ICIC Express Letters
Volume **18**, Number **2**, February **2024**
 ISSN 1881-803X
pp. **167–174**

AUTOENCODER-BASED CARGO RECOMMENDATION SYSTEM WITH LATENT FACTOR MODEL

Yeonsoo Kim[1] and Tai-Woo Chang[1,2,*]

[1]Department of Industrial and Systems Engineering
[2]Intelligence and Manufacturing Research Center
Kyonggi University
154-42 Gwanggyosan-ro, Yeongtong-gu, Suwon, Gyeonggi 16227, Korea
yeonsoo98@kyonggi.ac.kr; *Corresponding author: keenbee@kgu.ac.kr

Received May 2023; accepted July 2023

Abstract. *As the volume of cargo transportation increases, the cargo brokerage platform is also growing. With the growth of the market, many cargo transactions go unfulfilled because cargo owners cannot find vehicle owners to transport the cargo, and vehicle owners cannot find the shipment they are looking for. To solve these problems, we propose a recommendation system that supports the matching of cargo owners and vehicle owners when brokering cargo transportation. This study was conducted using transaction data from a real platform company and aims to reduce the number of unfulfilled contracts by providing a cargo recommendation system. In this case, we recommend cargo items using latent factor collaborative filtering using a decomposition of the matrix of the user (vehicle owner) and the desired item (cargo). To extract latent factors, we use an autoencoder, one of the deep learning models. Latent factors are used to explain the features of users and items and to calculate similarities and predicted rating. The recommendation system developed in this study could provide personalized cargo recommendations to users by considering user and item features.*
Keywords: Cargo recommendation system, Latent factor filtering, Autoencoder

1. **Introduction.** The platform-based business of the logistics market is progressing under the influence of the digital transformation. The cargo marketplace creates an environment where different types of cargo can be traded across markets and locations [1]. While these platforms have innovatively transformed the transactional relationships between cargo owners and vehicle owners, the expansion of this transactional form poses problems with incomplete contracts [2]. Due to the huge volume of cargo, cargo owners often struggle to find vehicle owners to transport their cargo, and vehicle owners have difficulty finding their desired cargo due to overshadowing by other cargo. These incomplete contracts can result in unnecessary costs and delays. As a result, the development of recommendation system technology has gained traction due to the growing importance of this trade form [3].

In this study, we propose a method to reduce the number of incomplete contracts by developing a recommendation system. This system utilizes transaction data from a real cargo brokerage platform company. The cargo brokerage platform provides a three-way dispatching system that enables digital communication between cargo owners, vehicle owners, and operators through web and app-based online platform service. The proposed method employs a collaborative filtering recommendation system of a latent factor based on matrix factorization, taking account of the features of the cargo datasets. An autoencoder model is utilized to generate a vector of latent features that represent the features of users and items. These features are then used to make cargo recommendations. Additionally, the system provides predicted ratings to improve the performance of the

DOI: 10.24507/icicel.18.02.167

recommendation system and enhance user satisfaction. In conclusion, this research addresses the challenges posed by incomplete contracts in the platform-based business of the logistics market. The expansion of transactional platforms has introduced complexities in matching cargo owners with vehicle owners, leading to unnecessary costs and delays. To mitigate these issues, we proposed and developed a recommendation system to reduce the number of incomplete contracts.

In the remainder, we organize the content of this paper as follows. Section 2 discusses the related works and Section 3 presents and explains the research process. Section 4 describes the results of the research and the recommendation system. Finally, Section 5 concludes the content of this paper.

2. **Related Works.** The cargo transport market has grown alongside the expansion of the logistics market. In particular, the rise of e-commerce has accelerated the quantitative growth of this market. However, compared to the quantitative growth, the qualitative growth has been somewhat stagnant. Currently, the level of information management in the logistics industry is relatively low, and logistics information is predominantly provided in a passive manner [4]. The development of e-commerce has increased the demand for transportation of various types of cargo, which requires flexible communication. However, the existing logistics industry remains entrenched in offline transportation transactions and outdated work methods, impeding innovation according to market needs. As a solution, the digital transformation of the cargo transportation market is underway, with domestic and foreign companies establishing cargo transportation networks through various means [5]. The cargo brokerage platform in our study is one such example.

Recommendation systems play an important role in reducing open contracts in cargo transportation brokerage platforms. These systems utilize technologies that provide personalized recommendations based on user preferences and item features. Recommender system models are divided into two main approaches: collaborative filtering, which recommends items that were preferred by users with similar tastes to the target user, and content-based filtering, which recommends items with similar content information to the items preferred by the target user [6]. Recently, there has been considerable research on applying deep learning to collaborative filtering. Deep learning, which has demonstrated favorable outcomes in fields like computer vision and natural language processing, has received positive evaluations for its potential in recommendation systems due to its nonlinear transformation and flexibility [7]. Among these approaches, utilizing autoencoders, which are effective in learning low-dimensional representations of input data, has shown good performance [8].

It can generate a list of recommendations by finding similar users or items based on the learned latent factors. The latent factors then utilize each other's preferences in the latent space to make suitable recommendations. This method is called "latent factor collaborative filtering" [9]. Among collaborative filtering algorithms, it uses matrix factorization for users and items to extract features and use them for recommendations. It yields results similar to traditional collaborative filtering algorithms but offers the advantage of learning latent factors without relying heavily on rating data during training. As a result, it performs well even when data is limited or sparse. In this study, we use an autoencoder to learn the latent factors of users and items for cargo recommendation.

An autoencoder is an unsupervised neural network model that approximates the input and output values to the same value. It consists of an encoder and a decoder, with a symmetrical structure based on a code layer. The middle layer, called the Bottleneck, is the output of the encoder, which outputs a mapping of the input data into a low-dimensional latent space. The decoder learns by reconstructing the input data using that vector as input again. During this learning process, the encoder aims to preserve as much information as possible for reconstruction and attempts to extrude and encode only the most important key information [10].

Considering the significance of digital transformation and the development of recommendation systems in the cargo transportation market, it becomes evident that this study aims to address these aspects. By utilizing deep learning techniques in a cargo brokerage platform, we aim to meet the market's needs. Through this research, we hope to provide a solution that promotes the digital transformation of the cargo transport market, enhances information management practices, and delivers improved cargo transport services to users.

3. Research Process.

3.1. Designing recommendation system. In this study, an autoencoder model is utilized to develop a cargo recommendation system that takes account of the unique characteristics of cargo brokerage data. Unlike general recommendation systems, there are no evaluation results available for concluded cases in cargo brokerage data. Furthermore, each registered transportation case has different attributes, and cargo is dynamically added and removed in real time. Therefore, it is crucial to develop a recommendation system that can effectively handle these features.

The employed autoencoder trains the model using user and item attributes as input. During the training process, the dimensionality is reduced by converting the user and item attributes into low-dimensional vectors. This low-dimensional representation of vectors proves valuable for processing large volumes of user and item data, as it enhances computational efficiency and expedites the learning and inference processes of the model. Moreover, it enables the extraction of significant features pertaining to users and items [11].

It does not take consideration of all the features of the dataset but is represented as a vector that identifies patterns in the data and expresses key features. In addition, due to the real-time nature of cargo transactions that generate new data, recommendations can be provided based on existing data even when new users or items are added. The above is the advantage of using an autoencoder in a recommendation system.

The trained autoencoder model transforms the input features into latent features and subsequently decodes them back into the original features. This training process enables the model to effectively reconstruct the original features using the learned latent factors [12]. Based on the calculated latent factors of users and items, recommendation lists are generated. The model extracts latent factors by utilizing the user's features and item's features as inputs, which are then used by the decoder part of the model to generate predicted features. Using these attributes, a predicted rating is calculated for items that the user has not yet experienced.

Furthermore, the latent factor collaborative filtering model maps the features of users and items to latent factors, leveraging these mappings to generate predicted ratings and facilitate cargo recommendations.

3.2. Data description. This study analyzes data collected from a cargo brokerage platform, specifically focusing on transactions completed during a six-months period from April 2020 to September 2020. The dataset comprises a total of 1,885,033 data points, encompassing 78 factors related to cargo, delivery, cargo owner, and vehicle owner information. To implement the latent factor filtering-based recommendation system, we selected data containing the features of vehicle owner and cargo, representing the user and item features, respectively. The selected data was used for training and recommendation purposes within the recommendation system model.

The user data was grouped based on the user's registration key. When selecting vehicle owner data, we considered the user's profile information and behavior, focusing on variables that aid in identifying the user's transportation needs and preferences.

The item data was grouped based on the transaction registration key. Regarding item features, we selected data that could effectively describe and categorize the cargo. This information helps recommend the most suitable cargo based on the input values provided by the cargo owner. Table 1 presents the details of the user and item data.

Table 1. Selected user and item data

User data	Item data
• PayType	• CargoType
• CarType	• CargoWeight
• CarSize	• LoadingAreaCode
• MemberPrice	• LoadingType
• LoadingAreaCode	• UnloadingAreaCode
• LoadingType	• UnloadingType
• Distance	• Distance

Due to user omissions and mistakes, the collected cargo transaction data often contains missing values. In the selected data, these missing values were replaced with zeros. To handle categorical variables, a label encoder was utilized, enabling the application of categorical data in the autoencoder algorithm. StandardScaler was also employed to enhance algorithm performance and normalize the data. Furthermore, the features of the selected users and items were taken into consideration, and weights were assigned to improve the model's performance and prioritize specific features.

3.3. **Constructing and training an autoencoder model.** To construct the autoencoder model, we set the settings as follows. The number of nodes in the input and output layers is set equal to the size of the user-item matrix. The number of nodes in the hidden layer is set to the number of latent factors of the user and item, so that the model can accurately reconstruct the input data. These settings enable the model to effectively represent and compress the input data extracting latent factors that reflect the interaction between users and items.

The autoencoder model consists of an input layer, an encoder, a decoder, and an output layer. Each layer is constructed using dense layers with ReLU serving as the activation function. The encoder and decoder employ the sigmoid function, which is useful for reconstructing input data by limiting the output to values between 0 and 1. The number of nodes in each dense layer is adjusted to regulate algorithm performance and model complexity. Two hidden layers are used in the encoding and decoding layers. The hidden layers are used to learn an intermediate representation containing latent factors of the user and item. In the first hidden layer, 50 nodes are incorporated to enhance the expressive power to learn the complex features of the model, and 15 nodes are used in the second hidden layer to reduce the complexity to learn the simple features of the model. This ensured that the model was sufficiently complex to learn the patterns in the data to avoid overfitting and consequently achieve optimal latent factor extraction. During training, the weights of each layer are internally adjusted to minimize the error between input and output. The Adam optimizer is used to compile the model, minimizing the mean squared error (MSE) value. Subsequently, training is conducted separately for the user autoencoder and item autoencoder. Each characteristic is trained independently to determine similarity and recommendation targets.

The training environment utilizes the following hardware specifications: GPU GeForce RTX 4090, CPU i9-13900K, 128GB of RAM. The parameters for each autoencoder are set as follows: the number of epochs is set to 100, and the batch size is set to 64.

3.4. **Calculating similarity and recommending items.** Considering the features of the cargo market and dataset described above, the cargo recommendation system takes account of records of users similar to the target user in order to recommend suitable cargo options. The system utilizes an autoencoder model to process the user's features and item's features as input, extracting latent factors through the encoder. This approach allows the recommendation system to incorporate the characteristics of the cargo transaction dataset and provide personalized recommendations.

The user's attributes are passed through the autoencoder model using their registration key, resulting in an encoding vector that captures the most relevant information about the user's preferences in a low-dimensional representation. This encoding vector is representative of the user's preferences. Next, the feature matrix for all users is processed through the autoencoder model to generate encoding vectors for each user.

To measure the similarity between a given user and other users based on their preferences, the cosine similarity is calculated between the encoding vector of the target user and the encoding vectors of all other users. Cosine similarity is used to increase efficiency through simple and fast calculations because the amount of data in the cargo transaction dataset is vast. After defining user features and item features, we present formulas for the metrics we provide as follows.

$$\textit{user feature: } U,\ \textit{item feature: } I \tag{1}$$
$$\textit{user encoding vector: } V(U) \in \mathbb{R}^d \tag{2}$$
$$\textit{item encoding vector: } V(I) \in \mathbb{R}^d \tag{3}$$
$$\textit{user similarity}(U, i) = \cos(V(U), V_i(U)) \tag{4}$$
$$\textit{item similarity}(I, j) = \cos(V(I), V_j(U)) \tag{5}$$
$$\textit{predicted rating} = \textit{Rating}(I, j) \tag{6}$$

This similarity measure is defined as the user similarity. Higher user similarity indicates that other users have similar behaviors and transaction records to the target user. Simultaneously, the item's features are passed through the autoencoder model using the transaction registration key to obtain the item's encoding vector. Then, the cosine similarity is calculated between the encoding vector of a specific item and the item vectors of

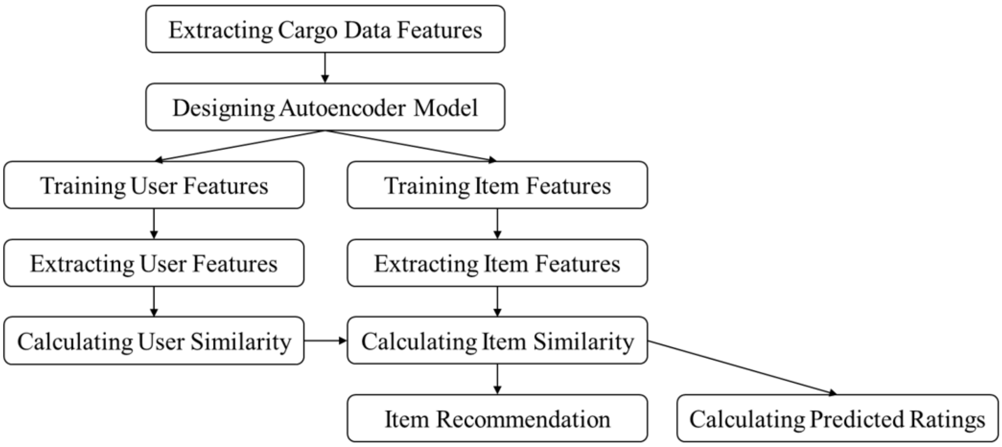

Figure 1. Autoencoder-based cargo recommendation system configuration diagram

users with high user similarity to the target user. This similarity measure is referred to as the item similarity.

Based on the calculated item similarity, the system recommends a predetermined number of cargo options to the target user in descending order of similarity. Additionally, the system calculates the predicted rating, which combines the item similarity and the full item attribute encoder vector. The predicted rating is scaled, ensuring the lowest value is set to 0 and the highest value is set to 5, providing an indication of the level of recommendation for each cargo option.

4. **Results.** This paper proposes a recommendation system for cargo transportation using transaction data from a brokerage platform. The system utilizes autoencoders to generate personalized cargo recommendations by considering the characteristics of users and items. The results of the autoencoder-based cargo recommendation system are summarized as follows.

1) User Similarity: User similarity is calculated by measuring the cosine similarity between the encoding vector of the target user and the encoding vectors of all other users. This similarity metric helps identify users with similar preferences and extract relevant information from their records.
2) Item Similarity: Item similarity is determined by calculating the cosine similarity between the item vector of a specific user and the item vectors of users with high user similarity. It quantifies the similarity between each item and the preferences of the target user.
3) Predicted Rating: The predicted rating is computed by combining the item similarity of recommended items and the encoding vector of the overall item features. This rating indicates the similarity between the item and the user's preferences, with higher values indicating a stronger preference. The predicted rating is scaled to provide a relative rating based on the user's preferences.

The results demonstrate that the autoencoder-based cargo recommendation system can effectively recommend items that are highly relevant to the preferences of similar users. This approach enables personalized recommendations that take account of user preferences, and a fixed number of items are recommended to the user based on the item similarity and predicted rating.

A specific user has conducted approximately 50,000 transactions, and each transaction is associated with specific cargo types and geographical locations. The regional-related codes are divided into province first, and then into city/county/district. We can compare the user's transaction data with the recommendation results by checking out Table 2 and Table 3. As for the cargo recommendation result, five items with high item similarity have been recommended. The metrics provided by the system are good, but they are not validated because there is no real user feedback.

TABLE 2. Specific user transaction data

LoadingAreaCode	UnloadingAreaCode	CargoType	CargoWeight	CarType	CarSize	LoadingType
H160020	H290019	regular	1.5 ton	Wing	1.4 ton	regular
H150101	K080020	regular	5.5 ton	Cargo	5 ton	hoist
H260017	I160009	regular	5.5 ton	Cargo	5 ton	regular
H180004	K130000	regular	9 ton	Wing	11 ton	regular
H280036	H220011	regular	12 ton	Wing	11 ton	forklift

Table 3. Cargo recommendation results of a specific user TOP 5

LoadingAreaCode	UnloadingAreaCode	CargoType	CargoWeight	User Similarity	Item Similarity	Predicted Rating
H160020	K030006	regular	25.0 ton	0.7956	0.9642	4.3357
H150101	K020000	regular	5.5 ton	0.7944	0.9640	3.0541
H260017	A160012	regular	22.0 ton	0.7902	0.9639	3.0421
H180004	E010066	regular	7.0 ton	0.7902	0.9639	4.0451
H280036	H140028	regular	27.0 ton	0.7915	0.9638	3.4219

5. **Conclusions.** In this study, we propose a personalized cargo recommendation system using transaction data from a cargo brokerage platform. In order to consider the user and item features, latent factors were extracted using an autoencoder, and user similarity and item similarity were calculated and applied to the recommendation process.

The developed cargo recommendation system reflects the features of the cargo transaction dataset and provides personalized recommendations. Considering the user's preferences and the records of other users similar to the target user, the system recommended cargo and provided a predicted rating. Therefore, the autoencoder-based latent factor collaborative filtering method proposed in this study can be used to provide efficient and personalized recommendation service in the field of cargo transportation.

For future research, it is suggested to enhance the model's performance by incorporating real user feedback and optimizing the weighting of user and item features. Additionally, conducting comprehensive evaluations using various metrics would provide a more robust assessment of the recommendation system. Ultimately, the development of such a system can contribute to satisfying customers' needs and increasing profitability for cargo transportation businesses.

Acknowledgment. This work was supported by Kyonggi University's Graduate Research Assistantship 2023. And this work was supported by a Korea Institute for Advancement of Technology (KIAT) grant funded by the Korea Government (MOTIE) (P0008691, HRD Program for Industrial Innovation).

REFERENCES

[1] J. Lee, Y. Oh and K.-S. Park, A study on the effects of logistics platform features on the work productivity of logistics companies according to the platform utilization, *Korea Logistics Review*, vol.32, pp.15-29, 2022.

[2] J. Lim, S. Lee and Y.-D. Seo, A group recommender system with low-dimensional clustering, *Proc. of KIIT Conference*, pp.374-377, 2022.

[3] M. Matteo, Z. Alessandro, A. Andrea and G. Andrea, Recommendation systems: An insight into current development and future research challenges, *IEEE Access*, vol.10, pp.86578-86623, 2022.

[4] H.-M. Oh and H.-S. Lee, A study on the choice factors of online freight marketplace, *Korea Trade Review*, vol.46, no.4, pp.189-204, 2021.

[5] Y.-M. Ha, J.-R. Cho and K.-S. Park, A study on multimodal transport operators' intention to accept on-line freight marketplace, *International Commerce and Information Review*, vol.20, no.4, pp.27-48, 2018.

[6] D.-G. Hong, Y.-C. Lee, S.-W. Kim and J. Lee, Effectiveness of content-based filtering technique on recommendation system, *Proc. of the Korean Information Science Society Conference*, pp.266-267, 2017.

[7] S. Zhang, L. Yao, A. Sun and Y. Tay, Deep learning based recommender system: A survey and new perspectives, *ACM Computing Survey*, vol.52, no.1, 2020.

[8] M. K. Seo and W. Y. Yun, Condition monitoring and diagnosis of a hot strip roughing mill using an autoencoder, *Journal of the Korean Society for Quality Management*, vol.47, no.1, 2019.

[9] D. Park, J. Kang and J.-G. Lee, Latent representation learning for autoencoder-based top-K recommender system, *Journal of KIISE*, vol.47, no.2, pp.207-215, 2020.

[10] B. Min, J. Yoo, S. Kim, D. Shin and D. Shin, Network intrusion detection with one class anomaly detection model based on auto encoder, *Journal of Internet Computing and Services*, vol.22, no.1, pp.13-22, 2021.
[11] S. Sedhain, A. K. Menon, S. Sanner and L. Xie, AutoRec: Autoencoders meet collaborative filtering, *Proc. of the 24th International Conference on World Wide Web*, pp.111-112, 2015.
[12] W.-J. Bong, H. Bang and J.-H. Lee, Similarity-based movie recommender system using auto-encoder, *Processdings of KIIS Spring Conference*, vol.26, no.1, 2016.