes-driven recommendations have a superior performance, with an overall high statistical confidence, according to the most widespread metrics for evaluation of ranked list; for each metric we present the rationale of the metric and discuss the results. For the case of 'serendipity', where there is little agreement in the RS community on how to quantitatively frame such concept, we introduce a novel definition of serendipity (or list variety) that fits well the scope of item complex categorical features—which are to be seen often as the most descriptive features of items metadata. Also according to our definition of serendipity we obtain further evidence corroborating the adoption of stereotypes in RS addressing cold starts.

Hit Rate (HR). The simplest way to evaluate top-N recommendation is HR, which measures the proportion of successfully recommended items in top-N recommendations. The hit rate is evaluated at different N (10, 20 and 30), and the results are shown in Tables 12 and 13 for the new user and new item experiments, respectively.

For both the new user and new item experiments and both the MovieLens/IMDb and Amazon datasets, the tables attain a statistically significant improvement over

Table 11 New user and new item cold-start comparisons between the recommendation model (with stereotypes) and the SVD-driven models (with and without metadata)

	Dataset	MovieLens/IMDb		Amazon	
	Model	RMSE	MAE	RMSE	MAE
New user	Stereotype	0.901	0.710	0.593	0.512
	SVD without metadata	0.961	0.768	0.733	0.542
	SVD with metadata	0.924	0.736	0.613	0.534
New item	Stereotype	0.918	0.729	0.550	0.474
	SVD without metadata	1.059	0.861	0.773	0.529
	SVD with metadata	0.932	0.748	0.588	0.511
	SVD++ with metadata	0.905	0.727	–	–

Table 12 Ranking of accuracy metrics for the new user problem as indicated by the model with stereotypes and the SVD with metadata

	n	MovieLens/IMDb Dataset			Amazon Dataset		
		Stereotype	SVD w. metadata	*p*-value	Stereotype	SVD w. metadata	*p*-value
HR	10	34%	26%	$\ll$0.01	3.16%	1.50%	$\ll$0.01
	20	29%	22%	$\ll$0.01	3.13%	1.47%	$\ll$0.01
	30	26%	21%	$\ll$0.01	2.96%	1.44%	$\ll$0.01
MRR	-	66%	66%	**0.86**	4.97%	2.80%	$\ll$0.01
MAP	10	22%	15%	$\ll$0.01	2.89%	2.25%	<0.05
	20	17%	11%	$\ll$0.01	2.99%	2.55%	<0.05
	30	14%	9%	$\ll$0.01	3.22%	2.66%	<0.05
nDCG	10	61%	57%	<0.05	5.80%	3.40%	<0.01
	20	60%	55%	$\ll$0.01	7.50%	4.20%	<0.01
	30	60%	54%	$\ll$0.01	8.70%	4.80%	<0.01
HLU	10	48	44	<0.05	2.52	1.54	<0.05
	20	42	39	<0.05	2.41	1.49	<0.05
	30	40	35	<0.05	2.29	1.44	<0.05

For each metric, the *p*-value of testing the null hypothesis 'the difference in the metric value between the stereotype and the SVD with metadata is not significant'. Low *p*-value indicates the need to reject the null. Bold is meant to highlight large *p*-value for which we have no statistical significance

Table 13 Ranking of accuracy metrics for the new item problem, including the model with stereotypes and SVD (SVD++) with metadata

	n	MovieLens/IMDb Dataset			Amazon Dataset		
		Stereotype	SVD++ w. metadata	*p*-value	Stereotype	SVD w. metadata	*p*-value
HR	10	23%	20%	$\ll$0.01	2.53%	1.40%	$\ll$0.01
	20	21%	14%	$\ll$0.01	2.54%	1.45%	<0.01
	30	21%	11%	$\ll$0.01	2.66%	1.61%	<0.01
MRR	-	51%	55%	$\ll$0.01	3.66%	2.28%	$\ll$0.01
MAP	10	12%	9%	$\ll$0.01	2.52%	2.08%	<0.01
	20	10%	6%	$\ll$0.01	2.71%	2.13%	<0.01
	30	9%	4%	$\ll$0.01	2.77%	2.66%	<0.05
nDCG	10	49%	52%	<0.05	3.31%	2.90%	<0.1
	20	49%	51%	<0.05	4.47%	4.10%	<0.1
	30	49%	51%	**0.13**	6.00%	5.60%	<0.1
HLU	10	36	32	<0.01	2.33	1.14	<0.01
	20	29	28	**0.11**	2.30	1.13	<0.01
	30	26	26	**0.58**	2.29	1.12	<0.01

For each metric, the table reports the *p*-value of testing the null hypothesis 'the difference in the values between the stereotype and the SVD with metadata is not significant'. A low *p*-value indicates the need to reject the null. Bold is meant to highlight large *p*-value for which we have no statistical significance

the baseline with respect to HR. Two noteworthy observations are illustrated. The first concerns the new user case for which as N increases the HR decreases; this is due to the number of hits identified in the lists being higher in the ranks. Therefore, as the list grows, the denominator of the HR definition increases faster than the numerator, indicating that the discovered hits tend to fall in the high-ranking portion of the list. The second observation clarifies the fact that, for the Amazon dataset, despite the slightly higher RMSE of the new item case compared with the new user case, the latter displays a higher HR. This fact can be explained by the mechanics of the experiment: while in the new user case every user is scored against all of the items (whether items were rated or not), in the new item case, only 30% of the items are retained in the new item test set. The reduced set of items, many of which are not reviewed, is responsible for reducing the likelihood that a recommended item will actually be reviewed by any of the users.

One extra observation concerns the difference in the values of the metrics obtained in the two datasets, where the Amazon values are generally one order of magnitude lower than the same metrics for the movie dataset. The explanation for this rests on the much larger Amazon catalogue (two orders of magnitude larger than the movie catalogue and two orders of magnitude more unbalanced), under such a condition a fixed length list is expected to produce lower statistical rank accuracy metrics.

Mean Reciprocal Rank and Mean Average Precision. Mean reciprocal rank (MRR) is another measure for evaluating systems that return a ranked list (Baeza-Yates and Ribeiro-Neto 2011), which accounts for the rank of the position of the *first* correctly identified recommendation. While MRR can be thought of as a score for evaluating only the top hit, the mean average precision (MAP) provides a more suitable measure for ranking the quality of a list rather than just the highest-ranking hit. The MAP metric provides a single summary of the user's ranking preferences as described by Baeza-Yates and Ribeiro-Neto (2011). The terminology used is 'MAP @N' to describe the relevance of the list of the N recommended items.

The results for the MRR and MAP in the cold-start experiments are shown in Tables 12 and 13. For the new user case, if only the quality of the top hit is examined (via MRR), in the MovieLens/IMDb case there is no statistically significant difference between the stereotype and the SVD with-metadata RS. For the Amazon dataset, the MRR confirms the improvement brought by stereotypes in the new user case. In the new item case, the RS based on SVD++ with metadata displays a higher-quality top hit in the case of MovieLens/IMDb. This suggests that, for this particular dataset, the use of implicit feedback indeed provides valuable information for improving the quality of the top recommendation. Once focus is extended past the single top recommendation to a basket of recommendations (HR and MAP), then the recommendations provided by the stereotype-based approach constitute a statistically significant improvement over the SVD with metadata techniques.

Normalised Discounted Cumulative Gain. The nDCG is a single-number measure of the effectiveness of a ranking algorithm that allows non-binary judgments of relevance. nDCG uses graded relevance, which is accumulated starting at the top of the ranking and may be reduced, or discounted, at lower ranks (Jarvelin and Kekalainen 2002).

The results obtained for the two cold-start scenarios comparing the stereotype-based models and the models based on matrix factorisation with metadata are reported in Tables 12 and 13. nDCG confirms the results obtained with the other ranking metrics analysed by measuring the average usefulness of our recommendations to the users. The model with stereotypes outperforms the SVD-with-metadata-based RS with a confidence level of over 95% across all the nDCG tests for the new user case and both datasets. For the new item case, the stereotype-based system was slightly less performant in the MovieLens/IMDb experiment and slightly more performant than the SVD with metadata in the Amazon dataset. As N grows, the statistical confidence on the nDCG of the SVD++ outperforming that of stereotypes wanes. Our interpretation of this result in the MovieLens/IMDb data is that it arises due to the use of implicit feedback. Giving a score of zero to items that no users in the training sample watched prevents these items from being recommended.

Half-life Utility Metric. The HLU was introduced by Breese et al. (1998) on the premise that a user presented with a ranked list of results, is unlikely to browse deeply into the list. The HLU evaluation metric postulates that the probability of a user selecting a relevant item drops exponentially as they move further down the list. The metric examines an unbounded recommendation list containing all the items. Given such a list, an item at position j has a probability of $\frac{2^{(j-1)}}{(a-1)}$ of being selected, where a is a half-life parameter.

The utility is defined as the difference between the user's rating for an item and the 'default rating' for an item (Breese et al. 1998), with the default rating generally assumed to be a neutral or slightly negative rating. As represented in Eq. 4, R_u is the expected utility of recommendations given to user u, r_{uj} represents the rating of user u on item j of the ranked list, d is the default rating and a is the half-life factor (or decay factor).

$$R_u = \sum_j \frac{max(r_{uj} - d, 0)}{2^{(j-1)/(a-1)}} \tag{4}$$

Figure 5 displays the HLU for the new user and new item experiments using the MovieLens/IMDb data for the stereotype model with a decay factor a ranging between 3 and 10 and a default rating equal to the median rating in the dataset (three

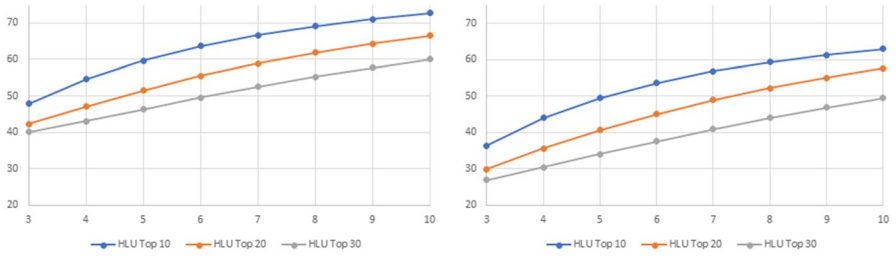

Fig. 5 Half-life utility (R) new user and (L) new item cases as a function of the a decay factor (x-axis) for the MovieLens/IMDb dataset

in the MovieLens/IMDb dataset). The HLU increases with the decay factor (assuming that a user is also interested in items further down the list).

Tables 12 and 13 show the comparison between the HLU value at a decay factor of three using the model with stereotypes versus the SVD-with-metadata-based RS. This metric displays a different picture for the two datasets used in this study. In the MovieLens/IMDb case, for the new user case, we can assert that the model with stereotypes outperforms the factorisation RS with a confidence level over 95% and an improvement of approximately 10%. However, for the new item case, the HLU values are considerably closer and, given the p-values, we cannot assert that the values are statistically different. This behaviour can be ascribed once more to the presence of implicit feedback in the MovieLens/IMDb data. When we focus on the Amazon dataset, the HLU improvements provided by stereotypes are significant with HLU values for the stereotype-based RS as much as double those of the factorisation-driven RS. These values are corroborated by a high statistical significance, with a confidence level greater than 95% in the new user case and 99% in the new item case.

Serendipity. Various definitions have been proposed for this concept in the recommender systems domain. For example, Herlocker et al. (2004) define serendipity as a measure of the extent to which the recommended items are both attractive and surprising to the users. To date, various definitions and evaluation metrics for measuring serendipity have been proposed, and there is no wide consensus on a single definition. For a comprehensive review of the various definitions and challenges, see (Kotkov et al. 2016). Authors often suggest that the definition should adapt to the field of application.

The complex categorical features of a dataset enable the introduction of a proxy for serendipity by measuring how variegated the top-N recommendation lists are concerning such features. We first introduce our proxy for serendipity via an example and then generalise to any complex categorical feature. Based on the MovieLens/IMDb data, it is evident that the genre—a complex categorical feature for such a dataset—plays a key role in the selection process for many users. If a system obtained high prediction accuracy but did so by always recommending the same genre to a given stereotyped user (e.g. a male in his 40s who likes only thriller and action movies), then recommendations would not be variegated despite the high accuracy that one may achieve. The union of labels for complex categorical features in all entries of a top-N list can illustrate the variety within the list. One complication is the fact that some items are categorised with many labels. It can be argued that an item with a well-specified label has more information content (e.g. movie A: Drama) than an item categorised with many labels for different genres (e.g. movie B: drama, war, history, romance, documentary). The weight of a movie in representing a genre should be inversely proportional to the number of labels used (i.e. a movie whose categorisation represents all 24 genres would add a weight of 1/24 to each genre, and a movie with many genres would not significantly represent any single genre). Therefore, for all items in the top-N list, one can compute the sum of the weight contributions of each label (e.g. genre).

The operative definition for a generic complex categorical feature arises naturally from the example discussed: assembling a top-N recommendation list and counting

all labels represented in the list. Each label is weighted according to its contribution to the item representation it is attached to. The sum of all weights for each label provides a spectrum of how many labels and which aggregate weight they are covered in the top-N list. This definition can be applied to any complex categorical feature. One RS will be more novel and serendipitous than another if its top-N list covers more of those labels.

A parameter k is introduced to represent the minimum value of the score required to claim that a certain label was represented. The examination of the abundance of labels in the top-N recommendation list at various k-score thresholds can provide a detailed picture of the variety in the list. Hence, there is an increased possibility of discovering novel and unexpected recommendations. The most representative categorical feature in each dataset (genre in the MovieLens/IMDb dataset and product category in Amazon dataset) are depicted in Fig. 6. The figure shows the number of labels covered in the top-N recommendation list (y-axis) as a function of the growing value of the threshold k (the significance cut off, x-axis) for the stereotype-based models. The results obtained for the model with stereotypes for the new user case for three different N values, namely Top 10, 20 and 30 are shown in the figure. As the list of recommendation increases in size, moving from top 10 to 20 and 30, the variety increases, as expected from a serendipitous RS. Each curve can be seen as a representation of the potential serendipity of the list recommended, for a given k and a given top-N, the higher the number of labels covered the more the potential novelty of the list.

Figure 7 shows the comparison in label diversity (number of labels covered) for the top 10 recommendation lists produced by the model with stereotypes and the SVD-based RS with metadata. If one agrees that a low k value should be in the range of 0.5–1, then the model with stereotypes outperforms the SVD model in this proxy of novelty and serendipity on both datasets and by a substantial amount (in terms of increased novelty provided by a larger number of labels covered). For instance, for the top 10 list, a k value of one means there must be at least one item that fully represents such a label or two items with such a label represented at 50%. The novelty tends to align for higher values of k as expected. With these findings, we can conclude that a stereotype-based recommendation should be more serendipitous than

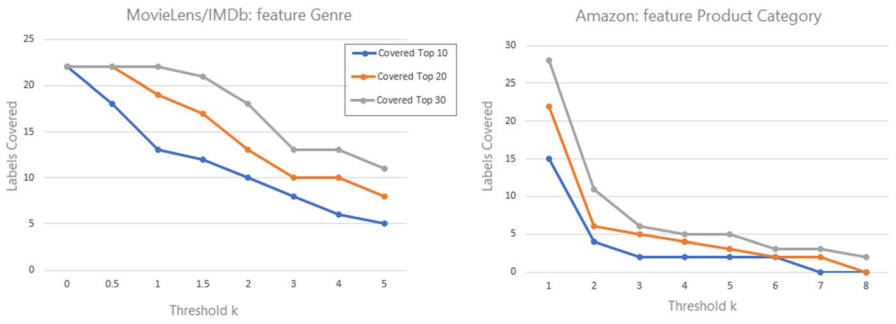

Fig. 6 Diversity (the number of distinct categorical features such as genre and product category) recommended for the model with stereotypes in the MovieLens/IMDb and Amazon datasets

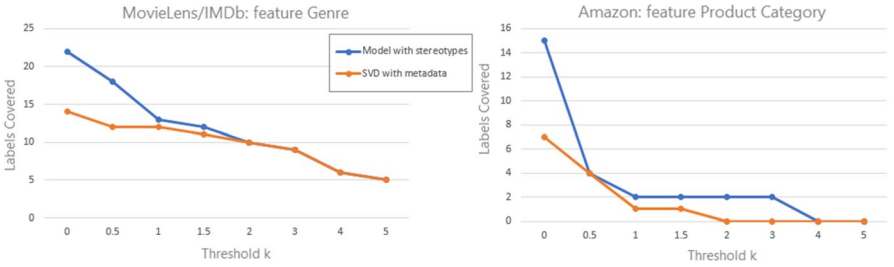

Fig. 7 Comparison of diversity for the model with stereotypes and the SVD-based RS (with metadata) in the MovieLens/IMDb and Amazon datasets

(or at least as equally serendipitous as) the cold-start recommendations derived from factorisation-based SVD with metadata.

Model Complexity and Computation Time. Given a recommendation problem with u users, i items, a single-user dimension of encoded features of size u_f and an item size of i_f, we proceed to estimate the order of magnitude of the models' complexity. For the stereotype model, the clustering of metadata features has a complexity $O(i_f^2 + u_f^2)$, and it results in stereotype coordinates of the order s_u and s_i for users and items, respectively. Hence, the complexity of the learning model applied to the stereotyped coordinates is $O[max(i \cdot s_i, u \cdot s_u)]^3$. The latter is based on the neural network with one-layer solver—the one with the highest complexity among those tested.

The complexity of the SVD with metadata is of $O\{k_1(u \cdot u_f)^2(i \cdot i_f) + k_2(i \cdot i_f)^3\}$ (Gene H. Golub 2013). For example, in a simplified scenario similar to that of the MovieLens/IMDb dataset, with 1000 users (u) and 1000 items (i), with an encoded users features of 20 (u_f) and an encoded items features of 100 (i_f), the stereotype generation process has an initial value with a complexity on the order of $20^2 + 100^2$, or 10400 operations. The stereotype generation process reduces the encoded user features and the item features by a factor between 4 and 5 (this appears to be the case in both of the dataset used), leading, for example, to an s_u of 5 and an s_i of 25. The learning model has complexity on the order of $[max(1000 \cdot 5, 1000 \cdot 25)]^3$, or roughly $1.5 \cdot 10^{13}$ operations. In the same example, the SVD with metadata would require an order of operations (omitting the first user term for simplicity) of $(1000 \cdot 100)^3$ or 10^{15} operations.

8 Conclusion and future work

In this paper, we propose a method for automatically discovering item stereotypes for different data types. We demonstrate that clustering metadata, when performed independently of the user-to-item matrix, provides new metadata features (stereotypes) which allow for improved recommendation in cold-start phases. The contributions of this paper are twofold. First, enriching the user and item metadata via stereotypes leads to enhanced cold-start performance regardless of the machine

learning algorithm chosen to fit the user-to-item preferences. Second, the improvement via stereotypes is greater than the improvement realised when moving from a basic learning algorithm (e.g. linear regression) to a more sophisticated learning algorithm (e.g. neural network). The improvement achieved when using stereotypes is orthogonal compared to that obtained by refining the underlying solver mechanism. This is the key finding of the research and it suggests that our method can be employed in other contexts (e.g. in a deep learning algorithm).

To validate the proposed approach on a movie and retail sales datasets factorisation machines, SVD, SVD++, were used as benchmarks, in addition to baseline models that employed the primitive features. The satisfactory results have demonstrated the effectiveness of employing stereotypes in cold-start phases under widely applied performance metrics. The limitation of the current methodology is intrinsically embedded in the simplification and generalisation that the stereotypes introduce. When a user or an item are better known (i.e. past the extreme cold-start phase), the generalisations are not as predictive as more detailed and specific recommendations driven by the information acquired about a user or item. Stereotype-based recommendations should be phased out from the RS as more personalised information about the new user or item is acquired by the system. However, given the findings on the various labels of complex categorical features when discussing stereotype serendipity, we argue that stereotypes could also be used beyond cold-start phases to add elements of novelty to recommendation lists.

Several lines of future work arise from the present research, including (1) the possibility of embedding implicit feedback into the stereotype-based RS learning process; (2) the application of the methodology to further datasets, and its extension to domain specific features that fall outside the three general types discussed; (3) using the stereotypes as base coordinates in more sophisticated deep learning algorithms for recommendation; and (4) further investigating the application of stereotypes for restoring novelty to recommendation lists when overspecialisation of an RS is detected.

Declarations Not applicable.

Ethics This paper or a similar version is not currently under review by a journal or conference. This paper is void of plagiarism or self-plagiarism as defined by the Committee on Publication Ethics and Springer Guidelines.

References

Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Trans. Know. Data Eng. **17**(6), 734–749 (2005)

Agresti, A., Coull, B.A.: Approximate is better than "exact" for interval estimation of binomial proportions. Am. Stat. **52**(2), 119–126 (1998)

Alahmadi, D.H., Zeng, X.J.: Twitter-based recommender system to address cold-start: A genetic algorithm based trust modelling and probabilistic sentiment analysis. In: 2015 IEEE 27th International Conference on Tools with Artificial Intelligence (ICTAI), pp. 1045–1052. Vietri sul Mare, Italy (2015)

ALRossais, N., Kudenko, D.: Evaluating stereotype and non-stereotype recommender systems. In: Proceedings of the First Workshop on Knowledge-aware and Conversational Recommender Systems co-located with the 12th ACM Conference on Recommender Systems KaRS@ RecSys, Vancouver, Canada, pp 23–28 (2018a)

ALRossais, N., Kudenko, D.: iSynchronizer: A tool for extracting, integration and analysis of movielens and imdb datasets. In: UMAP'18 Adjunct: 26th Conference on User Modeling, Adaptation and Personalization Adjunct, July 8-11, 2018, Singapore, Singapore, p 5 (2018b)

ALRossais, N.A., Kudenko, D.: Generating stereotypes automatically for complex categorical features. In: Proceedings of the Second Workshop on Knowledge-aware and Conversational Recommender Systems co-located with 28th ACM International Conference on Information and Knowledge Management, KaRS@CIKM 2019, Beijing, China, pp 8–14 (2019)

Aranganayagi, S., Thangavel, K.: Improved k-modes for categorical clustering using weighted dissimilarity measure. World Acad. Sci. Eng. Technol. **3**, 813–819 (2009)

Baeza-Yates, R., Ribeiro-Neto, B.: Modern information retrieval: the concepts and technology behind search. Addison-Wesley Professional, New Jersey, USA (2011)

Barkan, O., Koenigstein, N., Yogev, E., Katz, O.: Cb2cf: a neural multiview content-to-collaborative filtering model for completely cold item recommendations. In: Proceedings of the 13th ACM Conference on Recommender Systems, Copenhagen, Denmark, pp 228–236 (2019)

Beel, J., Dinesh, S., Mayr, P., Carevic, Z., Raghvendra, J.: Stereotype and most-popular recommendations in the digital library sowiport. In: Proceedings of the 15th International Symposium of Information Science (ISI) (2017)

Bell, R.M., Koren, Y.: Scalable collaborative filtering with jointly derived neighborhood interpolation weights. In: Proceedings of the 2007 seventh IEEE international conference on data mining, Omaha, NE, USA, vol 7, pp 43–52 (2007)

Billsus, D., Pazzani, M.J.: Learning collaborative information filters. Proceedings of the fifteenth international conference on machine learning, San Francisco, CA, United States Vol. 98, pp. 46–54 (1998)

Brajnik, G., Tasso, C.: A shell for developing non-monotonic user modeling systems. Int. J. Hum. Comput. Stud. **40**(1), 31–62 (1994)

Braunhofer, M., Elahi, M., Ricci, F.: User personality and the new user problem in a context-aware point of interest recommender system. In: Information and Communication Technologies in Tourism 2015, pp. 537–549. Lugano, Switzerland (2015)

Breese, J.S., Heckerman, D., Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In: Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence, Morgan Kaufmann, Burlington, pp 43–52 (1998)

Burke, R.: Hybrid recommender systems: Survey and experiments. User Model. User-Adap. Inter. **12**(4), 331–370 (2002)

Cao, F., Liang, J., Bai, L.: A new initialization method for categorical data clustering. Expert Syst. Appl. **36**(7), 10223–10228 (2009)

Cao, F., Huang, J.Z., Liang, J., Zhao, X., Meng, Y., Feng, K., Qian, Y.: An algorithm for clustering categorical data with set-valued features. IEEE Trans. Neural Netw. Learn. Syst. 1–14 (2017)

Cella, L., Cereda, S., Quadrana, M., Cremonesi, P.: Deriving item features relevance from past user interactions. In: Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization, Bratislava, Slovakia, pp 275–279 (2017)

Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: synthetic minority over-sampling technique. J. Artif. Intell. Res. **16**, 321–357 (2002)

Chen, T.: Scalable and flexible gradient boosting. https://xgboost.ai/, Accessed 2019-11-01 (2016)

Chen, Y.C.: A tutorial on kernel density estimation and recent advances. Biostat. Epidemiol. **1**(1), 161–187 (2017)
Cohen, D., Aharon, M., Koren, Y., Somekh, O., Nissim, R.: Expediting exploration by attribute-to-feature mapping for cold-start recommendations. In: Proceedings of the Eleventh ACM Conference on Recommender Systems, Como, Italy, pp 184–192 (2017)
Yashar, Deldjoo M.E.M.Q., Cremonesi, P.: Using visual features based on mpeg-7 and deep learning for movie recommendation. Int. J. Multimed. Inform. Ret. **7**(4), 207–219 (2018)
Deldjoo, Y., Dacrema, M.F., Constantin, M.G., Eghbal-Zadeh, H., Cereda, S., Schedl, M., Ionescu, B., Cremonesi, P.: Movie genome: alleviating new item cold start in movie recommendation. User Model. User-Adap. Inter. **29**(2), 291–343 (2019)
Deshpande, M., Karypis, G.: Item-based top-n recommendation algorithms. ACM Trans. Inform. Syst. (TOIS) **22**(1), 143–177 (2004)
Du, X., Liu, H., Jing, L.: Additive co-clustering with social influence for recommendation. In: Proceedings of the Eleventh ACM Conference on Recommender Systems, Como, Italy, pp. 193–200 (2017)
Edelsbrunner, H., Harer, J.: Computational Topology: An Introduction. American Mathematical Society, Providence (2010)
Elahi, M., Ricci, F., Rubens, N.: Active learning in collaborative filtering recommender systems. In: International Conference on Electronic Commerce and Web Technologies, pp. 113–124. Munich, Germany (2014)
Elahi, M., Braunhofer, M., Gurbanov, T., Ricci, F.: Collaborative Recommendations: Algorithms, Practical Challenges And Applications, World Scientific Publishing, Singapore, chap User Preference Elicitation, Rating Sparsity and Cold Start, pp 253–294 (2018)
Enrich, M., Braunhofer, M., Ricci, F.: Cold-start management with cross-domain collaborative filtering and tags. In: International Conference on Electronic Commerce and Web Technologies, pp. 101–112. Czech Republic, Prague (2013)
Eskandanian, F., Mobasher, B., Burke, R.: A clustering approach for personalizing diversity in collaborative recommender systems. In: Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization, Bratislava, Slovakia, pp 280–284 (2017)
Eskandanian, F., Sonboli, N., Mobasher, B.: Power of the few: Analyzing the impact of influential users in collaborative recommender systems. In: Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization, Larnaca, Cyprus, pp 225–233 (2019)
Felício, C.Z., Paixao, K.V., Barcelos, C.A., Preux, P.: Preference-like score to cope with cold-start user in recommender systems. In: 2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI), pp. 62–69. CA, USA, San Jose (2016)
Felício, C.Z., Paixão, K.V., Barcelos, C.A., Preux, P.: A multi-armed bandit model selection for cold-start user recommendation. In: Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization, Bratislava, Slovakia, pp 32–40 (2017)
Fernández-Tobías, I., Braunhofer, M., Elahi, M., Ricci, F., Cantador, I.: Alleviating the new user problem in collaborative filtering by exploiting personality information. User Model. User-Adap. Inter. **26**(2–3), 221–255 (2016)
Fernández-Tobías, I., Cantador, I., Tomeo, P., Anelli, V.W., Di Noia, T.: Addressing the user cold start with cross-domain collaborative filtering: exploiting item metadata in matrix factorization. User Model. User-Adap. Inter. **29**(2), 443–486 (2019)
Friedman, J., Hastie, T., Tibshirani, R.: The elements of statistical learning, vol. 1. Springer, Heidelberg (2001)
Friedman, J.H.: Greedy function approximation: a gradient boosting machine. Ann. Stat. 1189–1232 (2001)
Friedman, J.H.: Stochastic gradient boosting. Comput. Stat. Data Anal. **38**(4), 367–378 (2002)
Frolov, E., Oseledets, I.: Hybridsvd: when collaborative information is not enough. In: Proceedings of the 13th ACM Conference on Recommender Systems, Copenhagen, Denmark, pp 331–339 (2019)
Golub, Gene H., CFVL, : Matrix Computations, 4th edn. Johns Hopkins Studies in the Mathematical Sciences, Baltimore (2013)
Goldberg, D., Nichols, D., Oki, B.M., Terry, D.: Using collaborative filtering to weave an information tapestry. Commun. ACM **35**(12), 61–71 (1992)
Goldberg, K., Roeder, T., Gupta, D., Perkins, C.: Eigentaste: A constant time collaborative filtering algorithm. Inform. Ret. **4**(2), 133–151 (2001)

Hadash, G., Shalom, O.S., Osadchy, R.: Rank and rate: multi-task learning for recommender systems. In: Proceedings of the 12th ACM Conference on Recommender Systems, Vancouver, British Columbia, Canada, pp 451–454 (2018)

Harper, F.M., Konstan, J.A.: The movielens datasets: History and context. ACM Trans. Inter. Intell. Syst. (TiiS) **5**(4), 1–19 (2016)

He, R., McAuley, J.: Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In: proceedings of the 25th international conference on world wide web, Montréal, Québec, Canada, pp 507–517 (2016)

Herlocker, J., Konstan, J.A., Riedl, J.: An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. Inf. Retrieval **5**(4), 287–310 (2002)

Herlocker, J.L., Konstan, J.A., Borchers, A., Riedl, J.: An algorithmic framework for performing collaborative filtering. In: 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 230–237. California, USA, Berkeley (1999)

Herlocker, J.L., Konstan, J.A., Riedl, J.: Explaining collaborative filtering recommendations. In: Proceedings of the 2000 ACM conference on Computer supported cooperative work, Philadelphia, Pennsylvania, USA, pp 241–250 (2000)

Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. ACM Trans. Inform. Syst. (TOIS) **22**(1), 5–53 (2004)

Huang, Z.: Extensions to the k-means algorithm for clustering large data sets with categorical values. Data Min. Knowl. Disc. **2**(3), 283–304 (1998)

Jannach, D., Zanker, M., Ge, M., Gröning, M.: Recommender systems in computer science and information systems–a landscape of research. In: International Conference on Electronic Commerce and Web Technologies, Springer, Berlin, pp. 76–87 (2012)

Jarvelin, K., Kekalainen, J.: Cumulated gain-based evaluation of IR techniques. ACM Trans. Inform. Syst. (TOIS) **20**(4), 422–446 (2002)

Kalloori, S., Ricci, F.: Improving cold start recommendation by mapping feature-based preferences to item comparisons. In: Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization, Bratislava, Slovakia, pp 289–293 (2017)

Kamitsios, M., Chrysafiadi, K., Virvou, M., Sakkopoulos, E.: A stereotype user model for an educational game: Overcome the difficulties in game playing and focus on the educational goal. In: 2018 9th International Conference on Information, pp. 1–6. Intelligence, Systems and Applications (IISA), IEEE (2018)

Kay, J.: Lies, damned lies and stereotypes: pragmatic approximations of users. University of Sydney, Basser Department of Computer Science (1994a)

Kay, J.: The UM toolkit for cooperative user modelling. User Model. User-Adap. Inter. **4**(3), 149–196 (1994b)

Khalaji, M., Mansouri, K., Mirabedini, S.J.: Improving recommender systems in e-commerce using similar goods. J. Softw. Eng. Appl. **5**(02), 96–101 (2012)

Kluver, D., Konstan, J.A.: Evaluating recommender behavior for new users. In: Proceedings of the 8th ACM Conference on Recommender Systems, Foster City, Silicon Valley California, USA, pp 121–128 (2014)

Koprinska, I., Poon, J., Clark, J., Chan, J.: Learning to classify e-mail. Inf. Sci. **177**(10), 2167–2187 (2007)

Koren, Y.: Factorization meets the neighborhood: a multifaceted collabora-tive filtering model. In: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, pp 426–434 (2008)

Koren, Y.: The BellKor solution to the netflix grand prize. https://www.netflixprize.com/assets/GrandPrize2009_BPC_Bellkor.pdf, accessed: 2019-11-05 (2009)

Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. IEEE Comput. **42**(8), 30–37 (2009)

Kotkov, D., Veijalainen, J., Wang, S.: Challenges of serendipity in recommender systems. In: Proceedings of the 12th International conference on web information systems and technologies, SCITEPRESS, pp 251–256 (2016)

Krulwich, B.: Lifestyle finder: Intelligent user profiling using large-scale demographic data. AI magazine **18**(2), 37–37 (1997)

Lamche, B., Pollok, E., Wörndl, W., Groh, G.: Evaluating the effectiveness of stereotype user models for recommendations on mobile devices. In: UMAP Workshops, Citeseer (2014)

Latif, M.H., Afzal, H.: Prediction of movies popularity using machine learning techniques. Int. J. Comput. Sci. Netw. Sec. (IJCSNS) **16**(8), 127–131 (2016)

Linden, G., Smith, B., York, J.: Amazon. com recommendations: Item-to-item collaborative filtering. IEEE Internet Comput. **1**, 76–80 (2003)

Lops, P., De Gemmis, M., Semeraro, G.: Content-based recommender systems: State of the art and trends. In: Recommender systems handbook, Springer, pp 73–105 (2011)

Mauro, N., Ardissono, L.: Extending a tag-based collaborative recommender with co-occurring information interests. In: Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization, ACM, pp 181–190 (2019)

Mirbakhsh, N., Ling, C.X.: Improving top-n recommendation for cold-start users via cross-domain information. ACM Trans. Know. Disc. Data (TKDD) **9**(4), 1–19 (2015)

Mirbakhsh, N., Ling, C.X.: Leveraging clustering to improve collaborative filtering. Inform. Syst. Front. **20**(1), 111–124 (2018)

Misztal-Radecka, J., Indurkhya, B., Smywiński-Pohl, A.: Meta-user2vec model for addressing the user and item cold-start problem in recommender systems. User Modeling and User-Adapted Interaction pp 1–26 (2020)

Musto, C., de Gemmis, M., Semeraro, G., Lops, P.: A multi-criteria recommender system exploiting aspect-based sentiment analysis of users' reviews. In: Proceedings of the eleventh ACM conference on recommender systems, Como, Italy, pp 321–325 (2017)

Nasery, M., Braunhofer, M., Ricci, F.: Recommendations with optimal combination of feature-based and item-based preferences. In: Proceedings of the 2016 Conference on User Modeling Adaptation and Personalization, Halifax, Canada, pp 269–273 (2016)

O'Connor, M., Herlocker, J.: Clustering items for collaborative filtering. In: Proceedings of the ACM SIGIR workshop on recommender systems, Berkeley, vol 128 (1999)

Paliouras G, Karkaletsis, V., Papatheodorou, C., Spyropoulos, C.D.: Exploiting learning techniques for the acquisition of user stereotypes and communities. In: UM99 User Modeling, Springer, pp 169–178 (1999)

Pazzani, M.J.: A framework for collaborative, content-based and demographic filtering. Artif. Intell. Rev. **13**(5–6), 393–408 (1999)

Podani, J.: Introduction to the exploration of multivariate biological da-ta. Backhuys Publishers, Kerkwerve (2000)

Rana, A., Bridge, D.: Explanations that are intrinsic to recommendations. In: Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization, Singapore, Singapore, pp 187–195 (2018)

Rendle, S.: Factorization machines. In: 2010 IEEE International Conference on Data Mining, pp. 995–1000. Australia, Sydney (2010)

Ricci, F., Rokach, L., Shapira, B.: Recommender systems: introduction and challenges. In: Recommender systems handbook, Springer, Berlin, pp 1–34 (2015)

Rich, E.: User modeling via stereotypes. Cogn. Sci. **3**(4), 329–354 (1979)

Rimaz, M.H., Elahi, M., Bakhshandegan Moghadam, F., Trattner, C., Hosseini, R., Tkalčič, M.: Exploring the power of visual features for the recommendation of movies. In: Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization, Larnaca, Cyprus, pp 303–308 (2019)

Sacharidis, D.: Group recommendations by learning rating behavior. In: Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization, Bratislava, Slovakia, pp 174–182 (2017)

Saito, T., Rehmsmeier, M.: The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. PloS One **10**(3), e0118e0118432 (2015)

Sangam, R.S., Om, H.: The k-modes algorithm with entropy based similarity coefficient. Proc. Comput. Sci. **50**, 93–98 (2015)

Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Application of dimensionality reduction in recommender system-a case study. In: Proceedings of ACM WebKDD Workshop, ACM (2000)

Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Incremental singular value decomposition algorithms for highly scalable recommender systems. Fifth international conference on computer and information science, Citeseer **27**, 27–28 (2002)

Sarwar, B.M., Karypis, G., Konstan, J.A., Riedl, J., et al.: Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th international conference on World Wide Web, Hong Kong, Hong Kong, pp 285–295 (2001)

Schein, A.I., Popescul, A., Ungar, L.H., Pennock, D.M.: Methods and metrics for cold-start recommendations. In: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval, Tampere, Finland, pp 253–260 (2002)

Sedhain, S., Sanner, S., Braziunas, D., Xie, L., Christensen, J.: Social collaborative filtering for cold-start recommendations. In: Proceedings of the 8th ACM Conference on Recommender systems, Foster City, Silicon Valley, California, USA, pp 345–348 (2014)

Shani, G., Meisles, A., Gleyzer, Y., Rokach, L., Ben-Shimon, D.: A stereotypes-based hybrid recommender system for media items. In: Workshop on Intelligent Techniques for Web Personalization, pp. 76–83. Vancouver, Canada (2007)

Sollenborn, M., Funk, P.: Category-based filtering and user stereotype cases to reduce the latency problem in recommender systems. In: European Conference on Case-Based Reasoning, Springer, pp 395–405 (2002)

Spiegel, S., Kunegis, J., Li, F.: Hydra: a hybrid recommender system [cross-linked rating and content information]. In: Proceedings of the 1st ACM international workshop on Complex networks meet information & knowledge management, ACM, pp 75–80 (2009)

Trattner, C., Jannach, D.: Learning to recommend similar items from human judgments. User Model. User-Adap. Inter. **30**(1), 1–49 (2020)

Tsokos, C.P.: Mathematical Statistics with Applications. Elsevier, Amsterdam (2009)

Ungar, L.H., Foster, D.P.: Clustering methods for collaborative filtering. In: AAAI workshop on recommendation systems, AAAI, pp 114–129 (1998)

Wasilewski, J., Hurley, N.: Bayesian personalized ranking for novelty enhancement. In: Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization, Larnaca, Cyprus, pp 144–148 (2019)

Wibowo, A.T., Siddharthan, A., Masthoff, J., Lin, C.: Incorporating constraints into matrix factorization for clothes package recommendation. In: Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization, Singapore, Singapore, pp 111–119 (2018)

Zhang, S., Yao, L., Xu, X.: AutoSVD++ an efficient hybrid collaborative filtering model via contractive auto-encoders. In: Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, pp 957–960 (2017)

Zheng, L., Lu, C.T., Jiang, F., Zhang, J., Yu, P.S.: Spectral collaborative filtering. In: Proceedings of the 12th ACM Conference on Recommender Systems, New York, NY, USA, pp 311–319 (2018)

Zimek, A.: Correlation clustering. PhD thesis, University Munchen, Munchen (2008)

Nourah AlRossais received her Ph.D in Computer Science at University of York, UK. She is a Lecturer at the Department of Computer Science at King Saud University, Saudi Arabia. Her current research interests fall in the areas of Recommender Systems, Knowledge representation, and User Modeling in order to incorporate it into business logic using AI and machine learning.

Daniel Kudenko is a Reserach Group Leader at the L3S Research Center, Leibniz University Hannover, Germany. His research focus is on machine learning (specifically reinforcement learning), and his interests also include user modeling, multi-agent systems, and interactive entertainment.

Tommy Yuan is a Senior Lecturer at the Computer Science Department of the University of York. Dr Yuan conducted research in argumentation and human-computer dialogue systems. His recent interest includes the application of machine learning techniques to various tasks such as argumentation and recommender systems.