IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 28 September 2022, accepted 30 October 2022, date of publication 7 November 2022, date of current version 10 November 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3219071

RESEARCH ARTICLE

Addressing the Cold-Start Problem in Collaborative Filtering Through Positive-Unlabeled Learning and Multi-Target Prediction

ALIREZA GHARAHIGHEHI[1,2], KONSTANTINOS PLIAKOS[3], AND CELINE VENS[1,2]

[1]Itec, IMEC Research Group, KU Leuven, 8500 Kortrijk, Belgium
[2]Department of Public Health and Primary Care, KU Leuven, Campus KULAK, 8500 Kortrijk, Belgium
[3]Department of Management, Strategy and Innovation, KU Leuven, 3000 Leuven, Belgium

Corresponding author: Alireza Gharahighehi (alireza.gharahighehi@kuleuven.be)

This work was supported in part by the Flanders Innovation & Entrepreneurship through the Immosite.com Project, an Innovation Project (Industrial Partners Immosite and g-company) under Project HBC.2020.2674; and in part by the Flemish Government (AI Research Program).

ABSTRACT The cold-start problem is one of the main challenges in recommender systems and specifically in collaborative filtering methods. Such methods, albeit effective, typically can not handle new items or users that do not have any prior interaction activity in the system. In this paper, we propose a novel two-step approach to address the cold-start problem. First, we view the user-item interactions in a positive unlabeled (PU) learning setting and reconstruct the interaction matrix between users and warm items, detecting missing links and recommending warm items to existing users. Second, an inductive multi-target regressor is trained on this reconstructed interaction matrix and subsequently predicts interactions for new items that enter the system. To the best of our knowledge, this is the first time that such a two-step PU learning method is proposed to address the cold-start problem in recommender systems. To evaluate the proposed approach, we employed four benchmark datasets from movie and news recommendation domains with explicit and implicit feedback. We compared our method against three other competitor approaches that address the cold-start problem and showed that our proposed method significantly outperforms them, achieving in a case an increase of 16.9% in terms of NDCG.

INDEX TERMS Recommender systems, collaborative filtering, cold-start problem, PU learning.

I. INTRODUCTION

In the era of digitization and e-commerce, people use online platforms to find their desired products and services. Online platforms can provide an enormous catalogue of items or services to their users, nevertheless, usually each user is interested in a very small fraction of such a catalogue. This makes the role of personalization and recommender systems pivotal. Recommender systems (RSs) are intelligent methods that learn users' preferences and recommend relevant items to them. RSs use user-item interaction history data as well as other types of available information, such as item and user side-information (i.e., features that describe the users/items in the system), to infer users' preferences. Generally, there are two main categories of RSs: content-based (CB) and collaborative filtering (CF) recommenders. CB RSs recommend items whose attributes match the target user profile. However, the main pitfall of CB RSs is that they typically provide over-specified recommendations and are unable to recommend any diverse content. On the other hand, CF RSs use the interactions of other users to infer the preferences of the target user. While CF RSs provide more surprising and usually more accurate recommendations compared to CB RSs [1], [2], they also suffer from some weaknesses. One of

The associate editor coordinating the review of this manuscript and approving it for publication was Fabrizio Messina.

the main pitfalls of CF RSs is that they can not serve new items or users that do not have any prior interaction data in the system. This issue is denoted as the cold-start problem and it is particularly challenging [3], hindering the performance of many applications of RSs. For instance, e-commerce websites need to satisfy the new users in order to gain their trust. Another example stems from the real-estate market where an online platform should immediately recommend a newly advertised property to the relevant users. Therefore, handling the cold-start problem both effectively and efficiently is crucial for RSs.

Two types of cold-start problems are distinguished. New entities (users or items) are called hard cold-start entities when no interactions exist for them or soft cold-start entities when the number of known interactions is very limited [4]. In this paper we focus on hard cold-start entities as they are the most challenging. Apart from cold-start entities, the system cold-start problem may occur [5], where a new RS is introduced to the users, which is not the focus of this paper.

Different approaches have been applied over the years to address the cold-start problem. The simplest way is to serve the new entities (users or items) with non-personalized recommendation such as popularity-based recommendations. Another approach relies on the exploitation of the user or item side-information to predict interactions between such new entities and ones that already exist in the system. Thus, the side-information is employed on top of the collaborative information between users and items to enable a CF RS that serves new items or new users. In this study, we focus on the second approach, handling new items by utilizing the relevant side-information.

Moreover, some approaches try to surpass the cold-start problem by employing classification or regression models built on the side-information of users or items. Such approaches handle missing user-item pairs (i.e. not recorded interactions) as negative instances while training their models. This is denoted as the closed-world assumption [6] and it is often applied as it allows for effective machine learning models to be utilized for solving the recommendation problem. However, we argue against such an assumption, as user-item pairs without any prior interactions are effectively unlabeled data and should not be considered as negative instances. To this end, we consider the recommendation problem as a positive unlabeled (PU) learning task [6]. PU learning is the setting where a learning model is trained on only positive and unlabeled data [6]. This setting naturally fits the recommendation problem, as a typical RS has access to positive user-item interactions (e.g. clicks, likes, scores, purchases) and the rest are unlabeled. The latter means that albeit there are no recorded interactions between the user and the item, the item might be interesting to the user if it was presented to him/her.

Typical RSs are driven through inference models, such as matrix completion (factorization) or graph learning methods. Such methods are typically transductive, meaning that the model requires the new items to be already present in the training process. Whenever a new item arrives, the model has to be re-trained (sometimes partly) in order to be able to provide any predictions. This is a crucial bottleneck that often impairs the performance of online RSs or even makes their application impossible. Although there are a few approaches that extend typical matrix completion or graph learning methods to incorporate side-information, most of them rely on plain neighborhood information and they often underperform when it comes to new items or users.

On the other hand, multi-target prediction (MTP) models can learn from the set of existing users or items and their related features that are already available in the system (i.e. training set). Next, the trained model can be used to predict interactions (probabilities) between cold-start items and the users of the platform. More specifically, multi-target prediction (MTP), also referred to as multi-output prediction, is an extension of standard classification or regression tasks where models learn to predict multiple outputs at the same time [7]. The fundamental assumption behind MTP is that each instance is associated to multiple targets, which are correlated with each other. Therefore, beyond the obvious computational advantages of such a methodology over learning a separate model per target, the model can benefit from existing correlations between the targets and therefore improve its predictive performance. MTP can be divided into multi-target classification (i.e., the targets have categorical values) and multi-target regression (MTR). A special case is multi-label classification where one has only binary values for each target.

In this article, we address the recommendation task through the scope of PU learning, proposing an effective two-step approach to address the cold-start problem in CF. More specifically, in the first step we reconstruct the user-item interaction matrix via semi-supervised learning and collaborative filtering. This way, we identify possible links between users and warm items (items with previous interactions), mitigating sparsity and class imbalance. This inferred set of interactions consists of positive, reliable negative, and predicted user-item interactions. In the second step we train a multi-target regressor (MTR) using relevant side-information of warm items as the features and the inferred user-item interactions as the targets. Then we are able to handle cold-start items using the trained MTR and the features of the new items, thereby accurately and efficiently predicting the user preferences for these new items. As it is known that there is no single model that performs generally best on all problems, we do not commit ourselves to a single algorithm in the different steps of our approach. Instead, in each step we use the model that performs the best among candidate models in a validation set.

For evaluation purposes, in this paper, after having built our two-step PU learning model, we compare it against other methods from the literature, such methods address the cold-start problem in recommendation using different approaches. In this study we focus on cold-start items, as most benchmark

datasets contain rich item related feature representations. Nevertheless, cold-start users can be treated in the same way.

Our contributions can be summarized as follows:

- We propose a novel two-step learning approach to address the cold start problem in recommendation. Our method is the first approach that combines collaborative filtering and multi-target prediction into a PU learning framework.
- We conducted a thorough evaluation study testing our method in the domains of both movie and news recommendation, showing that our approach achieves superior performance to all its competitors.
- We show to the RS community that recommendations for new items (users) can benefit from prior user-item matrix reconstruction.

This paper is organized as follows: in Section II studies about addressing the cold-start problem in RSs are reviewed. Next, in Section III, we discuss our proposed two-step approach. In Section IV, we describe the datasets and the experimental setup of our evaluation study. Next, the results of comparing the proposed method to different methods addressing the cold-start problem in the literature are reported and discussed in Section V. Finally, we conclude and outline some directions for future work in Section VI.

II. RELATED WORK

One way to address the cold-start problem is to serve the new entities with non-personalized recommendations such as random-based, recency-based or popularity-based recommendations. Wang [8] proposed a non-personalized approach called ''ZeroMat'' by using Zipf's Law for the user-item rating distribution which performs better compared to the random-based RS w.r.t. relevance and fairness. In this paper we exploit side-information to address the item cold-start problem. To provide personalized recommendations, one can use a hybrid approach that switches to CB or knowledge-based RSs when there is no available interaction for the new entities [9]. Kawai et al. [10] proposed two hybrid approaches based on content-based filtering and Latent Dirichlet Allocation (LDA) to address the cold-start problem. The difference between these two proposed approaches is whether the topics of the side-information are independent of the topics of the items. Tahmasebi et al. [11] proposed another hybrid approach based on profile expansion to address the cold-start problem. They used user's demographic information to augment the user neighborhoods and expanded the interaction matrix with additional ratings using two heuristic strategies. Feng et al. [12] proposed a hybrid approach which combines Probabilistic Matrix Factorization (PMF) and Bayesian Personalized Ranking (BPR) to address the user soft cold-start problem. Using this combination, their model is capable of exploiting both explicit and implicit feedback from users.

Another direction to address the cold-start problem in CF methods is to extend the CF methods, such as matrix factorization, with side-information in order to serve new entities. Collective Matrix Factorization (CMF) [13], [14] is an extension of matrix factorization where instead of factorizing only the interaction matrix between users and items, it collectively factorizes the interaction matrix as well as the item/user side-information matrix based on a common low-dimensional feature space. Saveski and Mantrach [15] extended the CMF optimization problem by adding non-negativity constraints on the factorized matrices for the sake of interpretability of the factors. They also considered the *manifold assumption* in the objective function, i.e., if two items are close in the real feature space they should be also close in the learned low-dimensional feature space. They called this method Local Collective Embeddings (LCE).

When it comes to PU learning, there have been many approaches that employ a combination of clustering and classification techniques to treat PU learning tasks. For instance, Liu and Peng [16] proposed a clustering-based method followed by an extension of tf-idf to identify strong negative samples prior to document categorization. In [17], k-means was combined with Rocchio [18] to mine strong positive as well as reliable negative examples. k-means was once more employed in [19] to extract strong negative and positive examples, employing SVM for the end task of classification. PU learning for categorical data was addressed in [20] where strong negative and positive samples were identified using kNN and a distance measure denoted as DIstance Learning for Categorical Attributes (DILCA) designed specifically for categorical data. Most of these techniques were originally designed for classification tasks without an extension to more complex tasks such as recommendation. Last, most PU learning methods focus on the detection of reliable negative samples prior to the application of a classifier, discarding the rest of the unlabeled data. In our approach, we discard no information, instead we assign a fuzzy score to ambiguous user-item pairs and let the multi-target prediction models learn from the whole data corpus.

III. METHODOLOGY

In this section we explain the proposed approach. We use the notations defined in Table 1. In recommendation tasks usually there are two main sets of entities, the users and the items. Let $U = \{u_1, u_2, \ldots, u_m\}$ and $I = \{i_1, i_2, \ldots, i_n\}$ be two finite sets, representing users and items, respectively. The already known interactions between such items and users are stored in an interaction matrix $\mathbf{Y} \in \mathbb{R}^{m \times n}$, which can contain ratings when the user feedback is explicit or binary values ($y(u_i, i_j) \in \{0, 1\}$) when the user feedback is implicit. In both cases this interaction matrix is typically very sparse, i.e. there is typically a tiny percentage of positive user-item interactions while most of the pairs are marked as zero. This setting inherently falls under the scope of PU data. This means that user u_i likes item i_j if we have a positive rating but when $y(u_i, i_j) = 0$ the result is inconclusive. Indeed, a zero value is ambiguous and could mean that the user does not like the corresponding item, but could also mean that the item has not yet been presented to the user.

More specifically, the task of a CF-based RS, given the sparse interaction matrix between users and items, is modeling the user preferences over the unseen items and generating ranked lists of recommendations. As it was mentioned, the hard cold-start problem occurs when new entities enter the system and there are no historical interactions for these new entities in the interaction matrix $\mathbf{Y}$. Therefore CF-based RSs are unable to learn the preferences of these new entities. In this paper we focus on the item cold-start problem, where a set of new items $I_C = \{i_{c_1}, i_{c_2}, \ldots, i_{c_w}\}$ are entering the system and the RS should recommend them to relevant users. The only information that is given for these new items is their side-information. For instance, in a movie recommendation task, item-related side-information could be movie genres and cast. For the warm items the feature matrix $\mathbf{X} \in \mathbb{R}^{n \times f}$ and the interaction matrix $\mathbf{Y} \in \mathbb{R}^{m \times n}$ are given, while for the cold-items only the feature matrix $\mathbf{X_C} \in \mathbb{R}^{w \times f}$ is given (w is the number of cold-start items and f is the number of item features). The interaction matrix $\mathbf{Y_C} \in \mathbb{R}^{m \times w}$ is not observed, i.e., it contains only zeros. In this paper we propose a recommendation approach that recommends these new items I_C to most relevant users.

TABLE 1. Notations.

Notation	Description
U	Set of users
I	Set of items
I_C	Set of new items
u_i	User i
i_j	Item j
m	Number of users
n	Number of items
w	Number of new items
f	Number of item features
Y	Interaction matrix between users and items
$\hat{Y}$	Reconstructed matrix between users and items
Y_C	Interaction matrix between users and new items
X	Feature matrix for items
X_C	Feature matrix for new items
$y(u_i, i_j)$	Feedback of user u_i on item i_j
$\hat{y}(u_i, i_j)$	Predicted feedback of user u_i on item i_j
f_{CF}	CF-based recommender system
f_{MTR}	Multi-target regressor
p_{u_i}	Learned latent features of user u_i
q_{i_j}	Learned latent features of item i_j
x_{u_i}	Rating vector of user u_i
w_{i_j}	Learned vector of aggregation coefficients of item i_j
f_θ	Decoder with the learned parameters θ
μ_ϕ	Encoder with the learned parameters ϕ

Our methodology is motivated by the profound link between RSs and PU learning. More specifically, we propose a new methodology that treats user-item interaction data as PU data. Our approach first reconstructs the interaction matrix $\mathbf{Y}$, detecting any missing links between users and items that are already present in the system. This way, our approach forces matrix $\mathbf{Y}$ to become less sparse, mitigating class imbalance and removing part of the noise that innately exists due to the limited user feedback. We subsequently tackle the cold-start problem by handling new items exploiting related side-information. More specifically, we propose the training of multi-target prediction models, such as tree-ensembles, upon such a reconstructed and information enriched interaction matrix. The underlying assumption is that the multi-target prediction model would learn from a substantially less sparse and more informative interaction set.

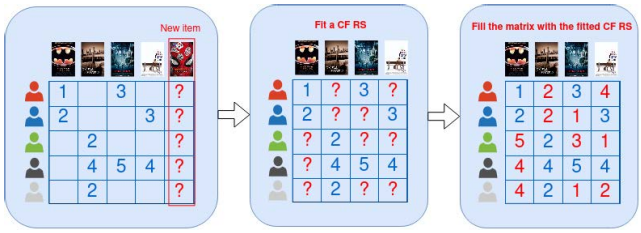


FIGURE 1. Filling the sparse rating/interaction matrix between users and warm items.

A. RECONSTRUCTING THE SPARSE INTERACTION MATRIX

The first step of the proposed approach is to learn the users' preferences on the warm items. As it is shown in Fig.1, we fit a CF-based RS on the interaction matrix to learn the user preferences. The fitted model (f_{CF}) is then used to reconstruct the whole interaction matrix between users and warm items. The elements of this reconstructed matrix ($\hat{\mathbf{Y}}$) are:

$$\hat{y}(u_i, i_j) = \begin{cases} y(u_i, i_j), & \text{if } y(u_i, i_j) > 0 \\ f_{CF}(u_i, i_j), & \text{otherwise.} \end{cases} \quad (1)$$

Based on Eq.1 the reconstructed matrix $\hat{\mathbf{Y}}$ contains the real feedback from users when it is available or the predicted feedback when it is missing.

The choice of f_{CF} only depends on the type of feedback. Below, we illustrate for some example CF methods dealing with explicit or implicit feedback, how they can be plugged into our approach. In the absence of a priori preference, we propose to compare multiple CF methods in a validation set and then select the best performing one to reconstruct the interaction matrix between users and warm items.

1) RECONSTRUCTING THE INTERACTION MATRIX WITH EXPLICIT FEEDBACK

Pure Singular Value Decomposition (SVD) [21] and Non-negative Matrix Factorization (NMF) [22] are CF methods that decompose the interaction matrix to two low-rank matrices for users and items. The learned user and item matrices in NMF contain non-negative values. The interaction matrix can be reconstructed with these two CF methods using Eq. 2:

$$\hat{y}(u_i, i_j) = \begin{cases} y(u_i, i_j), & \text{if } y(u_i, i_j) > 0 \\ \mathbf{p_{u_i}} \cdot \mathbf{q_{i_j}}, & \text{otherwise.} \end{cases} \quad (2)$$

where $\mathbf{p_{u_i}}$ and $\mathbf{q_{i_j}}$ are the learned latent features of user u_i and item i_j respectively.

SLIM [23] is a linear method that learns the sparse aggregation coefficient square matrix $\mathbf{W}$ using the optimization

problem regularized with L1 and L2 norms. The interaction matrix can be reconstructed with SLIM using Eq. 3:

$$\hat{y}(u_i, i_j) = \begin{cases} y(u_i, i_j), & \text{if } y(u_i, i_j) > 0 \\ \mathbf{x_{u_i}} \cdot \mathbf{w_{i_j}}, & \text{otherwise.} \end{cases} \quad (3)$$

where $\mathbf{x_{u_i}}$ is the rating vector of user u_i and $\mathbf{w_{i_j}}$ is the learned sparse size-n column vector of aggregation coefficients for item i_j.

User-based and item-based KNN (UKNN and IKNN) are memory-based CF methods that predict the missing interactions using the interactions of neighbor users/items. The missing scores in the interaction matrix are predicted by UKNN and IKNN using the weighted average of the scores of neighbor users/items. The weight of each neighbor is the similarity of its interaction vector with the interaction vector of the target user/item.

2) RECONSTRUCTING THE INTERACTION MATRIX WITH IMPLICIT FEEDBACK

Bayesian Personalized Ranking (BPR) [24], Weighted Approximate-Rank Pairwise (WARP) [25] and Weighted Regularized Matrix Factorization (WRMF) [26] are CF methods for implicit feedback. BPR is a learning-to-rank CF method that uses pairwise preferences to learn users' and items' latent features. (WARP) [25] is another CF method for implicit feedback that was initially proposed for annotating images, but later on was used as a learning-to-rank RS. Weighted Regularized Matrix Factorization (WRMF) [26] uses the alternating-least-squares optimization approach to learn parameters. All of these three methods learn users' and items' latent features (q and p) and therefore the interaction matrix can be reconstructed using Eq. 2.

MVAE [27] is a CF RS for implicit feedback based on variational autoencoders with the assumption that the user logs are from a multinomial distribution. Given a trained MVAE recommender on X, the interaction matrix can be reconstructed by:

$$\hat{y}(u_i, i_j) = \begin{cases} y(u_i, i_j), & \text{if } y(u_i, i_j) > 0 \\ f_\theta(\mathbf{p_{u_i}})_{i_j}, & \text{otherwise.} \end{cases} \quad (4)$$

where f_θ is the decoder with the learned parameters θ, $\mathbf{p_{u_i}} = \mu_\phi(\mathbf{x_{u_i}})$ represents the learned latent features for user u_i and μ_ϕ is the learned encoder.

Last, the reconstructed matrix $\hat{\mathbf{Y}}$ with the mentioned methods or any other CF method may need re-scaling to have the same scale as the original interaction matrix $\mathbf{Y}$.

B. CASTING THE COLD-START RECOMMENDATION PROBLEM AS MULTI-TARGET REGRESSION

The second step of the proposed approach is to fit a MTR using warm items as training instances and users as targets (See Figure 2). Features of the warm items $\mathbf{X}$ are considered as inputs to the MTR and the reconstructed matrix $\hat{\mathbf{Y}}$ from the previous step is used as target set:

$$\hat{\mathbf{Y}}^\intercal \leftarrow f_{MTR}(\mathbf{X}) \quad (5)$$

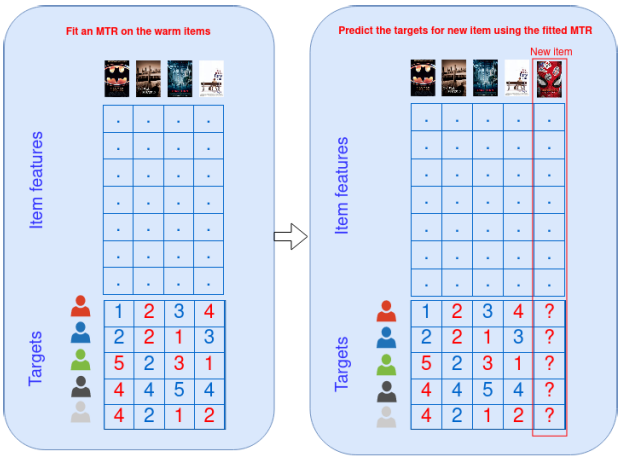


FIGURE 2. Predicting the targets (users) for new items using the fitted multi-target regressor.

The trained MTR f_{MTR} is then used to predict the scores of the users for the cold-items $\mathbf{X_C}$:

$$\hat{\mathbf{Y}}_C^\intercal = f_{MTR}(\mathbf{X_C}) \quad (6)$$

where $\hat{\mathbf{Y}}_C$ is the predicted preferences of the users on the cold-items. Then these predictions are used to decide for each cold-item which users should receive it as a recommendation.

MTR versions of tree-ensemble algorithms, such as Random Forests (RF) [28] or Extremely Randomized Trees (ERT) [29], have been proved very effective. RF consists of a collection of multiple decision trees. The tree growing process is driven by a splitting criterion, selecting the best split. Many such splitting criteria exist, with variance reduction being the most typical one. A key factor of RF is the diversity that is enforced among the trees by utilizing bootstrap replicates of the training set as well as implementing a random selection mechanism of the features during the tree learning process. ERT is an extension of RF where, similar to RF, each tree of the ensemble is trained using a random subset of the features as split-candidates in each node. The difference in ERT is that for every feature from the κ selected ones, one split threshold is picked at random. Next, the best split from the κ picked ones is selected.

Both RF and ERT have been innately extended to MTP by transferring the splitting criterion to the multi-output space. More specifically, the criterion is computed over the whole set of outputs, typically as the sum of the variance of each output.

Tree-ensemble learning algorithms are computationally very efficient. They are inductive methods, naturally memory efficient, and can be very easily parallelized, as every tree in the ensemble can grow independently. Last, tree ensembles are also known for their innate interpretability, since they can first provide the user with a feature ranking, disclosing the features that are crucial for a prediction, and second, provide a set of rules that explain a specific prediction. The latter can be further leveraged with existing tree-approximation

strategies [30]. Other MTP models also exist, for example, multi-target K-Nearest Neighbors Regressor (KNNR) is a simple approach based on KNN, where the predictions are based on averaging the outputs of the k nearest neighbors of the test sample.

IV. EXPERIMENTAL SETUP

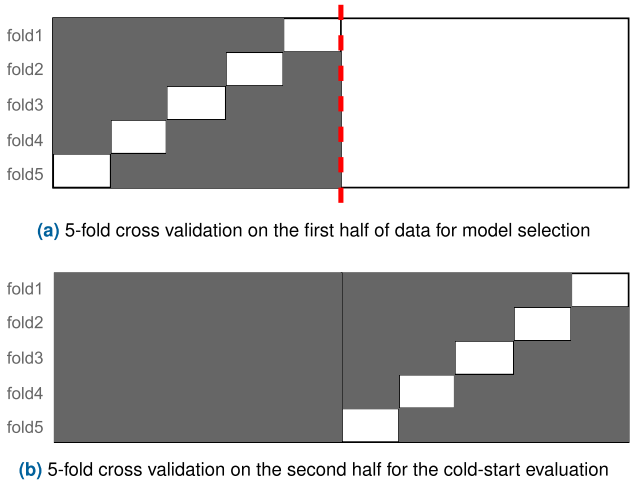
A. DATASET DESCRIPTION

In this paper we used 4 datasets from two different domains with explicit and implicit feedback. In particular, we used MovieLens-1m and MovieLens-20m datasets [31] (hereafter, we refer to these datasets as *ML-1m* and *ML-20m*, respectively), which contain users' explicit ratings on movies and *Adressa* [32] as well as *Globo* [33] datasets which contain user implicit feedback on news articles. These datasets are described in Table 2. In the movie datasets the genres and cast of movies are available and used as item features. In the *Adressa* dataset, for each news article the related keywords, authors and topic are available and considered as features describing the news articles. For the *Globo* dataset, the generated article embeddings by a deep neural network model [34] based on article text and tags are used as item features.

B. EXPERIMENT DESIGN

As mentioned in the previous section (Sec. III), we propose to select the best CF model and the best MTR model among candidate models before generating recommendations for the cold-start items. Users and items with less interactions than a threshold are dropped from the experiments.[1] We use a cross validation scheme to avoid any information leakage between the model selection step and evaluation of the cold-start recommendation task (See Fig. 3). We first randomly split the dataset items to two disjoint parts, one for the model selection step (Fig. 3a) and the other for evaluating the cold-start recommendation task (Fig. 3b). In the first part, we select the best CF model and the best MTR based on 5-fold cross validation (CV). For selecting the best CF model five interactions per item are considered as test interactions in the test fold. For selecting the best MTR the items in the test fold are considered as test items. The hyperparameters of the models (CF and MTR) are internally tuned[2] in one of the folds. In the selected fold the parameters are tuned again based on CV. To tune the hyperparameters of CF methods and cold-start RSs, we used the "forest_minimize" from "scikit-optimize" library and for MTR methods we used "GridSearchCV" from "scikit-learn". For CF methods and cold-start RSs we used *NDCG*, and for *MTR* methods we used MAE to select the best hyperparameters.

Then, when the best CF model, the best MTR as well as their corresponding hyperparameters are fixed, we use the second part of the datasets to evaluate the item cold-start recommendation task. The second part of the datasets is also split in 5-fold CV and each time we use the items in the test fold as the test cold-items to evaluate our proposed approach. The items in the other four folds are combined with the items from the model selection part, for a final matrix completion calculation and for training a final MTR model, using the selected approaches.



(a) 5-fold cross validation on the first half of data for model selection

(b) 5-fold cross validation on the second half for the cold-start evaluation

FIGURE 3. The scheme of 5-fold cross validation for model selection and the cold-start evaluation.

Last, we evaluate statistically significant differences between two methods by employing a *Wilcoxon signed-rank test* [35] ($\alpha = 0.01$) to show that our proposed approach is significantly better compared to the second best comparative method in the results. Typically for one to apply such a test, more than ten paired independent observations are required. As we did not have such volume of data at our disposal, we computed the test on the different folds and repeated the experiments with three different random seeds (overall 15 paired observations).

C. COMPETITOR APPROACHES

In this section, we first present the CF and MTR approaches from which we select the best performing models for the two parts of our approach. In the first step, which is selecting the CF model, SVD [21], NMF [22], UKNN [36], IKNN [37] and SLIM [23] are considered for the datasets with explicit feedback and BPR [24], WRMF [26], WARP [25] and MVAE [27] are used for datasets with implicit feedback. For the second step RF, ERT and KNNR are used as MTRs in the proposed approach.

Finally, to evaluate our proposed approach we compare it against the following competitor approaches that address the cold-start problem:

- **CB**: the classic CB RS that aggregates users' previous interactions to create user profiles and then recommends users whose profiles have the highest *cosine similarity* with the cold-item features.
- **CMF**: the Collective Matrix Factorization (CMF) [14] method explained in Section II. In this method we

[1] This threshold is 30, 100, 100 and 50 for ML-1m, ML-20m, Adressa and Globo respectively.

[2] The detailed information about the selected hyperparameters is reported in Appendix VI.

TABLE 2. Datasets descriptions.

	ML-1m	ML-20m	Adressa	Globo
item type	movie	movie	news article	news article
# users	6,040	138,493	159,103	322,897
# items	3,706	26,744	2,363	46,033
# features	36,626	33,097	1,261	250
density	4.5%	0.54%	1.3%	0.02%
feedback	Explicit	Explicit	Implicit	Implicit
features	genres (18) and cast (36,608)	genres (19) and cast (33,078)	keywords (1028), authors (205) and topic (28)	embeddings

collectively factorize the interaction matrix and the item feature matrix.

- **LCE**: the Local Collective Embeddings (LCE) [15] approach explained in Section II, which extends CMF with non-negativity constrains and the locality assumption.

D. EVALUATION MEASURES

To evaluate the models in each step of the proposed approach we use different evaluation measures. To select the best CF RS in the first step, we use three measures, namely *recall*, *MAP*[3] and *NDCG*.[4] *Recall* is a standard information retrieval measure that reflects the proportion of relevant items that are recommended. *MAP* and *NDCG* are rank-sensitive relevance measures. In the second step, we use MAE[5] averaged over the targets to evaluate predictions of MTRs. Finally, *NDCG* and *MAP* are used to evaluate the proposed approach and the other competitor methods that address the cold-start problem.

V. RESULTS AND DISCUSSION

The results of applying the proposed approach with different base models are summarized in Table 3, 4, 5 and 6.

TABLE 3. Results of CF RSs on datasets with explicit feedback (measures are all based on top@10).

	ML-1m			ML-20m		
	MAP	**recall**	**NDCG**	**MAP**	**recall**	**NDCG**
SVD	0.151	0.256	0.235	0.139	0.220	0.202
NMF	0.143	0.233	0.213	0.127	0.190	0.172
SLIM	**0.157**	**0.276**	**0.258**	**0.153**	**0.264**	**0.255**
IKNN	0.138	0.228	0.210	0.140	0.236	0.224
UKNN	0.142	0.230	0.211	0.131	0.204	0.188

As it is shown in Table 3, in both datasets with explicit feedback (ML-1m and ML-20), SLIM performs the best w.r.t. all three evaluation measures. For implicit datasets, as reported in Table 4, MVAE has the best performance in all three evaluation measures compared to the other CF-based RSs. Therefore, we select SLIM and MVAE to reconstruct the interaction matrix between users and items for datasets with explicit and implicit feedback, respectively. Next, using

[3]Mean Average Precision.

[4]Normalized discounted Cumulative Gain.

[5]Mean Absolute Error.

TABLE 4. Results of CF RSs on datasets with implicit feedback (measures are all based on top@10).

	Adressa			Globo		
	MAP	**recall**	**NDCG**	**MAP**	**recall**	**NDCG**
BPR	0.117	0.164	0.139	0.116	0.165	0.138
WARP	0.117	0.164	0.139	0.116	0.166	0.138
WRMF	0.141	0.217	0.182	0.136	0.209	0.179
MVAE	**0.145**	**0.226**	**0.191**	**0.140**	**0.219**	**0.187**

TABLE 5. Results of MTRs w.r.t. *MAE*.

	ML-1m	ML-20m	Adressa	Globo
RF	**0.8036**	**0.5261**	0.2133	0.1742
ERT	0.8167	0.5294	**0.2083**	**0.1737**
KNNR	0.8224	0.5289	0.2259	0.1741

the reconstructed matrices and items' features, three MTRs are trained and evaluated. As reported in Table 5, the best performing MTRs are RF and ERT in our explicit and implicit datasets, respectively.

The results of applying our proposed approach PULCO[6] with the selected CF and MTR method and the competitor methods that address the cold-start problem are summarized in Table 6. As shown in the table, PULCO has superior performance over all the competing methods and statistically significantly outperforms the second best approach in all datasets. In the explicit feedback datasets, the LCE method is the second best while in the implicit datasets, CMF is the second best performing approach. Although the CB approach is quite simple and straightforward, it performs relatively well in comparison to CMF and LCE. A possible reason is that the item feature space is relatively rich and therefore the CB approach can effectively model user profiles.

In the proposed approach we used different base models for each step of the algorithm. We selected the CF baselines based on the results of the award winning paper [38], which showed that the memory-based approaches (UKNN and IKNN), SLIM and MVAE outperform recent complex deep neural network based approaches. We selected RF, ERT and KNNR, as they are well-established multi-target regression models and are also computationally efficient.

[6]Positive-Unlabeled Learning for COld start problems.

TABLE 6. Results of recommendations for cold-start items w.r.t. *NDCG@10* and *MAP@10*.

	ML-1m		ML-20m		Adressa		Globo	
	NDCG	**MAP**	**NDCG**	**MAP**	**NDCG**	**MAP**	**NDCG**	**MAP**
PULCO	0.504*	0.099*	0.469*	0.099*	0.546*	0.097*	0.354*	0.092*
CMF	0.267	0.083	0.073	0.046	0.377	0.092	0.293	0.081
LCE	0.393	0.092	0.411	0.095	0.227	0.068	0.190	0.068
CB	0.264	0.086	0.142	0.057	0.310	0.081	0.222	0.081

"*" in each column means that the best approach statistically significantly outperforms the second best approach.

TABLE 7. Hyperparameters for datasets with explicit feedback.

			range	ML-1m	ML-20m
CF methods	SVD	# latent features	(20, 200)	26	76
	NMF	# latent features	(20,200)	26	81
		L1-ratio	(0.1,0.9)	0.828	0.376
	SLIM	λ	(1e-5,1.0)	0.003	0.549
		β	(1e-3, 1.0)	0.964	0.001
		topk	(50, 600)	160	327
	IKNN	# neighbors	(20, 800)	88	77
		shrink term	(0,1000)	432	767
	UKNN	# neighbors	(20, 800)	205	513
		shrink term	(0,1000)	398	390
MTR methods	RF	# estimators	[100,200]	200	200
		min samples leaf	[1-5]	5	2
	ERT	# estimators	[100,200]	1276	1393
		min samples leaf	[1-5]	5	2
	KNNR	# neighbors	[2,5,10,50]	50	50
		weights	[uniform, distance]	uniform	uniform
Cold RSs	CMF	# latent features	(5,150)	109	91
		λ	(0.001,600)	554.45	86.72
		# iterations	(5,150)	6	90
	LCE	# latent features	(50,700)	673	58
		α	(0.1,0.9)	0.822	0.512
		β	(0.05,0.5)	0.075	0.168
		λ	(0.1,10)	8.64	9.90
		max_iter	(100,700)	577	137

Nevertheless, we should highlight that the proposed approach does not depend on a specific CF or MTR model and it is robust enough to accommodate other possible combinations as well, handling the cold-start problem both efficiently and effectively.

Our proposed approach fits perfectly to real-case recommender systems, as the latter have typically limited interactions in the user-item matrix. This means that most of the possible user-item interactions have not been recorded, nonetheless, it is likely that the user would be interested in some items in case they are presented to him/her. Our method takes into account this fact, reconstructing the user-warm item matrix prior to handling any cold-start items. This way, the inferred set of interactions consists of positive, reliable negative, and predicted interactions between users and warm items. As it is reflected in the obtained results, this methodology gives us an advantage over the competitor methods.

In real-life recommendation tasks, before a system is put in production, usually several types of CF methods are compared via A/B testing with real users in order to select the best performing one. This effectively removes the need for the CF selection step of our proposed approach. Once this best performing CF method is known, it can be used to reconstruct the interaction matrix for the first step of our proposed approach. Multiple repetitions of such comparisons between several CF approaches will not be needed in such real-life applications, something that also extends to the second step of our proposed approach. Furthermore, the second step with the tree-ensemble methods can be parallelized and implemented effectively. Serving the cold-start items does not therefore present a large computational burden on the

TABLE 8. Hyperparameters for datasets with implicit feedback.

			range	Adressa	Globo
CF methods	BPR	# latent features	(50, 300)	296	101
		# iterations	(10, 300)	16	87
		learning rate	(10**-4, 10**-1)	0.00025	0.0007
		user regularization	(1e-6, 10**-1)	4.49e-06	2.93e-06
		item regularization	(1e-6, 10**-1)	0.060	0.076
	WARP	# latent features	(50, 300)	273	271
		# iterations	(10, 300)	134	236
		learning rate	(10**-4, 10**-1)	0.0015	0.0017
		user regularization	(1e-6, 10**-1)	1.19e-4	7.48e-06
		item regularization	(1e-6, 10**-1)	0.023	0.017
	WRMF	# latent features	(10, 300)	19	11
		# iterations	(10, 300)	35	204
		regularization	(1e-5, 10**-1)	0.66	0.027
	MVAE	# batch size	(25, 500)	59	391
		# iterations	(10, 250)	177	99
		anneal steps	(100000,300000)	100119	127354
MTR methods	RF	# estimators	[100,200]	200	200
		min samples leaf	[1-5]	2	3
	ERT	# estimators	[100,200]	200	200
		min samples leaf	[1-5]	5	2
	KNNR	# neighbors	[2,5,10,50]	10	50
		weights	[uniform, distance]	uniform	distance
Cold RSs	CMF	# latent features	(5,150)	27	71
		λ	(0.001,600)	13.64	389.96
		# iterations	(5,150)	14	9
	LCE	# latent features	(50,700)	683	144
		α	(0.1,0.9)	0.379	0.377
		β	(0.05,0.5)	0.174	0.468
		λ	(0.1,10)	4.52	2.700
		max_iter	(100,700)	192	289

whole recommendation task. This is a crucial advantage of inductive models over transductive competitors. The latter require the new items to be already present in the training process, which is not usually possible. In case recommendations have to be provided for new items, transductive models need to be re-trained, a process that is typically computationally very expensive and therefore makes the online application of corresponding RSs particularly difficult.

VI. CONCLUSION

In this paper we have addressed the cold-start problem in recommendation through PU learning. More specifically, we have deployed an effective two-step approach integrating powerful CF methods with fast and accurate multi-target prediction models. In the first step, we reconstructed the interaction matrix between users and warm items via a collaborative filtering recommender, identifying reliable negative user-item pairs and assigning a score to the rest of the (ambiguous) unlabeled data. Next, in the second step, we trained a multi-target regressor on warm item features and the reconstructed interaction matrix from the first step, efficiently predicting scores for hard cold-items. We showed that the proposed approach significantly outperforms the extended versions of matrix factorization for cold-start problem, i.e., collective matrix factorization and local collective embeddings models, as well the content-based recommender system in all four datasets considered. The proposed approach is flexible and robust in the sense that (1) it does not depend on the type of feedback (implicit or explicit) as well as the choice of CF or MTR models, and (2) it does not require retraining the model when a new cold item arrives.

The application of our work to the online learning setting would be a great direction for future research. In addition, the extension of our work to the field of multi-view learning, where one could integrate multi-modal feature sets of items or users to handle item or user cold-start cases, would be interesting. Last, it would be very interesting to extend this approach to pairwise learning handling user-item pairs by integrating user and item feature sets in a unified framework.

CONFLICT OF INTEREST

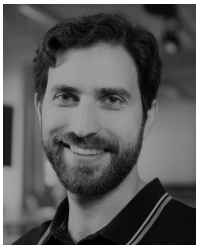
The authors declare that they have no conflict of interest.

APPENDIX A HYPERPARAMETER TUNING

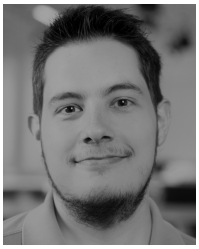
In this Appendix, details about the hyperparameter selection procedure are reported in Table 7 and Table 8, for explicit and implicit datasets, respectively.

REFERENCES

[1] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 734–749, Jun. 2005.
[2] P. Lops, M. D. Gemmis, and G. Semeraro, "Content-based recommender systems: State of the art and trends," in *Recommender Systems Handbook*. Boston, MA, USA: Springer, 2011, pp. 73–105.
[3] S.-T. Park and W. Chu, "Pairwise preference regression for cold-start recommendation," in *Proc. 3rd ACM Conf. Recommender Syst. (RecSys)*, 2009, pp. 21–28.
[4] K. Joseph and H. Jiang, "Content based news recommendation via shortest entity distance over knowledge graphs," in *Proc. World Wide Web Conf.*, May 2019, pp. 690–699.
[5] R. Sethi and M. Mehrotra, "Cold start in recommender systems-a survey from domain perspective," in *Intelligent Data Communication Technologies and Internet of Things*. Singapore: Springer, 2021, pp. 223–232.
[6] J. Bekker and J. Davis, "Learning from positive and unlabeled data: A survey," *Mach. Learn.*, vol. 109, no. 4, pp. 719–760, Apr. 2020.
[7] W. Waegeman, K. Dembczyński, and E. Hüllermeier, "Multi-target prediction: A unifying view on problems and methods," *Data Mining Knowl. Discovery*, vol. 33, pp. 1–32, Nov. 2018.
[8] H. Wang, "ZeroMat: Solving cold-start problem of recommender system with no input data," in *Proc. IEEE 4th Int. Conf. Inf. Syst. Comput. Aided Educ. (ICISCAE)*, Sep. 2021, pp. 102–105.
[9] C. C. Aggarwal, "Ensemble-based and hybrid recommender systems," in *Recommender Systems*. Cham, Switzerland: Springer, 2016, pp. 199–224.
[10] M. Kawai, H. Sato, and T. Shiohama, "Topic model-based recommender systems and their applications to cold-start problems," *Expert Syst. Appl.*, vol. 202, Sep. 2022, Art. no. 117129.
[11] F. Tahmasebi, M. Meghdadi, S. Ahmadian, and K. Valiallahi, "A hybrid recommendation system based on profile expansion technique to alleviate cold start problem," *Multimedia Tools Appl.*, vol. 80, no. 2, pp. 2339–2354, Jan. 2021.
[12] J. Feng, Z. Xia, X. Feng, and J. Peng, "RBPR: A hybrid model for the new user cold start problem in recommender systems," *Knowl.-Based Syst.*, vol. 214, Feb. 2021, Art. no. 106732.
[13] A. P. Singh and G. J. Gordon, "Relational learning via collective matrix factorization," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2008, pp. 650–658.
[14] D. Cortes, "Cold-start recommendations in collective matrix factorization," 2018, *arXiv:1809.00366*.
[15] M. Saveski and A. Mantrach, "Item cold-start recommendations: Learning local collective embeddings," in *Proc. 8th ACM Conf. Recommender Syst. (RecSys)*, 2014, pp. 89–96.
[16] L. Liu and T. Peng, "Clustering-based method for positive and unlabeled text categorization enhanced by improved TFIDF," *J. Inf. Sci. Eng.*, vol. 30, no. 5, pp. 1463–1481, 2014.
[17] F. Lu and Q. Bai, "Semi-supervised text categorization with only a few positive and unlabeled documents," in *Proc. 3rd Int. Conf. Biomed. Eng. Informat.*, Oct. 2010, pp. 3075–3079.
[18] C. D. Manning, P. Raghavan, and H. Schütze, "Introduction to information retrieval," *Natural Lang. Eng.*, vol. 16, no. 1, pp. 100–103, 2010.
[19] S. Chaudhari and S. Shevade, "Learning from positive and unlabelled examples using maximum margin clustering," in *Proc. Int. Conf. Neural Inf. Process.* Berlin, Germany: Springer, 2012, pp. 465–473.
[20] D. Ienco and R. G. Pensa, "Positive and unlabeled learning in categorical data," *Neurocomputing*, vol. 196, pp. 113–124, Jul. 2016.
[21] P. Cremonesi, Y. Koren, and R. Turrin, "Performance of recommender algorithms on top-n recommendation tasks," in *Proc. 4th ACM Conf. Recommender Syst. (RecSys)*, 2010, pp. 39–46.
[22] A. Cichocki and A.-H. Phan, "Fast local algorithms for large scale nonnegative matrix and tensor factorizations," *IEICE Trans. Fundam. Electron., Commun. Comput. Sci.*, vol. 92, no. 3, pp. 708–721, 2009.
[23] X. Ning and G. Karypis, "SLIM: Sparse linear methods for top-N recommender systems," in *Proc. IEEE 11th Int. Conf. Data Mining*, Dec. 2011, pp. 497–506.
[24] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: Bayesian personalized ranking from implicit feedback," 2012, *arXiv:1205.2618*.
[25] J. Weston, S. Bengio, and N. Usunier, "WSABIE: Scaling up to large vocabulary image annotation," in *Proc. 22nd Int. Joint Conf. Artif. Intell.*, 2011, pp. 2764–2770.
[26] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in *Proc. 8th IEEE Int. Conf. Data Mining*, Dec. 2008, pp. 263–272.
[27] D. Liang, R. G. Krishnan, M. D. Hoffman, and T. Jebara, "Variational autoencoders for collaborative filtering," in *Proc. World Wide Web Conf. World Wide Web (WWW)*, 2018, pp. 689–698.
[28] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
[29] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Mach. Learn.*, vol. 63, no. 1, pp. 3–42, Mar. 2006.
[30] K. Dedja, F. K. Nakano, K. Pliakos, and C. Vens, "Explaining random forest prediction through diverse rulesets," 2022, *arXiv:2203.15511*.
[31] F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *ACM Trans. Interact. Intell. Syst.*, vol. 5, no. 4, pp. 1–19, 2015.
[32] J. A. Gulla, L. Zhang, P. Liu, O. Özgöbek, and X. Su, "The Adressa dataset for news recommendation," in *Proc. Int. Conf. web Intell.*, 2017, pp. 1042–1048.
[33] G. de Souza Pereira Moreira, F. Ferreira, and A. M. da Cunha, "News session-based recommendations using deep neural networks," in *Proc. 3rd Workshop Deep Learn. Recommender Syst.*, 2018, pp. 15–23.
[34] G. D. S. P. Moreira, D. Jannach, and A. M. D. Cunha, "Contextual hybrid session-based news recommendation with recurrent neural networks," *IEEE Access*, vol. 7, pp. 169185–169203, 2019.
[35] R. F. Woolson, "Wilcoxon signed-rank test," in *Wiley Encyclopedia of Clinical Trials*. Hoboken, NJ, USA: Wiley, 2007, pp. 1–3.
[36] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Proc. 14th Conf. Uncertainty Artif. Intell.*, 1998, pp. 43–52.
[37] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proc. 10th Int. Conf. World Wide Web*, 2001, pp. 285–295.
[38] M. F. Dacrema, P. Cremonesi, and D. Jannach, "Are we really making much progress? A worrying analysis of recent neural recommendation approaches," in *Proc. 13th ACM Conf. Recommender Syst.*, Sep. 2019, pp. 101–109.



ALIREZA GHARAHIGHEHI received the Ph.D. degree in engineering sciences from KU Leuven, in 2022, and the master's degree in artificial intelligence, international business economics and management and industrial engineering. He is currently a Postdoctoral Researcher at itec, IMEC Research Group, KU Leuven. His research interests include recommender systems and personalization, beyond relevance evaluation, and cold-start problem in recommender systems.



KONSTANTINOS PLIAKOS received the Diploma degree in electrical and computer engineering and the M.Sc. degree in computational intelligence from AUTH, in 2011 and 2015, respectively, and the Ph.D. degree from KU Leuven, Belgium, in 2019. He is currently a Postdoctoral Researcher at KU Leuven. His research interests include multi-label and multi-target prediction, supervised and semi-supervised learning, dimensionality reduction, recommender systems, and biomedical network mining.

CELINE VENS received the Ph.D. degree in computer science (machine learning) from KU Leuven, Belgium, in 2007. She is currently an Associate Professor at the Faculty of Medicine, KU Leuven, and itec, where she is also at IMEC Research Group. Her research interests include multi-label, multi-target, hierarchical prediction, recommender systems, tree ensemble learning, survival analysis, and biological network mining.