A Diversity-aware Approach to Bundle Recommendations

Nastaran Ebrahimi[1], Zheying Zhang[1] and Kostas Stefanidis[1]

[1]Tampere University, Tampere, Finland

Abstract

Recommendation systems help users navigate vast amounts of data, with bundle recommendation systems enhancing personalization and customized experience by grouping related items. However, many existing methods overemphasize relevance, leading to repetitive suggestions and user fatigue. This paper introduces two novel bundling methods—Bundle Partition and Bundle Function—designed to balance both diversity and relevance. These methods were evaluated using Amazon datasets on the Appliances, All_Beauty, and Luxury_Beauty categories. Results show a significant increase in diversity, as measured by Intra-List Diversity (ILD), while maintaining high relevance through average ratings. Furthermore, the novelty, assessed via Mean Inverse User Frequency (MIUF), indicates that these methods offer a fresh and relevant experience. These findings emphasize the importance of diversity in enhancing user engagement.

Keywords

Bundle Recommendation Systems, Diversity, Novelty

1. Introduction

In many recommendation contexts, particularly in online shopping and travel package suggestions, users often prefer to purchase a collection of items rather than a single product. Therefore, recommending a set of related items collectively, rather than individually, is more effective. This strategy, known as bundle recommendation, involves suggesting groups of complementary items to enhance decision-making, align with real-world buying behavior, and boost both satisfaction and sales [1].

A significant advancement in recommendation system development is the incorporation of diversification into the recommendation process [2, 3]. While many recommendation systems prioritize accuracy over diversity [4], diversity is crucial in bundle recommendations for offering varied items that meet different customer preferences.

This paper introduces a hybrid bundle recommendation approach that balances relevance and diversity by integrating collaborative and content-based filtering. It predicts user preferences through collaborative filtering and refines recommendations using item features. The approach includes two diversity-aware bundling methods: Bundle Partition, which selects diverse items aligned with user interests; and Bundle Function, which ensures both user relevance and variation among items. Utilizing NLP techniques to calculate item similarities, this method enhances recommendation quality by reducing redundancy.

The proposed methods are evaluated using real-world datasets from Amazon's Appliances, All_Beauty, and Luxury_Beauty, whose extensive metadata, including product descriptions, categories, and user ratings, enabled advanced natural language processing (NLP) analysis [5, 6]. The effectiveness of the bundling approaches was assessed using Intra List Diversity (ILD) and Mean Inverse User Frequency (MIUF). The results showed that both the Bundle Partition and Bundle Function methods successfully introduced diversity, while maintaining relevance.

Overall, the main contributions of this work are : (i) a hybrid model that balances relevance and diversity in bundle recommendations using collaborative filtering and content-based techniques; (ii) the use of NLP to analyze item features, providing more content-rich and diverse bundle recommendations compared to existing user-centric or budget-focused models; (iii) an evaluation of the proposed bundling methods using ILD and MIUF metrics to illustrate the impact of diversity and novelty on user engagement.

DOLAP 2025: 27th International Workshop on Design, Optimization, Languages and Analytical Processing of Big Data, co-located with EDBT/ICDT 2025, March 25, 2025, Barcelona, Spain

nastaranebrahimi2021@gmail.com (N. Ebrahimi); zheying.zhang@tuni.fi (Z. Zhang); konstantinos.stefanidis@tuni.fi (K. Stefanidis)

2. Related Work

Bundle Recommendations. In e-commerce, users often purchase multiple items, making bundle recommendations essential for suggesting sets of products rather than individual ones [1]. Bundle sales serve as a cooperative marketing strategy where multiple brands collaborate to expand their reach and maximize impact [7]. For instance, [8] introduces a model integrating collaborative filtering, demand functions, and price modeling to optimize product selection for revenue maximization. Effective bundle recommendations should prioritize interconnected products, either complementary or alternative, aligning with user preferences [1]. Traditional methods [9, 10] identify frequently bought-together items but often overlook personalization and relevance. Techniques like integer programming [11, 12] fail to capture pairwise dependencies, treating cross-item relationships as rigid constraints, while association analysis [13, 14] applies uniform rules that lack personalization [15].

Diversity in Bundle Recommendations. Diversity and novelty are key to improving recommendation effectiveness [16], with diversity ensuring variation among recommended items [17, 18] and novelty introducing unfamiliar but relevant suggestions [18]. Studies have sought to balance relevance and variety, with [11] and [19] proposing the Bundle Generation Network (BGN), which leverages Determinantal Point Processes (DPPs) to enhance diversity in bundle recommendations.

This article presents techniques for generating diverse and relevant bundles, evaluated using metrics like ILD and MIUF. Unlike approaches that balance relevance and diversity with budget constraints—potentially compromising efficiency or diversity—this method optimizes novelty and user satisfaction while reducing computational costs. It achieves this through dynamic similarity-based bundling, randomized partitions, and strategic item selection.

3. Bundling Methodology

In this paper, we address the challenge of creating product bundles that balance diversity and relevance based on user preferences, aiming to reduce redundancy and enhance the user experience. we propose a hybrid approach to ensure both relevance and diversity while keeping relevant items within each bundle. The method combines collaborative filtering using SVD to identify user interests and a content-based approach to select suitable items for each bundle.

Determining User Preferences. First, we identify user preferences to ensure bundles align with individual interests. The SVD algorithm is used to generate personalized recommendations based on interaction data, and the top-rated item is selected as the "target" item for bundling. Bundling involves selecting items that not only align with user preferences and but also add value through diversity. After choosing the target item, additional items are selected based on distinct features to ensure variety and relatedness, aiming to create a well-rounded bundle that avoids redundancy and enhances user satisfaction. For example, if a user's top-rated item is a smartphone, the bundle may include a phone case, screen protector, or wireless earbuds. These items are selected based on the user's interest in technology (from the SVD-based analysis) and are diverse enough to offer a broader experience. This prevents repetition and ensures each item adds value in a different way. The process uses content-based filtering to assess item features, maximizing diversity within the bundle.

Computing Similarities. Once the target item t is identified, the next step is to locate items that share similar features to enhance recommendation relevance and user satisfaction. Each item i in the dataset can be represented by a feature vector $\mathbf{f}_i = [f_{i1}, f_{i2}, \ldots, f_{in}]$, where each f_{ij} represents a specific feature, such as brand or category. The similarity between the target item t and another item i is calculated using a similarity function, $\mathrm{sim}(\mathbf{f}_t, \mathbf{f}_i)$. Items with the highest similarity scores are chosen, ensuring that the recommendations align closely with user preferences.

Finding similar items is crucial for creating effective product bundles, as it ensures relevance and increases the likelihood of high user ratings, enhancing engagement and satisfaction. To calculate item similarity, metadata is processed and vectorized using NLP techniques like TF-IDF, which converts text into numerical vectors, assigning greater importance to key terms. Cosine similarity is then used to measure the similarity between items by calculating the cosine of the angle between their feature vectors. Items with high similarity scores are considered closely related to the target item and are selected as potential recommendations, ensuring relevance and higher user satisfaction.

4. Bundle Generation

This section outlines three methods for forming product bundles. The first method focuses on item similarity, grouping highly similar items with user preferences without considering diversity. The second method introduces diversity by selecting a mix of related but varied items, ensuring a balance between relevance and diversity using pairwise dissimilarity and randomization. The third method aims to maximize intra-bundle diversity by choosing items that differ from both the target item and each other, providing a broader set of recommendations to enhance user experience.

Similarity-based Bundling. The similarity-only bundling method creates product bundles based on items similar to those previously liked by the user, assuming similar items will be well-received. This method serves as the base model to compare with two diversity-aware models. The algorithm generates a list of items similar to the target item, identified from the user's preferences, by comparing it with other items in the metadata. While this approach ensures relevance, the resulting bundles may lack variety, leading to repetitive suggestions. Despite this, it provides a useful baseline for comparing more diverse bundling strategies.

Partition and Randomization Method. Formally, let T represent the target item, and let $S = \{s_1, s_2, \ldots, s_n\}$ be the set of items similar to T, determined based on the cosine similarity. The objective is to find items within S that maximize dissimilarity to T and place them in a list L: $L = \{s_i \in S \mid \text{maximize dissimilarity}(s_i, T)\}$. This way, we ensure that the final bundle includes varied items that still reflect the user's preferences, reducing redundancy.

After constructing the list L, the items are shuffled and divided into partitions. A random selection is made from each partition to add unpredictability, increasing novelty while maintaining relevance. This approach guarantees a fresh combination of items for each bundle, resulting in a dynamic and engaging recommendation process.

To measure diversity, TF-IDF vectors of item features are used, and Euclidean distance serves as the metric. This helps maintain a balance between similarity and diversity in the item list. For two points $\mathbf{P} = (p_1, p_2, \ldots, p_n)$ and $\mathbf{Q} = (q_1, q_2, \ldots, q_n)$ in a 2-dimensional space, the Euclidean distance d is: $d(\mathbf{P}, \mathbf{Q}) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2}$. Euclidean distance helps identify items that are both relevant and diverse. This is an ideal measure in this work due to its simplicity and effectiveness in distinguishing diverse items, especially when using data like TF-IDF vectors. After generating the list, the items are shuffled and divided into segments. One item is randomly selected from each segment: $B = \{b_i \mid b_i \in \mathrm{random}(\mathrm{segment}_i)\}$. This approach ensures diversity and unpredictability in the final bundle.

Algorithm 1 begins by retrieving and filtering texts of similar items, then computes TF-IDF vectors for these texts and the target item. It calculates Euclidean distances between the target item and similar items, shuffles the list of similar items, and divides it into partitions. One item is randomly selected from each partition to form bundles, with the first bundle including the target item.

Bundle Function Method. The Bundle Function method aims to curate bundles by strategically selecting items that are distinct from each other while still aligning with user preferences. Items similar to the target item T are identified from a pre-constructed list $S = \{s_1, s_2, \ldots, s_n\}$, using precomputed similarities, like cosine similarity. The goal is to create a bundle from list S, ensuring each successive item is as dissimilar as possible to previously selected items. This is achieved by calculating Euclidean distances between their feature vectors. Let $\mathbf{f}_i$ and $\mathbf{f}_j$ represent feature vectors of items s_i and s_j in S. The Euclidean distance $d(\mathbf{f}_i, \mathbf{f}_j)$ is given by: $d(\mathbf{f}_i, \mathbf{f}_j) = \sqrt{\sum_{k=1}^{n} (f_{ik} - f_{jk})^2}$. The algorithm selects items with the largest Euclidean distances to ensure variety: $B = \{s_1, s_2, \ldots, s_m\} \mid \text{maximize } d(\mathbf{f}_i, \mathbf{f}_j), \forall i, j \in B, i \neq j$. This selection ensures that the items within the bundle are not just variations of the same product, but instead represent diverse choices that cater to user preferences.

Algorithm 2 analyzes target items and finds similar ones

Algorithm 1 Partition and Randomization Method

Retrieve and Filter Texts
for each ASIN in *top_similar_items* **do**
    if ASIN exists in *subset_data* **then**
        Retrieve its text and store it.
    end if
end for
Compute TF-IDF Vectors for target item's text
Compute TF-IDF Vectors for all items' text
for each ASIN in *top_similar_items* **do**
    Calculate Euclidean Distances with target item
end for
Sort ASINs by distance in descending order.
Shuffle and Partition sorted ASINs list
for each bundle (10 total bundles) to select items **do**
    if it is the first bundle **then**
        Make 4 partitions to select items (since the target item is included)
    else
        Make 5 partitions to select items
    end if
    Randomly select one ASIN from each partition
    Create the bundle with the selected ASINs
end for
return list of bundles

Algorithm 2 Bundle Function (form_bundle) Method

Initialization
Convert T and M to dense arrays if necessary
Create a dictionary *asin_idx* mapping ASINs to their indices in D
Map S to their indices in D, resulting in *S_idx*
Create an empty list *bundles*
for each $b \in \{1, 2, \ldots, B\}$ **do**
    Start the bundle with the most similar item, *bundle_idx* $\leftarrow [S\_idx[0]]$
    Remove the first item from *S_idx*, $S\_idx \leftarrow S\_idx[1:]$
end for
while $|bundle\_idx| < n$ and $S\_idx \neq \emptyset$ **do**
    Set $last\_idx \leftarrow bundle\_idx[-1]$
    Retrieve $v_{last} \leftarrow M[last\_idx]$
    Compute Euclidean distances: $dists \leftarrow [\text{euclidean}(v_{last}, M[idx]) \ \forall \ idx \in S\_idx]$
    Identify index of max distance: $max\_dist\_idx \leftarrow \arg\max(dists)$
    Add $S\_idx[max\_dist\_idx]$ to *bundle_idx* and remove it from *S_idx*
end while
Convert *bundle_idx* to ASINs using D and append to *bundles*
return *bundles*

using a precomputed list. Euclidean distance introduces diversity by selecting items further apart. The algorithm iteratively adds distinct items until desired number is reached, creating multiple bundles with different starting points for diverse yet relevant content.

5. Experimental results

The 2018 Amazon dataset provides rich user-item interactions and detailed metadata, making it valuable for recommendation systems. This study used two files: "ratings only" and "metadata." The "ratings only" file contains items, users, ratings, and timestamps, with 371,345, 5,722,988, and 602,777 ratings for the All_Beauty, Luxury_Beauty, and Appliances categories, respectively. The metadata file includes product information like title, features, description, price, brand, and category. The All_Beauty and Luxury_Beauty categories, with 32,992 and 12,308 products, were combined as the beauty dataset, while the Appliances dataset, with 30,459 products, was also analyzed.

Table 1
ILD scores in bundles for each method

Bundles	Beauty Dataset			Appliances Dataset		
	Partition	Function	Similarity	Partition	Function	Similarity
Bundle1	0.93	0.90	0.43	0.93	0.90	0.43
Bundle2	0.96	0.98	0.60	0.96	0.98	0.60
Bundle3	0.98	0.97	0.33	0.98	0.97	0.33
Bundle4	0.94	0.98	0.38	0.94	0.98	0.38
Bundle5	0.97	0.98	0.42	0.97	0.98	0.42
Bundle6	0.95	0.99	0.52	0.95	0.99	0.52
Bundle7	0.97	0.98	0.52	0.97	0.98	0.52
Bundle8	0.96	0.99	0.52	0.96	0.99	0.52
Bundle9	0.96	0.87	0.53	0.96	0.87	0.53
Bundle10	0.95	0.98	0.65	0.95	0.98	0.65
Mean	0.96	0.96	0.50	0.96	0.96	0.50

Evaluating Diversity. In experiments, the Intra-List Diversity (ILD) metric [17] is calculated for each bundle generated using one of the three proposed methods. By evaluating ILD scores across different bundling techniques, we aim to determine how each method impacts diversity in recommendations. ILD is defined as the average pairwise distance between items within a set of recommended items. Formally, ILD $= \frac{1}{|R|(|R|-1)} \sum_{i \in R} \sum_{j \in R} d(i, j)$, where $|R|$ represents the number of items in the recommendation set R, and $d(i, j)$ is the distance between two items i and j within the set. ILD is flexible, as the distance measure $d(i, j)$ can be defined in various ways based on the recommendation system's context and requirements. We use cosine similarity to calculate $d(i, j)$, defined as the complement of similarity, $1 - \text{sim}(i, j)$.

In Table 1, you can see the improvement in diversity by the use of the methods Bundle Partition and Bundle Function for beauty and appliances. The Bundle Partition method consistently shows high ILD values ranging from 0.93 to 0.98, with an average of 0.96, indicating that it effectively introduces diversity and prevents redundancy in the bundles. Similarly, the Bundle Function method achieves high ILD scores ranging from 0.87 to 0.99, with an identical average of 0.96, suggesting that both methods are equally effective in ensuring item diversity and enhancing user engagement.

Evaluating Relevance. The Average Rating (AVGr) represents the mean of ratings for items within a bundle. In recommendation systems, each item receives a rating, either from user feedback or predictive algorithms. Based on the idea in [20], we propose using item ratings as a relevance score to improve the accuracy of collaborative filtering recommendations by predicting user preferences. The average rating helps assess the overall quality or appeal of the items within the bundle. It is calculated as: AVG $= \frac{1}{n} \sum_{i=1}^{n} r_i$, where n is the number of items in the bundle, and r_i is the rating of item i in the bundle. A higher average rating indicates that users generally like the items, suggesting the bundle's likely success, while a lower average may imply less appeal. Figures 1 and 2 show that both methods maintain high relevance scores.

Variance of Ratings (VAR) measures the spread of ratings

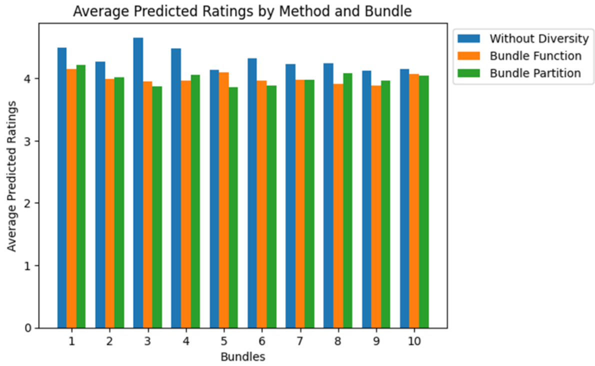

Figure 1: AVGr for appliances datasets.

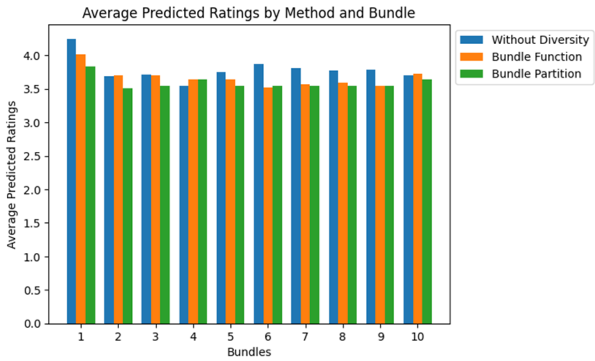

Figure 2: AVGr for bundles for beauty datasets.

Table 2
VAR scores for each method

Bundles	**Appliances Dataset**		**Beauty Dataset**	
	Partition	**Function**	**Partition**	**Function**
Bundle1	0.18	0.19	0.33	0.36
Bundle2	0.08	0.08	0.11	0.09
Bundle3	0.16	0.07	0.00	0.11
Bundle4	0.05	0.04	0.03	0.09
Bundle5	0.08	0.06	0.00	0.03
Bundle6	0.06	0.04	0.00	0.03
Bundle7	0.04	0.04	0.00	0.00
Bundle8	0.11	0.01	0.00	0.00
Bundle9	0.00	0.01	0.00	0.00
Bundle10	0.00	0.15	0.00	0.00

within a bundle, offering insight into the consistency of user satisfaction. Unlike AVGr, which provides an overall sense of the bundle's appeal, VAR indicates how much ratings vary from the average rating. It is calculated as: $\text{VAR} = \frac{1}{n}\sum_{i=1}^{n}(r_i - \text{AVG})^2$, where n is the number of items in the bundle, r_i is the rating of item i in the bundle, and AVG is the average rating of the bundle. A low variance indicates consistent ratings across the bundle, suggesting a uniform user experience. This can be advantageous when aiming for a consistent level of quality or satisfaction. Although the Similarity-based method achieves slightly higher relevance in Figures 1 and 2, the Partition and Function methods offer a better balance between relevance and diversity, resulting in more engaging bundles. The low VAR scores in Table 2 indicate that increasing diversity does not compromise relevance or perceived quality.

Evaluating Novelty. The Global Long-Tail Novelty [21] is used to determine how novel an item is by assessing its

Table 3
MIUF Scores for each method

Bundles	**Appliances Dataset**		**Beauty Dataset**	
	Partition	**Function**	**Partition**	**Function**
Bundle1	16.64	15.50	13.33	15.24
Bundle2	13.00	16.98	15.28	15.52
Bundle3	18.54	17.16	17.12	14.86
Bundle4	17.02	16.98	15.21	15.84
Bundle5	18.54	17.44	12.15	16.96
Bundle6	18.54	17.19	16.76	14.92
Bundle7	18.54	17.01	15.92	16.88
Bundle8	17.00	18.54	15.22	15.51
Bundle9	17.01	17.39	18.25	16.29
Bundle10	17.51	17.34	16.23	15.11

popularity among a broad audience. Items in the "long tail" of the popularity distribution are considered novel, meaning they are not widely known. This concept uses Inverse User Frequency (IUF) to measure an item's rarity among users, similar to inverse document frequency (IDF). IUF is defined as: $\text{IUF} = -\log_2\left(\frac{|U_i|}{|U|}\right)$, where $|U_i|$ is the number of users interacted with i, and $|U|$ the number of users in the system. For the average novelty of the recommended items, the Mean Inverse User Frequency (MIUF) is calculated by averaging the IUF values of all items in the recommendation set: $\text{MIUF} = -\frac{1}{|R|}\sum_{i\in R}\log_2\left(\frac{|U_i|}{|U|}\right)$, where R is the set of recommended items.

The MIUF score measures the novelty of recommended items by assessing how uncommon they are across the user base. To evaluate novelty, its distribution is analyzed, revealing that 90% of items have a MIUF below 13.33. All bundles generated by both methods meet or exceed this threshold, indicating their relative novelty. The Bundle Partition method achieves MIUF values between 13.00 and 18.54, demonstrating significant novelty, while the Bundle Function method shows even stronger novelty with MIUF values ranging from 15.50 to 18.54. Table 3 presents the novelty scores for each dataset.

Discussion. Bundle Partition and Bundle Function enhance the recommendations' diversity and novelty while maintaining high relevance. ILD scores for both methods are high in All_Beauty and Luxury_Beauty (0.93-0.98) and slightly lower in Appliances (0.55-0.98), indicating varied recommendations. For relevance, AVGr and VAR demonstrate users' preferences alignment, with AVGr scores up to 4.0. Novelty, measured by MIUF, is highest for the Bundle Function method, showing that the methods provide user-relevant bundles that outperform the Similarity-Based approach.

6. Summary

In this paper, we design, develop, and evaluate two bundling methods, alongside a baseline solution. Our goal is to improve product bundle recommendations by balancing relevance and diversity. We implemented a hybrid approach combining collaborative and content-based filtering, using NLP to analyze item features. Both methods successfully introduced diversity without sacrificing relevance, achieving promising results in maintaining high ratings and enhancing the overall diversity of recommendations.

References

[1] L. Mia, X. Bao, L. Chang, Z. Xu, L. Li, A survey of researches on personalized bundle recommendation techniques, in: Hongyang, Y. Qiben, Z. X. C. Xiaofeng, Yan (Eds.), Machine Learning for Cyber Security, Springer International Publishing, 2020, pp. 290–304.

[2] M. Kyriakidi, K. Stefanidis, Y. Ioannidis, On achieving diversity in recommender systems, in: Proceedings of the ExploreDB 2017, Association for Computing Machinery, Inc, 2017. doi:10.1145/3077331.3077341.

[3] E. Ntoutsi, K. Stefanidis, K. Rausch, H. Kriegel, "strength lies in differences": Diversifying friends for recommendations through subspace clustering, in: Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, CIKM 2014, Shanghai, China, November 3-7, 2014, ACM, 2014, pp. 729–738.

[4] D. M. Fleder, K. Hosanagar, Recommender systems and their impact on sales diversity, EC'07 - Proceedings of the Eighth Annual Conference on Electronic Commerce (2007) 192–199. URL: https://dl.acm.org/doi/10.1145/1250910.1250939. doi:10.1145/1250910.1250939.

[5] V. Christophides, V. Efthymiou, K. Stefanidis, Entity Resolution in the Web of Data, Synthesis Lectures on the Semantic Web: Theory and Technology, Morgan & Claypool Publishers, 2015. doi:10.2200/S00655ED1V01Y201507WBE013.

[6] T. B. Araújo, K. Stefanidis, C. E. S. Pires, J. Nummenmaa, T. P. da Nóbrega, Incremental blocking for entity resolution over web streaming data, in: 2019 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2019, Thessaloniki, Greece, October 14-17, 2019, 2019, pp. 332–336.

[7] M. E. Drumwright, A demonstration of anomalies in evaluations of bundling, Marketing Letters 3 (1992) 311–321. URL: https://doi.org/10.1007/BF00993916. doi:10.1007/BF00993916.

[8] M. Beladev, L. Rokach, B. Shapira, Recommender systems for product bundling, Knowledge-Based Systems 111 (2016) 193–206. doi:10.1016/j.knosys.2016.08.013.

[9] Y. L. Chen, K. Tang, R. J. Shen, Y. H. Hu, Market basket analysis in a multiple store environment, Decision Support Systems 40 (2005) 339–354. doi:10.1016/J.DSS.2004.04.009.

[10] R. Garfinkel, R. Gopal, A. Tripathi, F. Yin, Design of a shopbot and recommender system for bundle purchases, Decision Support Systems 42 (2006) 1974–1986. doi:10.1016/J.DSS.2006.05.005.

[11] M. Xie, L. V. Lakshmanan, P. T. Wood, Breaking out of the box of recommendations: From items to packages, RecSys'10 - Proceedings of the 4th ACM Conference on Recommender Systems (2010) 151–158. URL: https://dl.acm.org/doi/10.1145/1864708.1864739. doi:10.1145/1864708.1864739/SUPPL_FILE/RECSYS2010-29092010-08-01.MOV.

[12] Q. Liu, Y. Ge, Z. Li, E. Chen, H. Xiong, Personalized travel package recommendation, in: Proceedings - IEEE International Conference on Data Mining, ICDM, 2011, pp. 407–416. doi:10.1109/ICDM.2011.118.

[13] T. C. Yang, H. Lai, Comparison of product bundling strategies on different online shopping behaviors, Electronic Commerce Research and Applications 5 (2006) 295–304. doi:10.1016/j.elerap.2006.04.006.

[14] V. L. Miguéis, A. S. Camanho, J. F. E. Cunha, Customer data mining for lifestyle segmentation, Expert Systems with Applications 39 (2012) 9359–9366. doi:10.1016/j.eswa.2012.02.133.

[15] D. Le, H. W. Lauw, Y. Fang, Basket-sensitive personalized item recommendation, in: C. Sierra (Ed.), Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017, ijcai.org, 2017, pp. 2060–2066. URL: https://doi.org/10.24963/ijcai.2017/286. doi:10.24963/IJCAI.2017/286.

[16] S. Vargas, P. Castells, Rank and relevance in novelty and diversity metrics for recommender systems, in: Proceedings of the fifth ACM conference on Recommender systems, 2011, pp. 109–116.

[17] C.-N. Ziegler, S. M. McNee, J. A. Konstan, G. Lausen, Improving recommendation lists through topic diversification, in: Proceedings of the 14th international conference on World Wide Web, 2005, pp. 22–32.

[18] P. Castells, N. Hurley, S. Vargas, Novelty and diversity in recommender systems, in: Recommender systems handbook, Springer, 2021, pp. 603–646.

[19] J. Bai, C. Zhou, J. Song, X. Qu, W. An, Z. Li, J. Gao, Personalized bundle list recommendation, in: The World Wide Web Conference, 2019, pp. 60–71.

[20] Z.-P. Zhang, Y. Kudo, T. Murai, Y.-G. Ren, Enhancing recommendation accuracy of item-based collaborative filtering via item-variance weighting, Applied Sciences 9 (2019) 1928.

[21] T. Zhou, Z. Kuscsik, J.-G. Liu, M. Medo, J. R. Wakeling, Y.-C. Zhang, Solving the apparent diversity-accuracy dilemma of recommender systems, Proceedings of the National Academy of Sciences 107 (2010) 4511–4515.