Addressing the Blurry Collaborative Embedding Problem for Cold-Start Music Recommendation

Mahsa Aghazadeh
Department of GSDS
Human Factors & Ergonomics Lab
KAIST, South Korea
mahsa_agz@kaist.ac.kr

Dewei Zhu
Department of ISysE
Human Factors & Ergonomics Lab
KAIST, South Korea
deweizhu@kaist.ac.kr

Karin Rizky Irminanda
Department of ISysE
Human Factors & Ergonomics Lab
KAIST, South Korea
karinirminanda@kaist.ac.kr

Anurag Yadav
Graduate School of Data Science
CSD lab
KAIST, South Korea
anusim2003@kaist.ac.kr

ABSTRACT

The continuous expansion of the internet society, marked by the rapid increase in new items and users, highlights the critical of the cold-start problem in the music domain. Existing strategies often resolve this by projecting new item content onto warm-item embeddings, enabling collaborative filtering. While effective, this approach is limited by available item features and past interactions, reducing serendipity and novelty. Moreover, neglecting user-item interaction data during training can lead to blurry collaborative embeddings, impacting recommendations. We propose the application of Contrastive Collaborative Filtering for Cold-Start Recommendation (CCFCRec) in the music domain, combining co-occurrence collaborative signals and contrastive learning. Further, we improved the model by expanding the attributes to include artists and albums in addition to genres. The final model exhibits improved performance from the former and outperforms KNN model. Nevertheless, the model can be improved through involving more diverse attributes and implementing adaptive reinforcement learning for increased adaptability.

CCS CONCEPTS

• Information system → Recommender systems.

KEYWORDS

Music Recommender Systems, Co-occurrence Collaborative Filtering, Content Collaborative Filtering, Cold-start Recommendation, Contrastive Learning

1 INTRODUCTION

Constructing an effective recommendation system comes with challenges, including continuously changing data and user preferences and lack of data availability [1]. Among those obstacles, we have chosen to focus on a specific concern — the cold-start problem. This choice is grounded by the continuous growth of the internet society. Netflix, for instance, as one of the leading OTT (Over-The-Top) platforms, experienced the addition of 11 million new subscribers and the introduction of about 700 new content items last year [2]. Similarly, Spotify, a major music streaming platform, gained 25 million new users in 2022 and added around 100 thousand new music contents every year [3]. As the internet society keeps expanding, the constant arrival of new users and items, often without historical data, implies that the cold-start issue remains a persistent concern. Thus, addressing the specific complexities posed by the cold-start problem is crucial in the context of recommendation systems.

Among various recommendation system domains, music possesses characteristic that makes it particularly susceptible to the cold start problem. First, the creation of a song demands less time and effort compared to the development of a game or a two-hour movie. In addition, artists are often compensated by the music platforms based on the play count of the artists' songs, encouraging the production of shorter-duration tracks in recent years. Not to mention, platforms like Spotify empower users to publish their music independently, even without the backing of recording labels. Consequently, this dynamic has led to a substantial increase in the annual number of music releases, surpassing that of other domains.

This unique landscape has resulted in the music domain having a significantly more diverse array of artists and genres when compared to movies, books, or games. Therefore, it is essential to recognize that, cold-start problem holds a distinctive and crucial position in the music recommendation systems.

Existing methods typically address the issue of cold-start problem by projecting the content of a new item (cold start data points) onto an embedding of an already existing item (warm data points). Using content-based method, DropoutNet randomly drops (ignoring) parts of the input data to make the system produce a generalize result even with missing information [4], while MWUF generates warm embeddings for cold-start items based on their features and ID embeddings [5]. LARA, on the other hand, used GANB-based methods by learning a mapping from cold-start items' attributes to user embeddings to generate virtual users for cold-start items [6]. The drawback of those methods is that the recommendation will only be limited to the item's available features and cannot convey the cold item's unique characteristics. In addition, since there is no interaction history between users and cold items, the recommendations rely heavily on the item's available features and users' past interactions, resulting in recommendations of items with similar characteristics and lack of serendipity and novelty.

In addition, as cold-items and warm-items exhibit differences in attributes or embeddings, simply projecting cold-items into existing warm-item embeddings will potentially diminish the specificity of the collaborative embeddings. This phenomenon is often called blurry collaborative embeddings [7]. To illustrate this problem further, let's consider a user who has a strong preference for K-pop songs. During the training process, the embedding of K-pop songs will be closely aligned with that of the user. Let the user dislike a band called BTN, which has the attributes of K-Pop, popular, male group. Such user behavior will cause divergence between the user embedding and the K-POP embedding. Consequently, when making recommendations using trained algorithms, they may not adequately account for the user's specific liking for K-pop music and thus reduce its inclusion in recommended choices – contradicting their preferences.

In music recommendation, several ideas were proposed to tackle the cold-start problem. Pulis and Bajada utilize content-based, specifically by transforming the frequency (using Mel spectrogram) of songs and recommend users new songs based on the similarity with the songs they interacted before [8]. This method gives bigger opportunities for new artists that are less popular so that their songs can be recommended to wider audiences. However, it could limit the diversity of song discovery, as the recommendation is given based on songs' frequency similarity. Soleymani, et. al. proposed a set of attributes derived from psychological studies of music preference, to better describe the underlying factors of music preference compared to just music genre. This way, the recommendation can more accurately find the user's real preferences. Yet, as it promotes the discovery of less popular songs, the recommendation could potentially become irrelevant to users [9].

2 PROBLEM FORMULATION

If we let $V_{u}\, and\, U_{V}$ are the set of items that user $u \in U$ interacted with and the set of users who interacted with item $v \in V$ respectively with their observed interaction represented by $O = \{o_{u,v}\}$. Each item is associated with a set of attributes, which is represented in the diagram by X_{V}, attributes represented by one-hot vector as well as multi-hot vector.

Given a warm training dataset, D consisting of {U, V, O}, our aim is to learn a cold-start item recommendation model to estimate the probability $\hat{y}_{u.v}$ that a user interacted with an item.

3 METHODOLOGY

3.1 Overview

In order to address the persistent challenges associated with cold-start items in recommendation systems, we propose the implementation of a novel approach that combines Co-occurrence Collaborative Signals with Content Collaborative Filtering. The approach discussed aims to enhance the recommendation accuracy for cold-start items for music recommendation, particularly by addressing the issue of less accuracy due to warm training data. The proposed solution comprises two main components:

1. **Content Collaborative Filtering Module:**

This module is responsible for generating a Content-Based Collaborative Embedding (CBCE). It leverages the item's content characteristics to create an embedding that reflects the inherent features of the item. By doing so, it allows the system to capture the item's essence and characteristics, which is especially beneficial for cold-start items that may lack historical interaction data.

2. **Co-occurrence Collaborative Filtering Module:**

The Co-occurrence CF Module generates a Co-Occurrence Collaborative Embedding (COCE) based on co-occurrence signals i.e., the patterns and relationships of the user-item interaction. It considers how items are frequently used together, capturing patterns and relationships within previous user-item interactions. This co-occurrence information helps to overcome the precision limitations associated with cold-start items by inferring their relevance based on the behavior of users towards other items.

The CCFCRec model introduces a novel collaborative filtering (CF) framework tailored for music recommendation, comprising a Content-Based CF module and a Co-occurrence CF module. In this setup, the Content-Based CF module is responsible for generating content-based collaborative embeddings for training items, while the Co-occurrence CF module focuses on creating co-occurrence collaborative embeddings. Throughout the joint training process of these two CF modules, we apply a contrastive learning approach that fosters knowledge transfer between the collaborative embeddings. This approach effectively refines the collaborative embeddings, particularly during the application phase, ensuring more precise music recommendations. Our proposed model harnesses co-occurrence collaborative signals, especially in the warm training data, to address the issue of ambiguous collaborative embeddings, particularly relevant for making recommendations for cold-start music items. The efficacy of our model is substantiated through comprehensive experiments conducted on real-world music datasets, demonstrating its superior performance in music recommendation.

3.2 Architecture

Figure 1 shows the basic structure of the proposed model. As we can see in figure 1, for the content part of the model, we have a CBCE encoder g_c and a CBCE-based predictor f_q. Similarly, the Co-occurrence part is composed of the COCE encoder g_v and the COCE-based predictor f_v and the UCE (user Collaborative Embedding) encoder g_u is shared between both the CF modules.

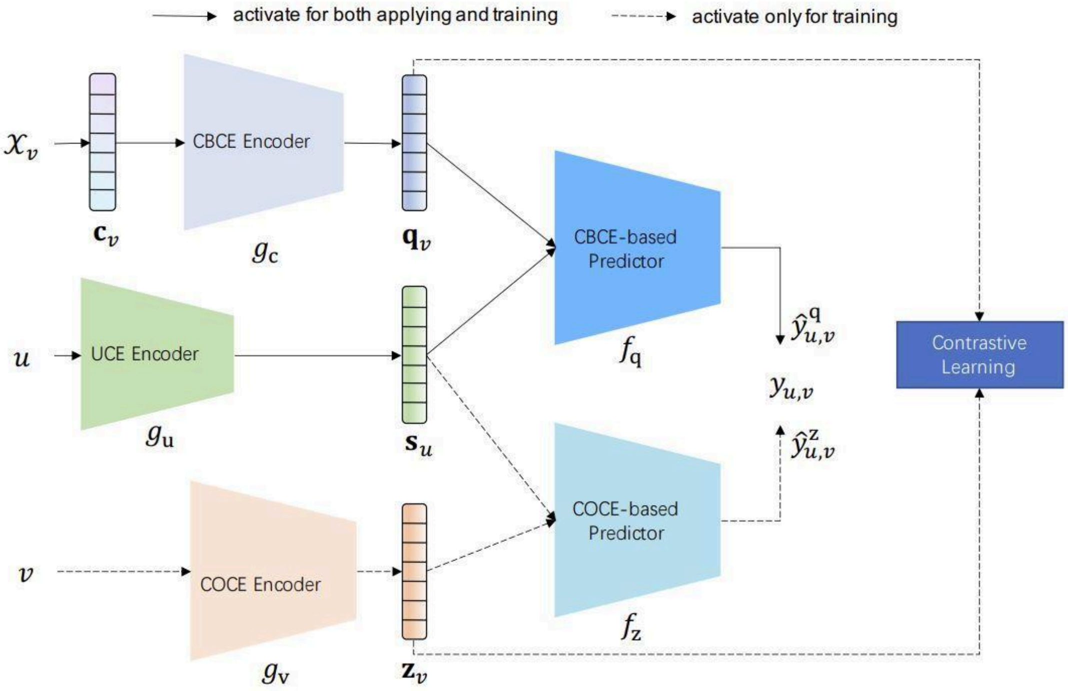

Figure 1. The overview of the proposed model in CCFCRec

Let's look at how the different modules act individually as well as a whole to make appropriate decisions of predicting cold-start items. Consider a training sample (u, v, X_v, $y_{u,v}$), u and v being the user and song respectively, Xv being the attributes of the song and yu,v being the interaction between the user and the song (in this case, ratings) CCFCRec starts with using the encoder g_c to produce the CBCE q_u based on the input content embedding cv, which is an aggregation of the attributes embedding Xv. Simultaneously, CCFCRec also produces the COCE z_v using the encoder gv and UCE s_u using encoder gu. Then the model with move ahead to the prediction part of the interaction $y_{u,v}$. This is done by initializing the predictors f_q and f_z with inputs q_v, s_u and z_v, s_u respectively.

To let the model learn from both the modules, we would transfer the knowledge about the co-occurrence collaborative signals into the content-based module. To do this we would use the concept of contrastive learning, conducted between qv and z_v, by optimizing the objective to maximize the mutual information. Even though we don't have the embedding vector z_v for the cold start item, the content CF will memorize a triple correlation captured by the contrastive learning during the training phase and rectify the blurry or the not-so-accurate qv accordingly. The triple correlation established here is

$c_v \leftrightarrow q_v \leftrightarrow z_v$.

Finally, it's important to highlight that both CF modules are trained together using the same supervision signal $y_{u,v}$, employing a multi-task learning approach. This joint training mechanism guarantees the effective transfer of beneficial co-occurrence collaborative signals from the co-occurrence CF module to the content CF module, and this explains the basic architecture of the model proposed in the paper.

3.3 Loss Function

The loss function, the overall loss function in our case is defined as:

$$L = L_q + L_z + \lambda L_c + ||\ominus||$$

Talking about the individual terms of the overall loss function above, it is composed of 3 loss functions, a $\ominus$ term representing the learnable parameters and λ being the factor used for regularizing the contribution of the loss, L_c.

Now, there are 3 loss functions L_q, L_z, L_c. L_q and L_z being the loss functions associated with the interaction predictions. During the training phase, CCFCRec will make 2 predictions as discussed before. Probability of the interaction between a user u and an item v, invokes these two predictions, CBCE-based predictors f_q and the COCE-based predictors f_z.

$$L_q =- \sum_{(v,u+,u-)\in D} \ln ln\, \sigma(\hat{y}^{q}_{u+,\ v} - \hat{y}^{q}_{u-,\ v})$$

$$L_z =- \sum_{(v,u+,u-)\in D} \ln ln\, \sigma(\hat{y}^{z}_{u+,\ v} - \hat{y}^{z}_{u-,\ v})$$

It is noteworthy that in the training phase, both the CF modules share the UCE and are jointly trained with the same supervision. This helps in offering consistent optimizing objective ensuring the positive transfer of the co-occurrence collaborative signals to the content CF-module.

At-last, the final individual loss, L_c is the contrastive loss. In essence, CBCE captures user preferences for item attributes while COCE captures the collaborative signals from interactions. To address the limitation of blurry CBCE, we employ a novel approach. During training, we train the content CF module to memorize co-occurrence collaborative signals, allowing it to rectify CBCEs for cold-start items during application. This involves conducting contrastive learning between CBCE and COCE, adjusting the CBCE encoder parameters to maximize mutual information between the two item views. So, for this we have applied the concept of contrastive learning to our model and the loss associated with it is given by:

$$L_c =- E_{v\in D,\, v+\in N_v^+}\left[ln \frac{\exp exp\left(\frac{\langle q_v, z_{v^+}\rangle}{\tau}\right)}{\exp exp\left(\frac{\langle q_v, z_{v^+}\rangle}{\tau}\right) + \sum_{v^-\in N_v^-} \exp exp\left(\frac{\langle q_v, z_{v^-}\rangle}{\tau}\right)} \right]$$

We have chosen to use Adam as the optimizer.

4 EXPERIMENTS

The experiments aim to answer the following research questions:

- RQ1 How does the CCFCRec perform in the music domain recommendation system?
- RQ2 How does the CCFCRec in music domain perform as compared to existing method?

4.1 Experimental Settings

4.1.1 Dataset

Dataset	User	Items	Interactions
Yahoo(R2)	1,800,000	136,000	717,000,000

Table 1. Dataset Description

The "R2 - Yahoo" dataset, released by Yahoo! Research Alliance Webscope program, contains of over 717 million ratings of 136 thousand songs by 1.8 million users of Yahoo! Music services collected between 2002 and 2006. User privacy is preserved using randomly assigned numeric IDs for users, songs, artists, and albums. The dataset includes attributes for each song, such as artist, album, and genre, in which genre is comprised of three-level-hierarchy. The dataset is divided into training and test sets. The dataset was chosen as it contains the rating data (from 1-5 scales), and relatively cold-start items (songs rated by at least 1 person) which were the interest of Authors. For simplicity and to reduce computational load, we pre-processed the dataset so that it is sampled down to 2,677 users, 126,841 songs, and 1,026,699 interactions data.

4.1.2 Baseline Method

K-Nearest Neighbor (KNN) is a collaborative filtering technique that makes personalized recommendations by relying on user-item interactions.

4.1.3 Evaluations Protocols

- Hit Ratio (HR@*k*)

Hit Ratio (HR) is used to assess the ability of the proposed method to include relevant items in the top-K recommendations.

$$HR@k = \frac{1}{|D_t|}\sum_{v \in D_t}\frac{\sum_{u\in U_v} \mathrm{I}(rank(u, l_v)\le k}{k}$$

- Normalized Discounted Cumulative Gain (NDCG@*k*)

NDCG is chosen to evaluate the ranking quality of recommended items, considering both relevance and position in the list.

$$NDCG@k = \frac{1}{|D_t|}\sum_{v \in D_t}\frac{1}{|U_v|}\sum_{u\in U_v}\frac{\mathrm{I}(rank(u,l_v)\le k}{loglog\,(1+rank(u))}$$

4.1.4 Parameter Setting

During the training, we set the batch size to 512 and the initial learning rates to 5e-4. Additionally, we formed positive-negative sample pairs by sampling 5 positive samples and 40 negative samples. The value of λ (balance factor of contrastive loss) is set to 0.5. The dimensionality *d* is set to 256.

4.2 Performance Result

We evaluate the performance of the CCFRec model when applied using basic attribute (item's genre) with using extended attributes (artist and album). In Figure 2, the model shows convergence with lower values for both total loss and contrastive loss when utilizing extended attributes. This suggests improved performance associated with the inclusion of artists and album attributes in the model, in contrast to the initial design demonstrated in the original model that solely relied on genres. This aligns with the expectation that incorporating a broader set of features could enhance the model's ability to personalize the recommendation system.

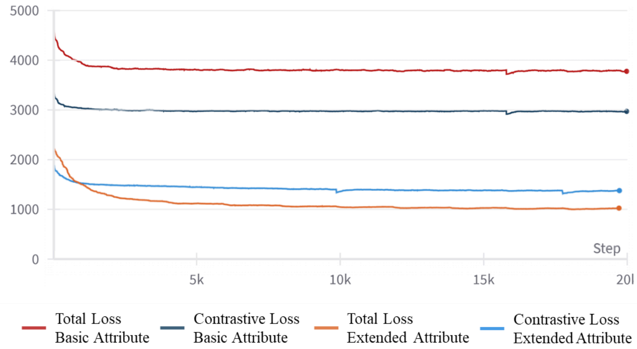

Figure 2. Total and Contrastive Loss with Basic and Extended Features

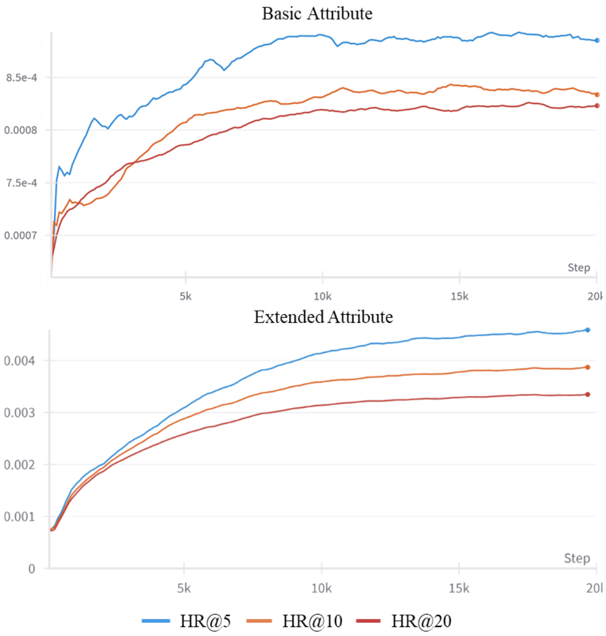

Figure 3. Hit Ratio metrics performances using basic and extended attributes

The Hit Ratio (HR) results further support the effectiveness of incorporating extended attributes into the recommendation model. Figure 3 shows consistently higher values for HR@5, HR@10, and HR@20 when the model utilizes extended attributes, indicating an improved ability to recommend items that users interact with. Furthermore, the model with extended attributes exhibits reduced fluctuation for all values of k (5, 10, and 20), suggesting a more stable and reliable performance compared to the basic one.

4.3 Performance Comparison

In the comparison evaluation of the CCFCRec model against the KNN method, it is observed that the CCFCRec outperforms the baseline in terms of Hit Ratio (HR) across all values of k (5, 10, 20). The higher HR values for CCFCRec was achieved mainly due to the contrastive CF in our model. The contrastive learning provides more information related to the cold-start items to help build and learn a stronger collaborative embeddings, thus, rectifying the blurry collaborative embeddings issue.

However, contrasting to the CCFCRec's superiority in terms of hit ratio, the Normalized Discounted Cumulative Gain (NDCG) metrics are lower across all k values in comparison to KNN. This suggests that while the model excels in including relevant items in the top-k recommendations, the ranking order of these items might not be as optimized, as reflected in the lower NDCG values.

It's worth noting that the main focus of the CCFCRec's model is addressing the cold-start problem, thus optimizing the ranking order might not be the priority of this model. In addition, lower NDCG doesn't necessarily imply a lack of recommendation quality; instead, it may indicate a difference in the distribution of user-favorite items throughout the recommendation list. The model might provide a more evenly distributed set of recommendations, rather than emphasizing a few items at the top of the list.

Dataset	Baseline	HR@5	HR@10	HR@20	NDCG@5	NDCG@10	NDCG@20
Yahoo(R2)	KNN	0.000503	0.001005	0.001068	0.001242	0.003684	0.006534
	CCFCRec	0.004253	0.003656	0.003263	0.000329	0.000372	0.000479

Table 2. Performance comparison

4.4 Limitations and Future Work

The model has some limitations that could be further refined in future study:

1. The model heavily relies on co-occurrence signals, which may be challenging when users engage with individual songs with diverse characteristics, making it difficult to capture meaningful co-occurrence signal. Future studies could explore more attributes in content-based embeddings to enrich the recommendations.
2. The model's dependence on memorized co-occurrence signals during training may limit its adaptability to sudden shifts in music trends. To address this, incorporating adaptive reinforcement learning, where the model learns from changing user behavior, could improve its ability to adapt to evolving trends and preferences.

5 Conclusion

This work extends prior research in addressing the cold-start problem and blurry collaborative embeddings within the context of music recommendation. The model incorporates two key components: the retention of co-occurrence collaborative signals learned during the training phase, which proves effective in mitigating blurry collaborative embeddings, and the application of contrastive learning to overcome the absence of co-occurrence signals in cold-items. The performance results demonstrate superior Hit Ratio values for all considered K values compared to the baseline KNN model. However, the model exhibits lower NDCG values, suggesting room for improvement in optimizing the ranking order of recommendations. It is essential to acknowledge limitations, such as the model's reliance on co-occurrence signals and potential challenges in capturing diverse user preferences. Future work should explore additional attributes for content-based embeddings, implement adaptive reinforcement learning for enhanced adaptability, and address the inherent limitations in co-occurrence signal availability.

REFERENCES

1. Mishra, N., S. Chaturvedi, A. Vij, and S. Tripathi, *Research Problems in Recommender systems.* Journal of Physics: Conference Series, 2021. **1717**.
2. Shewale, R. *44 Netflix Statistics In 2023 (Users, Revenue & Insights)*. 2023 [cited 2023 1 December 2023]; Available from: https://www.demandsage.com/netflix-subscribers/.
3. Willman, C. *Music Streaming Hits Major Milestone as 100,000 Songs are Uploaded Daily to Spotify and Other DSPs*. 2022 [cited 2023 1 December 2023]; Available from: https://variety.com/2022/music/news/new-songs-100000-being-released-every-day-dsps-1235395788/.
4. Volkovs, M., G. Yu, and T. Poutanen, *DropoutNet: addressing cold start in recommender systems*, in *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 2017, Curran Associates Inc.: Long Beach, California, USA. p. 4964–4973.
5. Zhu, Y., et al., *Learning to Warm Up Cold Item Embeddings for Cold-start Recommendation with Meta Scaling and Shifting Networks*, in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2021, Association for Computing Machinery: <conf-loc>, <city>Virtual Event</city>, <country>Canada</country>, </conf-loc>. p. 1167–1176.

6. Sun, C., et al., *LARA: Attribute-to-feature Adversarial Learning for New-item Recommendation*, in *Proceedings of the 13th International Conference on Web Search and Data Mining*. 2020, Association for Computing Machinery: Houston, TX, USA. p. 582–590.
7. Zhou, Z., L. Zhang, and N. Yang, *Contrastive Collaborative Filtering for Cold-Start Item Recommendation*, in *Proceedings of the ACM Web Conference 2023*. 2023, Association for Computing Machinery: Austin, TX, USA. p. 928–937.
8. Pulis, M. and J. Bajada, *Siamese Neural Networks for Content-based Cold-Start Music Recommendation*, in *Proceedings of the 15th ACM Conference on Recommender Systems*. 2021, Association for Computing Machinery: Amsterdam, Netherlands. p. 719–723.
9. Soleymani, M., A. Aljanaki, F. Wiering, and R.C. Veltkamp. *Content-based music recommendation using underlying music preference structure*. in *2015 IEEE International Conference on Multimedia and Expo (ICME)*. 2015.