Article

Extrapolative Collaborative Filtering Recommendation System with Word2Vec for Purchased Product for SMEs

Kyoung Jun Lee ¹, Yujeong Hwangbo ^{2,*} , Baek Jeong ¹, Jiwoong Yoo ³ and Kyung Yang Park ⁴

¹ Department of Big Data Analytics, Kyung Hee University, Seoul 02447, Korea; klee@khu.ac.kr (K.J.L.); ylbak@khu.ac.kr (B.J.)

² Department of Social Network Science, Kyung Hee University, Seoul 02447, Korea

³ AI & BM Lab, Seoul 02449, Korea; wldnd4011@gmail.com

⁴ Harex InfoTech, Seoul 04625, Korea; park@ubpay.com

* Correspondence: hwangbo@khu.ac.kr; Tel.: +82-2-961-0490

Abstract: Many small and medium enterprises (SMEs) want to introduce recommendation services to boost sales, but they need to have sufficient amounts of data to introduce these recommendation services. This study proposes an extrapolative collaborative filtering (ECF) system that does not directly share data among SMEs but improves recommendation performance for small and medium-sized companies that lack data through the extrapolation of data, which can provide a magical experience to users. Previously, recommendations were made utilizing only data generated by the merchant itself, so it was impossible to recommend goods to new users. However, our ECF system provides appropriate recommendations to new users as well as existing users based on privacy-preserved payment transaction data. To accomplish this, PP2Vec using Word2Vec was developed by utilizing purchase information only, excluding personal information from payment company data. We then compared the performances of single-merchant models and multi-merchant models. For the merchants with more data than SMEs, the performance of the single-merchant model was higher, while for the SME merchants with fewer data, the multi-merchant model's performance was higher. The ECF System proposed in this study is more suitable for the real-world business environment because it does not directly share data among companies. Our study shows that AI (artificial intelligence) technology can contribute to the sustainability and viability of economic systems by providing high-performance recommendation capability, especially for small and medium-sized enterprises and start-ups.

Keywords: extrapolative collaborative filtering; multi-merchant; recommendation system; Word2Vec

Citation: Lee, K.J.; Hwangbo, Y.; Jeong, B.; Yoo, J.; Park, K.Y. Extrapolative Collaborative Filtering Recommendation System with Word2Vec for Purchased Product for SMEs. *Sustainability* **2021**, *13*, 7156. <https://doi.org/10.3390/su13137156>

Academic Editor: Miltiadis D. Lytras

Received: 10 May 2021

Accepted: 21 June 2021

Published: 25 June 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

A good recommendation system provides magical experiences for users via relevant and serendipitous recommendations of products or services. Users who receive appropriate recommendations become more loyal to the company, and continuous purchases lead to increased sales. Various recommendation studies have been conducted in which the recommendation system learns past patterns of customer purchases. Recently, recommendation studies based on natural language processing technology have emerged [1]. Large companies such as Amazon, Alibaba, and eBay have been increasing sales via recommendation services [2,3]. However, not all companies can introduce recommendation services. Small and medium-sized enterprises (SMEs) have insufficient resources in all areas, including technology, funds, and manpower, so there is a limit to their transformation for environmental change [4]. In Korea, SMEs are classified by two criteria: sales and total assets according to the Framework Act on Small and Medium Enterprises. Although the standard for each industry is different, because the size of sales is different depending on the industry, the total amount of assets is the same in all industries and refers to a small company with less than 500 million dollars. Because of the lack of data and lack

of AI professionals, it is not easy to develop recommendation systems directly unless the company is large. It is not easy for SMEs to introduce recommendation systems [5] because high-quality data are needed to build recommendation systems, and a lot of time and money is spent to collect such data [6]. In addition, start-ups without a customer base have difficulty entering new markets. As the gap widens day by day, the focus on large companies is gradually maximizing. Therefore, we need to develop AI technology and policy support for small and medium-sized companies and start-ups.

Each small or medium-sized company also wants to increase sales by recommendation services but often does not have enough data. The more data, the better the performance. Small and medium-sized enterprises want to obtain data from other merchants and increase their recommendation performance. However, sharing raw data directly among companies is not easy because of both business concerns and legal concerns. To solve this problem, this study aims to propose a system that does not directly share data but can improve recommendation performance for small and medium-sized enterprises that lack data and technology. We expect this recommendation system to enable small and medium-sized enterprises and start-ups to pursue sustainable development.

In this study, we applied Word2Vec to purchased product (PP) data. The PP2Vec was developed by utilizing payment data written in natural language without the user's demographic information. Through the PP2Vec algorithm, SMEs lacking data will be able to provide appropriate recommendation services to users.

2. Related Work

2.1. Recommendation System

A recommendation system is a type of information filtering (IF) technology that filters information based on user preferences. Recommendation systems are widely studied and applied in many industries because they provide helpful information to users [7,8]. These recommendation systems are used in various areas, including Amazon, YouTube, Netflix, and Spotify [3,9,10]. The most usual recommendation systems include collaborative filtering and content-based filtering, and collaborative filtering techniques are generally known to be effective in recommending items [11]. Basically, as a type of collaborative filtering, user-based recommendations measure similarity between users, and item-based recommendation systems are based on the similarity between items [12,13]. In order to utilize the recommendation system, sufficient data must be collected, and a hybrid filtering methodology using both user-based and item-based recommendation techniques has been developed [14,15].

2.2. Word2Vec-Based Recommendation System

Word2Vec, a natural language processing method, was proposed by Google in 2013 [16]. It is a technique for representing words as vectors by embedding words into vector spaces. These methods show excellent performance in natural language processing and have been used in many studies, including those on recommendation systems [17,18]. Item2Vec-based recommendation services are implemented by introducing Word2Vec into Item-Based CF. Item2Vec is able to analyze the relationships between items without user information [19]. In [20], based on users' visit history, the next place to visit was recommended. Product2Vec [21] enhances marketing by using shopping baskets as input. Similarly, the recommendation algorithms we propose also leverage users' payment history to build PP2Vec to make appropriate recommendations.

2.3. Multi-Merchant

Cross-domain collaborative filtering has emerged as a way to solve the cold-start problem [17,22]. It focuses heavily on patterns of evaluation scores learned in the auxiliary domain (secondary domain) or the transfer of latent factors to the target domain. The methodology utilizes data from other domains to improve the accuracy of the target domain. The underlying assumptions of existing cross-domain studies include that data

from different domains are shared, which is an unrealistic method for real-world businesses. Sharing customer data poses a risk of leakage of user privacy, so sharing customer data directly among different domains is challenging in business due to privacy concerns [20,23].

Although there has also been research on cross-domain recommendation methodologies that protect privacy [20], these are different from the methods proposed in this study by the passing of the weight of learned items to the target domain. The ECF that we propose in this study does not share the personal information of customers or the weight of learned models. These methods do not share data in business situations, making them more realistic models.

3. Motivation

Extrapolative Collaborative Filtering System Scenario for a Merchant

Merchant S is a startup that has just opened, and most of its customers are new customers. There are not enough data about its customers yet, so there is no proper way to recommend products to customers. Merchant S has signed up for the recommendation service. One day, a new customer, A, visits Merchant S. Merchant S is able to find similar customers to A among existing Merchant S customers by using the recommendation service, and using the information at this time, it is able to recommend products to A.

SMEs have difficulty in providing recommendation services because they have a small user base and insufficient budget for developing AI models. However, the recommendation service entity should solve privacy issues to enable direct data sharing between merchants. In our system, we assume that for the exploratory collaborative filtering, the recommendation service uses only payment transaction data without user demographic data. Extrapolation is a method of estimating unknown areas beyond the data obtained from past experiments. Herein, we develop an extrapolative collaborative filtering methodology that provides recommendation services by utilizing the user behavior data present in multiple domains, such as user purchases and content viewing history, without utilizing user demographics [24].

The different merchants do not share their internal product codes, while the user's purchase and payment history (e.g., receipts) is stored in natural language for the user's convenience in a refund or account management book. We then apply Word2Vec to analyze the purchase and payment data written in natural language. The Word2Vec algorithm represents each user as a vector by utilizing user-purchase payment data and explores similar users through these vectors. We name the new Word2Vec as PP2Vec, as it learns the purchasing propensity of users. By utilizing the learned purchasing tendency, users can receive appropriate recommendations even when they visit a new merchant.

We aim to confirm whether we could provide an appropriate recommendation service in an e-commerce environment using only the minimal payment transaction data. The recommendation service entity uses only the payment data from all merchants. To verify this methodology, two recommendation situations were defined, and the accuracy of the recommendation models in each situation was compared. The single-merchant method, shown in Figure 1, is only able to know the purchase information of users of that one merchant. The multi-merchant case provides recommendation services to users by reflecting purchase information from various merchants.

Figure 1. Single-merchant and multi-merchant methods.

The recommendation system in the single-merchant environment cannot recommend products when User N first visits Merchant A. User A, who has used Merchant A, visits Merchant A again after a long time. Merchant A is able to recommend to User A, a product recently purchased by User B, who has a purchase history similar to that of User A. However, this system could not make a recommendation to User N, a first-time visitor to Merchant A, because there is no purchase history.

On the other hand, the recommendation system in the multi-merchant environment can recommend products even when User N visits Merchant A for the first time. Since User N has a purchase history with Merchant B, User C with a similar purchase history from Merchant B could be searched for. In addition, a product that User C recently purchased from Merchant A may be recommended to User N.

4. Method

4.1. Data Description

To evaluate the performance of ECF with PP2Vec, we utilized the purchase transaction data of four Korean merchants [25]. The released data contain the four merchants' sales data and consist of details of products purchased from 2014 to 2015. However, Merchant S ('the smallest') only has data for 2015. A total of 28,592,566 purchases were used for analysis after pre-preprocessing, with 19,335 users and 4386 product types. As shown in Table 1, Merchant L1 ('the Largest') has the largest number of purchases, and the average number of purchases per user is about 719. Merchant S has the smallest number of purchases, i.e., 105,402, and the lowest number of users at 3791. Compared with the other merchants, Merchant S has 145 product types, the least.

Table 1. Data by Merchant Type.

	Merchant L1	Merchant L2	Merchant M	Merchant S	Total
# of Purchases	13,337,881	9,379,184	5,770,099	105,402	28,592,566
# of Users	18,547	17,097	19,113	3791	19,335
Product Types	2624	987	630	145	4386
Avg. Purchases Per User	719.14	548.59	301.89	27.80	1478.80

4.2. Extrapolative Collaborative Filtering (ECF)

Semantic similarities between language items can be quantified and classified based on the distribution properties of data in large-scale linguistic data. Words appearing in the same context tend to have similar meanings and the same tendency [26]. We validated the ECF system in two steps. First, we developed PP2Vec (purchased product to vector) by utilizing the users' transaction data (Figure 2). PP2Vec is composed of PP2Vec on user, which considers the purchasing tendency of users, and PP2Vec on product, which considers the patterns of products being purchased together. In PP2Vec on user, similar users are searched by considering three things: the purchased products, the purchase locations, and the purchase times. Second, we compared the recommendation results of the single-merchant and multi-merchant methods using the developed PP2Vec.

Figure 2. Purchased Product to Vector Architecture.

4.2.1. Word2Vec-Based Hybrid Collaborative Filtering (Hybrid CF)

We implemented a recommendation system by constructing a Hybrid CF model that combines user-based CF and item-based CF using the Word2Vec algorithm. PP2Vec on user operates as a user-based CF, and PP2Vec on product operates as an item-based CF. The final PP2Vec combining the two vectors operates as a hybrid CF model. This model excludes the demographic information of the user. In order to obtain the tendency vector of each user, we used the Gensim library, and the product vector, place vector, and time vector created through the Gensim library are denoted V_P , V_L , and V_T , respectively.

The definition of PP2Vec on user is as follows:

$$V_{PP_{User}} = V_P + V_L + V_T \quad (1)$$

The recommendation system proceeds in two steps (Algorithm 1). First, when a user u_r is input, the recommendation system searches for the most similar M_I through the PP2Vec calculated previously and recommends the products that the searched user mainly purchased. Next, to reflect user u_r 's recent purchase tendency, the recommendation system recommends a similar product to the product last purchased by the user u_r . By combining the products recommended in these two steps, the recommendation system finally recommends the top-N products.

Algorithm 1 Hybrid Collaborative Filtering based Word2Vec

Input: User ID u_r

User Metrics Based on Product of Purchased Product M_P

User Metrics Based on Location of Purchased Product M_L

User Metrics Based on Time of Purchased Product M_T

Product Metrics Based on User M_I

Output: Top-N Recommended Product List

Calculate Product Vector V_P using $Gensim(M_P)$

Calculate Location Vector V_L using $Gensim(M_L)$

Calculate Time Vector V_T using $Gensim(M_T)$

Calculate Purchased Product Vector $V_{PP_{User}} = V_P + V_L + V_T$

Calculate Product Vector $V_{PP_{Product}}$ using $Gensim(M_I)$

Initialize Recommended Product R

While len(R) == Top-N **do**

For Purchased Product Vector $V_{PP_{User}}$ **do**

 Calculate Cosine similarity $s(u_r, V_{PP_{User}})$

 Get Similarity User $u_s = \text{Max}(s(u_r, V_{PP_{User}}))$

 Recommended Product R Add Only Top-N/2 of u_s Purchased Product list

 Get u_r Recent Purchased Product p_r

For Product Vector $V_{PP_{Product}}$ **do**

 Calculate Cosine Similarity $s(p_r, V_{PP_{Product}})$

 Get Similarity Product $p_s = \text{sorted}(s(p_r, V_{PP_{Product}}))$ Descending Order

 Recommended Product R Add Only Top-N/2 of p_s

To compare the performance of user-based CF, item-based CF, and hybrid CF combining the two, the recommended product list was extracted as Hit-Rates 3, 5, and 10 to verify the recommendation system.

4.2.2. Single-Merchant vs. Multi-Merchant Recommendation Comparison and Validation

Based on the Hybrid CF model implemented in the previous step, we verified the two recommendation methods: single-merchant and multi-merchant. Single-merchant recommendation systems utilize only their merchant data to train the hybrid CF models.

The multi-merchant recommendation system trains the hybrid CF model by referring to other merchants' purchase transactions.

4.3. Evaluation Metric

Herein, we evaluate the accuracy of the ECF recommendation methodology in multi-merchant and single-merchant situations. We regarded the user's last purchase product as the label and chose a method to predict the label through the user's purchase record excluding the last product. To evaluate the accuracy of the proposed algorithm, we used a popular hit-rate (HR) method in the top-N recommendation methods [27]. Here, n is the total number of users, and the number of hits is the products actually purchased by the user from among the recommended products.

The definition of the hit-rate is as follows:

$$\text{Hit - Rate} = \frac{\text{The number of hits}}{n} \quad (2)$$

5. Results

5.1. Comparison of Various CF Algorithms

Table 2 and Figure 3 show the performance of the three recommendation algorithms. Overall, hybrid CF performed better than item-based CF and user-based CF. This can be seen as being consistent with prior studies where hybrid CF showed high performance. Based on these results, we performed a performance comparison using hybrid CF as the recommendation algorithm of the ECF system for single-merchant and multi-merchant methods.

Table 2. Comparison of Collaborative Filtering Algorithms.

	hit@3	hit@5	hit@10
User-Based CF	4.567	6.899	12.180
Item-Based CF	6.879	9.615	14.611
Hybrid CF	6.863	9.703	15.604

Figure 3. Comparison of Collaborative Filtering Algorithms by Hit-Rate.

5.2. Single-Merchant vs. Multi-Merchant Recommendation Algorithms

A comparison of the performance of the single-merchant and multi-merchant recommendation algorithms for each merchant is shown in Table 3 and Figure 4. First, for all three examined hit-rates, the average performance of multi-merchant recommendation

was higher than the average performance of single-merchant recommendation. Second, the smallest merchant, Merchant S, showed higher recommendation performance by multi-merchant recommendation algorithms than by single-merchant recommendation algorithms. However, for the larger Merchants L1 and L2, single-merchant recommendation algorithms outperformed multi-merchant recommendation algorithms overall. In the case of Merchant L1 and Merchant L2, which are rich in data, the data of smaller merchants would rather act as noise in the recommendation system, resulting in decreased performance. In the case of Merchant M and Merchant S, which have relatively insufficient data, the ECF system that includes data from other merchants could improve their performance. In particular, in the case of Merchant S, the recommendation performance increased significantly compared with Merchant M. Therefore, we can conclude that the fewer transaction data available, the greater the effect of the ECF system.

Table 3. Comparison of Single-Merchant and Multi-Merchant Algorithms.

	hit@3		hit@5		hit@10	
	Single	Multi	Single	Multi	Single	Multi
Merchant L1	5.30	< 5.43	7.45	> 7.31	11.23	> 10.76
Merchant L2	9.94	> 9.63	13.92	> 13.42	21.03	> 20.72
Merchant M	8.32	> 6.90	12.13	< 12.74	19.91	< 20.69
Merchant S	6.12	< 10.25	10.97	< 13.85	20.86	< 23.02
Average	7.42	< 8.05	11.12	< 11.83	18.26	< 18.80

Figure 4. Comparison of Single-Merchant and Multi-Merchant Algorithms.

6. Implementation of the Recommendation System

Harex InfoTech, a payment company in Korea, provides a food ordering and delivery service, Ulsan Pedal. The Ulsan Pedal service holds transaction data relating to various merchants. As a multi-merchant system connecting SMEs, ECF system applications can

provide relevant recommendations to users and provide merchants with new revenue opportunities. After configuring the data for the ECF system, the ECF system was applied to Ulsan Pedal. The data used for the simulation were 13,385 real payments made via Ulsan Pedal for two months from March 2021. The number of merchants in these data was 966, but because it is a new platform launched in March, it has few transaction data compared to the number of merchants. This model created a data set by considering only the purchased product, the purchase location, and the purchase time, excluding user demographic data. The development environment was Anaconda, python 3.7, and Docker (Table 4).

Table 4. Development Environment.

Development Environment	
Hardware	OS: Ubuntu version 18.4.1
	CPU: Intel Core i7-6850K(3.6 GHz)
	VGA: GTX 1080 Ti
	RAM: 64 GB
Software	Web Server: Nginx version 1.19.7
	Database: MariaDB version 10.5.10
	Web Application: Python version 3.8, Django version 3.2

We used Docker container technology, which provides virtualization technology for stable service development, deployment, and management of Ulsan Pedal. Harex InfoTech's main database sent Ulsan Pedal transaction data to the ECF recommendation system database using API. It updated the recommendation algorithm through the transaction data. The system structure for providing the recommendation service is shown in Figure 5. When the purchase history of Ulsan Pedal users was stored in Harex InfoTech's main database, Harex InfoTech transmitted the transaction data to the recommendation system server through API. The recommendation system then derived a product list.

Figure 5. Ulsan Pedal Recommendation System Architecture.

From the application of the recommendation system to a real business, the simulation results for Ulsan Pedal are shown in Table 5. The average recommendation response time of this recommendation system was 0.066 ms per case. With the multi-merchant ECF system, various merchants can be mutually recommended. Merchant products that have never been purchased can also be recommended, so it provides a magical experience to users. In addition, Ulsan Pedal is a shared platform that cooperates with various merchants and can provide benefits to merchants by lowering payment fees. The system recommendation details are at the merchant's product level, so the advertising effect is maximized. SMEs

have difficulty applying recommendation systems, but the AI divide can be reduced through a shared platform in Ulsan Pedal.

Table 5. Ulsan Pedal Recommendation System Simulation.

No.	Item Code	Item Name	Store Code	Store Name

1	38069	Ice Café Latte	2141002342	Duda
2	33917	Old-Style Chicken	2131001337	Maxican Chicken
3	13272	Carbonara Pasta	2121000843	Pasta House

7. Conclusions

In this study, we proposed an ECF system that can provide an appropriate recommendation service to users, especially small and medium-sized companies. To verify the proposed structure, we compared and verified the recommendation accuracy in two situations, single-merchant and multi-merchant, using an open dataset. We implemented a recommendation system using the ECF algorithm using actual payment company data.

Three models were compared: user-based CF using PP2Vec on user, item-based CF using PP2Vec on product, and hybrid CF, a method that combines the two. Through a comparative experiment, hybrid CF showed the highest performance. Thus, in this study, we finally selected hybrid CF as the algorithm for ECF systems. Finally, we compared single-merchant and multi-merchant performance using ECF.

The application of the recommendation algorithm showed that the multi-merchant recommendation algorithm has higher performance for relatively small merchants. We implemented a recommendation system for the ECF system using Ulsan Pedal data. The ECF system proposed in this study is more suitable for a real-world business environment because it does not directly share data among companies. Our study shows that the ECF system can contribute to the sustainability and viability of economic systems by providing high-performance recommendation capability to small and medium-sized enterprises and start-ups. This ECF system makes it possible for SMEs and start-ups to introduce recommendation services. In addition, the ECF system will be able to generate revenue through product recommendations relevant to customers.

The theoretical implications of this study are as follows. Through the development of the PP2Vec algorithm, companies in a data-scarce environment can make more relevant

recommendations to users. In addition, even if only minimum data are shared, it is possible to make appropriate recommendations to users. As a managerial implication, for SMEs that have difficulty in developing a recommendation system, ECF systems can increase their productivity. However, there is a limitation to this study. Although the algorithm was developed by minimizing the data and excluding the users' personal information, some user data or purchase history data are shared. In future studies, we intend to develop an algorithm that can make relevant recommendations to users without sharing data by using a federated learning model. We will also study whether this algorithm works in other domains.

Author Contributions: Conceptualization, K.J.L. and K.Y.P.; software, Y.H. and J.Y.; validation, Y.H., B.J. and J.Y.; formal analysis, Y.H. and B.J.; resources, K.Y.P.; data curation, J.Y.; writing—original draft preparation, Y.H.; writing—review and editing, K.J.L.; visualization, B.J.; supervision, K.J.L.; project administration, K.J.L.; funding acquisition, K.Y.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Harex InfoTech. This work was supported by the Ministry of Education of the Republic of Korea and the National Research Foundation of Korea (NRF-2020S1A5B8103855). This research was supported by the BK21 FOUR (Fostering Outstanding Universities for Research) funded by the Ministry of Education (MOE, Korea) and National Research Foundation of Korea (NRF).

Data Availability Statement: The data used in this study is proprietary as real payment company data.

Acknowledgments: The authors give special thanks to Jong Il Park's friendly editing help.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Lee, H.I.; Choi, I.Y.; Moon, H.S.; Kim, J.K. A Multi-Period Product Recommender System in Online Food Market based on Recurrent Neural Networks. *Sustainability* **2020**, *12*, 969. [\[CrossRef\]](#)
2. Ji, Z.; Pi, H.; Wei, W.; Xiong, B.; Woźniak, M.; Damasevicius, R. Recommendation based on review texts and social communities: A hybrid model. *IEEE Access*. **2019**, *7*, 40416–40427. [\[CrossRef\]](#)
3. Greenstein-Messica, A.; Rokach, L. Personal price aware multi-seller recommender system: Evidence from eBay. *Knowl. Based Syst.* **2018**, *150*, 14–26. [\[CrossRef\]](#)
4. Kim, Y.; Park, Y. Fourth Industrial Revolution and SME Supporting Policy. *J. Korea Technol. Innov. Soc.* **2017**, *20*, 387–405.
5. Adadi, A. A survey on data-efficient algorithms in big data era. *J. Big Data* **2021**, *8*, 1–54. [\[CrossRef\]](#)
6. Hansen, E.B.; Bøgh, S. Artificial intelligence and internet of things in small and medium-sized enterprises: A survey. *J. Manuf. Syst.* **2021**, *58*, 362–372. [\[CrossRef\]](#)
7. Goldberg, D.; Nichols, D.; Oki, B.M.; Terry, D. Using collaborative filtering to weave an information tapestry. *Commun. ACM* **1992**, *35*, 61–70. [\[CrossRef\]](#)
8. Resnick, P.; Varian, H.R. Recommender systems. *Commun. ACM* **1997**, *40*, 56–58. [\[CrossRef\]](#)
9. Covington, P.; Adams, J.; Sargin, E. Deep neural networks for youtube recommendations. In Proceedings of the 10th ACM Conference on Recommender Systems, Boston, MA, USA, 15–19 September 2016; pp. 191–198.
10. Elahi, E.; Chandrashekar, A. Learning Representations of Hierarchical Slates in Collaborative Filtering. In Proceedings of the Fourteenth ACM Conference on Recommender Systems, Virtual Event, Brazil, 22–26 September 2020; pp. 703–707.
11. Sarwar, B.; Karypis, G.; Konstan, J.; Riedl, J. Analysis of recommendation algorithms for e-commerce. In Proceedings of the 2nd ACM Conference on Electronic Commerce, Minneapolis, MN, USA, 17–20 October 2000; pp. 158–167.
12. Sarwar, B.; Karypis, G.; Konstan, J.; Riedl, J. Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th International Conference on World Wide Web, Hong Kong, 1–5 May 2001; pp. 285–295.
13. Bobadilla, J.; Ortega, F.; Hernando, A.; Gutiérrez, A. Recommender systems survey. *Knowl. Based Syst.* **2013**, *46*, 109–132. [\[CrossRef\]](#)
14. Bobadilla, J.; Ortega, F.; Hernando, A.; Bernal, J. A collaborative filtering approach to mitigate the new user cold start problem. *Knowl. Based Syst.* **2012**, *26*, 225–238. [\[CrossRef\]](#)
15. Burke, R. Hybrid recommender systems: Survey and experiments. *User Modeling User-Adapt. Interact.* **2002**, *12*, 331–370. [\[CrossRef\]](#)
16. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. *arXiv* **2013**, arXiv:1301.3781.

17. Yang, Z.; He, J.; He, S. A collaborative filtering method based on forgetting theory and neural item embedding. In Proceedings of the 2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference, Chongqing, China, 24–26 May 2019; pp. 1606–1610.
18. Jun, H.J.; Kim, J.H.; Rhee, D.Y.; Chang, S.W. “SeoulHouse2Vec”: An Embedding-Based Collaborative Filtering Housing Recommender System for Analyzing Housing Preference. *Sustainability* **2020**, *12*, 6964. [[CrossRef](#)]
19. Barkan, O.; Koenigstein, N. Item2vec: Neural item embedding for collaborative filtering. In Proceedings of the 2016 IEEE 26th International Workshop on Machine Learning for Signal Processing, Salerno, Italy, 13–16 September 2016; pp. 1–6.
20. Ozsoy, M.G. From word embeddings to item recommendation. *arXiv* **2016**, arXiv:1601.01356.
21. Chen, F.; Liu, X.; Proserpio, D.; Troncoso, I. Product2Vec: Understanding Product-Level Competition Using Representation Learning. *NYU Stern Sch. Bus.* **2020**. [[CrossRef](#)]
22. Li, B. Cross-domain collaborative filtering: A brief survey. In Proceedings of the 2011 IEEE 23rd International Conference on Tools with Artificial Intelligence, Boca Raton, FL, USA, 7–9 November 2011; pp. 1085–1086.
23. Zhang, H.; Kong, X.; Zhang, Y. Selective Knowledge Transfer for Cross-Domain Collaborative Recommendation. *IEEE Access*. **2021**, *9*, 48039–48051. [[CrossRef](#)]
24. Lee, K.J.; Hwangbo, Y.; Jeong, B.; Park, K.Y.; Park, J.I. User-Centric AI: Definition and Approach. In Proceedings of the 2020 Fall Conference of The Korea Society of Management Information Systems, Seoul, Korea, 17 December 2020.
25. Lotte Members. Bigdata Competition. In Proceedings of the 3rd Conference L.POINT, Seoul, Korea, 21 November 2016.
26. Harris, Z.S. Distributional structure. *Word* **1954**, *10*, 146–162. [[CrossRef](#)]
27. Deshpande, M.; Karypis, G. Item-based top-n recommendation algorithms. *ACM Trans. Inf. Syst.* **2004**, *22*, 143–177. [[CrossRef](#)]