Logistics Audience Expansion via Temporal Knowledge Graph

Hua Yan
Lehigh University, Bethlehem, USA
JD Logistics, Beijing, China
huy222@lehigh.edu

Yingqiang Ge
Rutgers University
Piscataway, USA
yingqiang.ge@rutgers.edu

Haotian Wang
JD Logistics
Beijing, China
wanghaotian18@jd.com

Desheng Zhang
Rutgers University
Piscataway, USA
desheng@cs.rutgers.edu

Yu Yang
Lehigh University
Bethlehem, USA
yuyang@lehigh.edu

ABSTRACT

Logistics audience expansion, the process for logistics companies to find potential long-term customers, is one of the most important tasks for business growth. However, existing methods for conventional audience expansion fall short due to two significant challenges, the intricate interplay of multiple complex factors in the logistics scenario and the emphasis on long-term logistics service usage instead of one-time promotions. To address the above limitations, we design **LOGAE-TKG**, a logistics audience expansion method based on a temporal knowledge graph, which consists of three components: (i) a temporal logistics knowledge graph pre-trained model to model the effect of multiple complex factors and build a solid logistics knowledge base for contracting and usage prediction; (ii) an intention learning model with data augmentation-based comparison to capture the contracting intention; (iii) a future pattern discovery model to uncover post-contract patterns. We evaluate and deploy our method on the JingDong e-commerce platform. Extensive offline experiment results and real-world deployment results demonstrate the effectiveness of our method.

CCS CONCEPTS

• **Computing methodologies → Knowledge representation and reasoning**; • **Applied computing → Transportation**.

KEYWORDS

Audience Expansion; Logistics System; Knowledge Graph

ACM Reference Format:
Hua Yan, Yingqiang Ge, Haotian Wang, Desheng Zhang, and Yu Yang. 2023. Logistics Audience Expansion via Temporal Knowledge Graph. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management (CIKM '23), October 21–25, 2023, Birmingham, United Kingdom.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3583780.3614695

1 INTRODUCTION

Logistics systems [5], as an essential component of supply chain management [31, 32, 49, 53], play an important role for today's business success. Logistics companies, such as FedEx [13] and UPS [46], take advantage of their well-built infrastructures [9, 10], such as warehouses and distribution centers, to seek growth with a new business model by providing supply chain services for third-party partners (e.g., a conventional company or a merchant on an online platform), from warehouse storage to inventory management to delivery. To enable the continuous growth of such a business model, one of the most important tasks for logistics companies is to find long-term partners, who are willing to use the service (i.e., sign long-term contracts). We name it the *logistics audience expansion problem.* In this work, we are especially interested in the online shopping scenario where the potential partners are online merchants on e-commerce platforms such as Amazon [1] or Taobao [45].

The conventional audience expansion problem has been extensively studied in the e-commerce scenario, where e-commence platforms aim to identify potential users interested in participating in marketing campaigns for specific products [29, 55, 56] (i.e., campaign audience expansion problem). In these studies, users with evident intentions or those who are already engaged in the campaign serve as seed users, and their similarity to other potential users is measured. Subsequently, potential users exhibiting high similarity are chosen for targeted advertising and marketing efforts. The success of the marketing is marked by the conversion of users who engage with the campaign.

We argue that the methods designed for the campaign audience expansion problem cannot solve our problem due to two significant differences. Firstly, **solving the logistic audience expansion problem requires considering more complex relations to better understand merchants' intentions.** In addition to the relationship between merchants and the logistics company, merchants' intentions are affected by multiple complex factors, including static and dynamic relationships. On the one hand, merchants' intentions are influenced by static relationships (e.g., industry and location relationships). For example, merchants selling fresh food may prefer logistics companies with a fast delivery speed that can help maintain their food quality. On the other hand, merchants' intentions are influenced by the dynamic relationship (e.g., the relation between logistics companies and merchants' delivery destinations). For example, if merchants' delivery destinations change, some logistics companies may become unavailable if the destinations are not in the main service areas. Secondly, **logistics companies care about the long-term usage of merchants for continuous profitability.** That is, logistics companies concern not only about the success of contracting but also merchants' future usage of the logistics service.

CIKM '23, October 21–25, 2023, Birmingham, United Kingdom

ACM ISBN 979-8-4007-0124-5/23/10.
https://doi.org/10.1145/3583780.3614695

Existing works did not answer the question how to find merchants who will have long-term usage of the service.

To bridge the gap, we take the opportunity of the recent development of knowledge graphs to model the complex relationships between entities [20]. Especially, e-commerce platforms record rich logistics knowledge (e.g., delivery destination and how merchants use logistics services) in a long duration, which can be used to build a temporal logistics knowledge graph to extract extensive knowledge of logistics, capture merchants' contracting intentions, and explore merchants' post-contract patterns. However, there are still challenges. (i) The relationships represented by the knowledge graph structure dynamically change over time (i.e., 48% relations, indicated by edges in the knowledge graph, get changed every two weeks in our study) , and it is not trivial to extract the dynamic knowledge. (ii) The sparse historical contract records in terms of quantity (i.e., a new contracting rate less than 2% per month) and diversity (i.e., limited contracting data from multiple logistics) make it challenging to capture merchants' contracting intentions and easily susceptible to the overfitting problem. (iii) Merchants' post-contract usage patterns can be different from those prior to contracting , making it challenging for models trained on historical data to effectively predict future usage.

To tackle the above challenges, we design a logistics audience expansion method based on a temporal knowledge graph, named **LOGAE-TKG**. We first construct a temporal logistics knowledge graph (TKG) with a large number of contracted and non-contracted merchants. Based on the TKG, we design a temporal knowledge graph pre-trained model with two pre-training tasks to model the dynamic relationships, capture the interactions between merchants and logistics companies, and identify characteristics of logistics companies, which provides a solid knowledge base for contracting and long-term usage prediction. Then we design an intention learning model with a data augmentation-based comparison to capture merchants' contracting intentions and mitigating data sparsity. For long-time usage prediction, we design a future pattern discovery model for learning contracted merchants' multiple future patterns. In particular, our main contributions are as follows.

- To our best knowledge, we are the first to study the logistics audience expansion problem taking the opportunity of the temporal logistics knowledge graph considering both contracting intentions and long-term usage.
- We design a novel audience expansion model for logistics systems via a temporal knowledge graph named LOGAE-TKG. It consists of (i) a temporal logistics knowledge graph pre-trained model to build a solid logistics knowledge base to extract extensive knowledge of logistics considering dynamics; (ii) an intention learning model with data augmentation-based comparison to address the data sparsity challenge for contracting intention modeling; (iii) a future pattern discovery model to learn merchants' dynamic future usage patterns.
- We implement, evaluate and deploy LOGAE-TKG in one of the largest e-commerce platforms, i.e., JingDong. We have conducted extensive offline experiments based on more than one year of real-world data, which demonstrate that our method outperforms other state-of-the-art methods by showing a 12.83% of improvement in the recall metric, 15.98% of improvement in the F1 score metric, and 21.32% of improvement in the MAE metric. For real-world deployment, the online experiments show a 13.6% of improvement compared to the existing used method.

2 PROBLEM FORMULATION

Temporal Logistics Knowledge Graph. A temporal logistics knowledge graph is defined as a sequence of logistics knowledge graph snapshots: $\mathcal{G} = \{G^1, G^2, ..., G^T\}$, where T denotes the total time steps. G^t refers to a logistics knowledge graph at time step t, which is defined as $G^t = \{\mathcal{V}^t, \mathcal{R}^t, \mathcal{E}^t\}$. $\mathcal{V}^t$ is a set of entities consisting of merchant m, stations s (representing the delivery areas where the merchants are located in), industry d, logistics l, and province p. $\mathcal{R}^t$ is a set of relations consisting of locating relations, belonging relations, serving relations, and shipping relations. $\mathcal{E}^t$ is the set of facts at time step t. A fact in $\mathcal{E}^t$ can be denoted as a quadruple containing the subject, the relation, the object, and the time, where the subject and object belong to entities. We build a logistics knowledge graph based on logistics data (described in Sec. 4.1.1), and an example of it is shown in Fig. 1.

Audience Expansion for Logistics Systems. We divide our problem into two stages: the pre-training stage and the audience expansion stage. In the pre-training stage, we aim to learn a model $f_{pre}(\cdot;\theta_{pre})$ parameterized with θ_{pre}. This model is designed to generate embeddings of entities $v \in \mathcal{V}$ in the temporal logistics knowledge graph $\mathcal{G}$, formulated as $h_v^t = f_{pre}(\mathcal{G};\theta_{pre})$.

In the audience expansion stage, we focus on two sub-tasks: contract prediction and future usage prediction. The goal of the contract prediction is to learn a model $f_s(\cdot;\theta_s)$ parameterized by θ_s, which can identify potential audiences from the candidate pool of a given logistics company. The input of the model is pre-trained embeddings of merchants (i.e., h_m^t) and logistics companies (i.e., h_c^t), and the output $\hat{y}_{m,c}^t$ is whether merchant m will contract with logistics company c near time step t. We formulate the problem of contract prediction as $\hat{y}_{m,c}^t = f_s(h_m^t, h_c^t;\theta_s)$. Meanwhile, the goal of another sub-task: usage prediction, is to learn a model $f_u(\cdot;\theta_u)$ parameterized with θ_u, capable of predicting the future usage of each contracted merchant. The input for this model consists of pre-trained embeddings of merchants (i.e., h_m^t), and the output $\hat{y}_m^t$ represents the usage of merchant after contracting whose contracting time is t. Consequently, we formulate the future usage prediction as $\hat{y}_m^t = f_u(h_m^t;\theta_u)$.

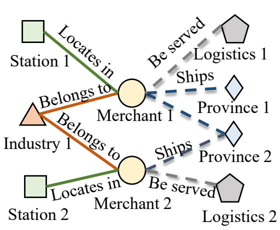

Fig. 1: An Example of the Logistics Knowledge Graph

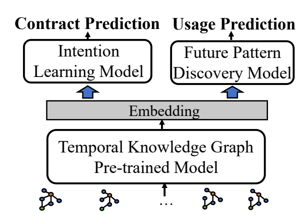

Fig. 2: Framework of LOGAE-TKG

3 OUR METHOD

Our method, LOGAE-TKG, is depicted in Fig. 2 and comprises the following three modules:

Module 1: Temporal Knowledge Graph Pre-trained Model. This module is designed to model the effect of dynamic relations,

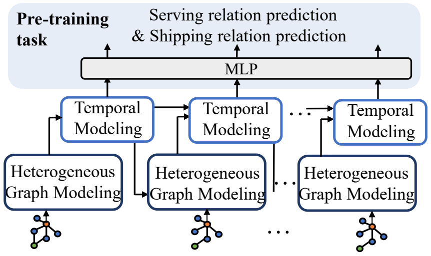

Fig. 3: Temporal Knowledge Graph Pre-trained Model

capture interactions between merchants and logistics companies, and learn the characteristics of logistics companies.

Module 2: Intention Learning Model. Having derived the embeddings from Module 1, we design this module to predict merchants' intentions of contracting with different logistics companies through data augmentation-based comparison.

Module 3: Future Patterns Discovery Model. Using the embeddings of merchants and logistics companies obtained from Module 1, we design this module to identify future patterns of contracted merchants and predict future usage.

3.1 Temporal KG Pre-trained Model

The temporal logistics knowledge graph provides rich information and a solid knowledge base for solving our problem. On the one hand, static relations (e.g., belonging and locating relations) could imply similar intentions. On the other hand, dynamic relations, such as the dynamic serving relations, reflect the merchant's evolving intentions toward logistics companies. Changes in the serving relations may cause a change in the shipping relations. For instance, the shipping time may vary depending on the chosen logistics company. To exploit this information from the knowledge graph, we design a temporal knowledge graph pre-trained model (as shown in Fig. 3) where heterogeneous graph modeling and temporal modeling collaboratively and iteratively learn entity embeddings.

3.1.1 **Heterogeneous Graph Modeling.** In each time step t, the logistics knowledge graph is a heterogeneous graph. Each entity aggregates information from its neighbors to update its embedding. To illustrate this process, we use the merchant entity as an example. For targeted merchant entity m at time t and its neighbors, the l-th layer embedding $z_m^{t,l}$ of the entity is learned by $z_m^{t,l} = \sigma(W_M^l \sum_{r \in \mathcal{R}^t} HSA(z_{u,t}^{l-1} | u \in \mathcal{N}_m^{r,t}) + z_m^{t,l-1})$, where $\mathcal{N}_m^{r,t}$ is the neighbors of entity m^t based on relation r. W_M^l is trainable weights of l-th layer and $\sigma(\cdot)$ is an activation function. HSA is a heterogeneous aggregation function [18, 41]. For example, through static relations, we gather the merchant's industry and the located station and acquire information about merchants with similar industries and stations.

3.1.2 **Temporal Modeling.** For the temporal aspect, we use LSTM [17] to model entity evolution following existing works [6, 25, 27]. To illustrate this process, we use the merchant entity as an example. Formally, the embedding of entity m at time step t is defined as $h_m^t = LSTM(h_m^{t-1}, z_m^{t-1,l})$, where h_m^{t-1} is the hidden state and $z_m^{t-1,l}$ is the output from heterogeneous graph modeling. Importantly, the temporal modeling is flexible and allows the LSTM module to be replaced with different sequential models such as Transformer.

3.1.3 **Pre-training Task.** To capture the dynamic intentions of merchants and the characteristics of logistics companies, we establish two pre-training tasks.

Merchant-Logistics Relation Prediction. We choose the merchant-logistics relation prediction task to learn the merchant's evolving intentions toward logistics companies. After obtaining embeddings of the merchant entity and logistics company entity, we first combine them and then feed them into a Multilayer Perceptron (MLP) followed by a nonlinear layer σ (e.g., sigmoid) to get the prediction at time step $t+1$, expressed as $\hat{y}_{m,c}^{t+1} = \sigma(W_{pre1}[h_m^t \oplus h_c^t])$, where $\oplus$ denotes the concatenation operation. We use the binary-cross-entropy loss function [54] for this task, which is defined as $\mathcal{L}_{pre1} = -\frac{1}{|\mathcal{D}_{pre1}|} \sum_{t \in T-1} \sum_{m,c \in \mathcal{D}_{pre1}^t} y_{m,c}^{t+1} \log(\hat{y}_{m,c}^{t+1}) + (1 - y_{m,c}^{t+1}) \log(1 - \hat{y}_{m,c}^{t+1})$, where $\mathcal{D}_{pre1}$ refers to a pre-training dataset composed of numerous serving relations sampled from the temporal logistics knowledge graph $\mathcal{G}$ at each time step.

Merchant-Province Relation Prediction. In this task, we predict the order number and shipping time between the merchant and the province in the next time step. Since those are related to logistics companies, it indirectly helps in learning the characteristics of logistics companies. Similar to the merchant-logistics relation prediction, the prediction is defined as $\hat{y}_{m,p}^{t+1} = \sigma(W_{pre2}[h_m^t \oplus h_p^t])$. We employ the mean squared error loss function [54] for this task, defined as $\mathcal{L}_{pre2} = \frac{1}{|\mathcal{D}_{pre2}|} \sum_{t \in T-1} \sum_{m,p \in \mathcal{D}_{pre2}^t} (y_{m,p}^{t+1} - \hat{y}_{m,p}^{t+1})^2$, where $\mathcal{D}_{pre2}$ is another pre-training dataset, similar to $\mathcal{D}_{pre1}$. Finally, the loss function of the pre-training task is formulated as $\mathcal{L}_p = \mathcal{L}_{pre1} + \gamma \mathcal{L}_{pre2}$, where γ is a trade-off parameter.

3.2 Intention Learning Model

After obtaining the pre-trained embeddings, we aim to understand the contracting intentions of merchants toward different logistics companies and predict whether to contract when given a specific logistics company. Due to the lack of absolute scores to measure the intention, we adopt pair-wise learning for implicit feedback from the recommender system [40]. Furthermore, to mitigate the bias induced by data sparsity, we design a data augmentation-based comparison. The structure of the intention learning model is depicted in Fig. 4. We first employ a matching network to generate the intention scores for different logistics companies. Subsequently, we utilize two comparison loss functions (i.e., observed comparison and data augmentation-based comparison) to learn these scores.

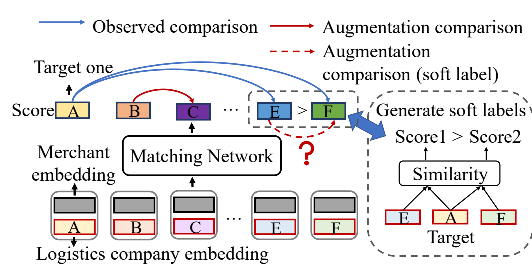

Fig. 4: Intention Learning Model

3.2.1 **Matching Network.** We compute each merchant's intention score for each logistics company using a matching network based on the results of the temporal KG pre-trained model. Specifically, we concatenate the embeddings of merchants h_m^t and logistics companies h_c^t and then input them into a feed-forward neural network to calculate the intention score at time step t, which is defined as $score_{mc}^t = \sigma(W_a[h_m^t \oplus h_c^t])$, where W_a is trainable weights. By ranking these scores, we determine whether to contract with a target logistics company based on its ranking position. For instance, if its ranking is within the top-k, we predict that the contract will be established.

3.2.2 **Observed Comparison.** Following the implicit feedback by pair-wise learning in the recommender system, we compare the target contracted company with the non-contracted logistics company to learn the intention score. The assumption is that the intention score of the contracted company is higher than those not contracted. In this way, we can generate observed comparison pairs $\mathcal{P}_o$ (e.g., pair$\{A, E\}$). Using the Bayesian Personalized Ranking (BPR) [40], we define the loss function for this task as $\mathcal{L}_{C_o} = -\sum_{i,k \in \mathcal{P}_o}(ln\sigma(score_{m,c_i}^t - score_{m,c_k}^t))$, where c_k denotes the non-contracted logistics companies.

3.2.3 **Data Augmentation-based Comparison.** For comparisons between non-contracted logistics companies, we generate two types of augmentation comparison pairs to mitigate the bias caused by data sparsity (i.e., only limited contracted logistics companies). Firstly, we utilize the serving relations to guide the comparison among currently served logistics companies. That is, merchants are assumed to have a higher intention toward logistics companies they use more frequently. The loss function for this task is defined as $\mathcal{L}_{C_{a_1}} = -\sum_{i,k \in \mathcal{P}_{a_1}}(ln\sigma(score_{m,c_i}^t - score_{m,c_k}^t))$, where $\mathcal{P}_{a_1}$ represents comparison pairs (e.g., pair$\{B, C\}$) among currently used logistics companies.

Secondly, for logistics companies (e.g., E and F in Fig 4) without interaction with a merchant, we create a soft label comparison between them to better understand the merchant's intentions toward these logistics companies. Specifically, if the merchant prepares to contract with the target logistics company (e.g., A), it indicates a strong intention toward the target. Under such conditions, if logistics company E is more similar to the target logistics company A than logistics company F, we assume that the merchant's intention score for E is greater than that of F. In this way, we obtain multiple soft label-based comparison pairs denoted as $\mathcal{P}_{soft}$. The loss function for this task is defined as $\mathcal{L}_{C_{a_2}} = -\sum_{i,j \in \mathcal{P}_{soft}}(ln\sigma(score_{m,l_i}^t - score_{m,l_j}^t))$. Finally, the loss function of the intention learning model is formulated as $\mathcal{L}_I = \mathcal{L}_{C_o} + \mu_1\mathcal{L}_{C_{a_1}} + \mu_2\mathcal{L}_{C_{a_2}}$, where μ_1 and μ_2 are trade-off parameters.

3.3 Future Pattern Discovery Model

In addition to contracting, the platform is concerned about long-term usage after contracting. However, individual merchants have different and complex patterns, and there may be unseen future patterns that are not captured in our existing data. Our solution is motivated by works in multi-interest recommendations [3, 24], which extract multiple interests and combine them to represent users' preferences. We assume that any future pattern can be represented by the fusion of basic patterns. These basic patterns are common patterns, such as increase and decrease, that are relatively easier to model. The structure is illustrated in Fig. 5. We first set x basic patterns with identical model structures but different parameters. Then we use past merchant embeddings and the attention mechanism to obtain the final combination result for usage prediction. Lastly, to ensure the different patterns are as divergent as possible, we introduce restrictions on the output of each pattern.

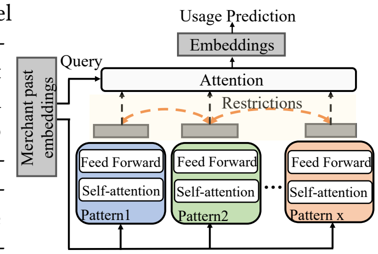

Fig. 5: Future Pattern Discovery Model

3.3.1 **Basic Pattern.** We establish x basic patterns, each of which generates a future prediction based on the past embeddings of merchants. Within each basic pattern, we utilize a self-attention mechanism [48] to compute the importance of different parts of the past embedding for future usage. Formally, we project the past merchant embedding h_m^t to a query vector with a linear projection $W_{q,b}$. The query vector is defined as $Q_b^t(m) = W_{q,b}h_m^t$. Similarly, the key vector and value vector are defined as $K_b^t(m) = W_{k,b}h_m^t$ and $V_b^t(m) = W_{v,b}h_m^t$. The output of the self-attention is defined as $h_{m,att}^t = softmax(\frac{Q_b^t(m)K_b^t(m)^T}{\sqrt{\delta_k}})V_b^t(m)$, where δ_k represents the embedding size. It should be mentioned that time t denotes the contracting time of the merchant. Subsequently, we feed the $h_{m,att}^t$ into a feed-forward neural network to obtain the final embedding of the basic pattern, which is defined as $h_{m,b}^t = \sigma(W_b h_{m,att}^t)$.

3.3.2 **Combination of Different Patterns.** After obtaining the output of each basic pattern, we use the merchant's past embedding h_m^t as a query to calculate the importance of each pattern [48]. The query vector is defined as $Q_p^t(m) = W_{q,p}h_m^t$. We project the concatenated embeddings from all the patterns into key and value vectors, which are defined as $K_p^t(m) = W_{k,p}([h_{m,b_1}^t \oplus ... \oplus h_{m,b_x}^t])$ and $V_p^t(m) = W_{v,p}([h_{m,b_1}^t \oplus ... \oplus h_{m,b_x}^t])$, respectively. Then the merchant's future embedding is the weighted sum of the value vector, which is defined as $h_{m,f}^t = softmax(\frac{Q_p^t(m)K_p^t(m)^T}{\sqrt{\delta_k}})V_p^t(m)$ where δ_k represents embedding size. Finally, the merchant's future embedding is fed into a multi-layer perceptron for usage prediction, which is defined as $\hat{y}_m^t = \sigma(W_u h_{m,f}^t)$. The loss function of this task is defined as $\mathcal{L}_{usage} = \frac{1}{|N|}\sum_{m \in M}(\hat{y}_m^t - y_m^t)^2$, where $|N|$ is the number of observed data and y_m^t is the ground truth usage of contracted merchant m whose contracting time is t.

3.3.3 **Restrictions.** To ensure the uniqueness of each pattern, we constrain the output of each pattern. Specifically, we compute the mutual information between the outputs of any two basic patterns with the aim to minimize it. The loss function for this task is defined as $\mathcal{L}_{MI} = \sum_{i,j \in \mathcal{P}_{basic}} MI(h_{m_i,f}^t, h_{m_j,f}^t)$, where $\mathcal{P}_{basic}$ is basic pattern model pairs. Finally, the loss function is formulated as $\mathcal{L}_u = \mathcal{L}_{usage} + \beta\mathcal{L}_{MI}$, where β is a trade-off parameter.

4 EXPERIMENT

4.1 Experimental Setup

4.1.1 **Data Description.** Our work is based on the JingDong (JD) e-commerce platform, where merchants manage their businesses online. Once users place orders, merchants have the freedom to choose their preferred logistics companies (e.g., JingDong Logistics (a sibling company of JD e-commence platform) [30], Shunfeng [43], and Yuantong [12]) to ship and deliver the orders to users. Therefore, our work relies on the following data. ***Merchant Information***: The platform records the basic information of each merchant, including merchant id, location, and industry; If merchants contract with JD Logistics, the contracting information is included, such as contract status and contract time. We select merchants whose locations are in three provinces in China. In addition, the dataset includes more than 60 types of industry. ***Waybill Data***: In addition to the merchant information, the platform records the waybill information of each merchant, which includes the start time, start location, delivery time, delivery location, and logistics company. In total, the dataset includes millions of waybill records from September 2021 to December 2022 and more than 30 different logistics companies.

4.1.2 **Data Pre-processing.** For the contract prediction task, we select merchants contracted with JD Logistics whose contract time is from January to July 2022 as positive samples and randomly select non-contracted merchants as negative samples. We set two weeks as a time step in the pre-trained model. We take the pre-trained embedding of each merchant before the contracting time as the input of the intention learning model to predict whether they will contract with JD Logistics. We calculate binary contract results based on intention scores. If the predicted position of JD Logistics falls within top-k, we set the predicted contract results as true. We set k as 3 because the data shows that the 3 most frequently used logistics companies by 80% of merchants can cover all their orders. For the future usage prediction, we use the order ratio of using JD Logistics within five months after contracting as ground truth. Since the number of newly contracted merchants is relatively small, we set different ratios of training to testing to validate the performance of different sizes of contracted merchants.

4.1.3 **Implementation Details.** We implement our method and baselines with Pytorch 1.9.0 in Python 3.6 environment and train it with 16GB memory and Intel(R) Xeon(R) CPU E5-2680 v4 @ 2.40GHz (CPU). We apply the Adam optimizer with a learning rate of 1e-5 and set the batch size to 32. For the hyper-parameters, we set the embedding size as 128 and the layer number of heterogeneous graph modeling as two. The trade-off parameter γ is tested with values of [0.5, 0.75, 1, 1.25, 1.5] and achieves the best performance with a value of 1. On the other hand, μ_1, μ_2, and β are tested with values of [0.1, 0.2, 0.3, 0.4, 0.5]. The best performance is achieved with values of 0.4, 0.1, and 0.2, respectively. Additionally, the basic pattern number is set as 3 as it has the best performance.

4.1.4 **Metrics and Baselines.** For contract prediction, we use recall and F1 scores as metrics [54], which are widely used in binary prediction tasks. For usage prediction, we use Mean Absolute Error (MAE) to evaluate the accuracy of the usage prediction.

We compare our model with two categories of baselines: audience expansion methods with and without pre-training. (1) **MLP:** We use the accumulated features and MLP for audience expansion. (2) **MetaHeac [55]:** MetaHeac is an audience expansion method that considers relations between different tasks. Considering there is no multiple training task in our work, we remove the meta-learning framework and utilize the accumulated features as input. (3) **MLP + pre-trained embeddings:** We utilize the accumulated features to construct a graph and get the pre-trained embedding for MLP. (4) **Pinterest [8]:** This method first trains a global user embedding, followed by using a scoring model to identify potential users. (5) **Hubble [56]:** Hubble is a two-stage audience expansion model using the GNN to obtain pre-trained embeddings. (6) **MetaHeac [55] + pre-trained embeddings:** In this setting, we utilize the pre-trained embedding from the accumulated knowledge graph to train this model.

Table 1: Contract Prediction Performance of Different Methods. Bold scores are used for the largest values.

Method	Training/testing ratio = 8:2		Training/testing ratio = 2:8	
	Recall ↑	F1 score ↑	Recall ↑	F1 score ↑
MLP	0.521 ± 0.093	0.543 ± 0.057	0.442 ± 0.150	0.440 ± 0.070
MetaHeac	0.623 ± 0.070	0.583 ± 0.037	0.472 ± 0.085	0.470 ± 0.043
MLP + pre-training	0.636 ± 0.010	0.581 ± 0.004	0.524 ± 0.080	0.529 ± 0.044
Pinterest	0.479 ± 0.076	0.4904 ± 0.053	0.415 ± 0.192	0.436 ± 0.130
Hubble	0.822 ± 0.091	0.6673 ± 0.034	0.658 ± 0.041	0.597 ± 0.092
MetaHeac + pre-training	0.721 ± 0.082	0.608 ± 0.026	0.545 ± 0.098	0.536 ± 0.038
LOGAE-TKG	**0.928 ± 0.018**	**0.774 ± 0.011**	**0.868 ± 0.066**	**0.697 ± 0.026**

4.2 Overall Performance

4.2.1 **Contract Prediction Performance.** We compare our approach with the baselines, and the comparison results are in Table 1.
Comparison to Audience Expansion Methods without Pre-training. Our model outperforms baselines without pre-training. This can be attributed to the fact that while the baselines are trained using accumulated features, our model can learn more comprehensively from the logistics knowledge graph, which provides a solid knowledge base for contract prediction.
Comparison to Audience Expansion Methods with Pre-training. Compared to audience expansion methods that use pre-training, our model yields better performance. Although these models use pre-training to obtain merchant embedding, they fail to account for temporal evolution (e.g., dynamic interactions between merchants and logistics companies). Furthermore, when comparing audience expansion methods without pre-training to those with pre-training, we observe that the latter are more efficient, even when temporal dynamics are not taken into account. This indicates that pre-training enables the model to acquire more knowledge and helps mitigate the risk of overfitting.
Performance with Different Ratios of Training to Testing sets. Table 1 demonstrates that as the quantity of training data decreases, the performance of all models deteriorates. However, the relative improvement over the baseline increases, indicating that our model remains effective even with a smaller dataset. One potential reason for this could be that our model learns knowledge from the logistics knowledge graph. This graph contains information from various merchants, including those who are not contracted merchants. These non-contracted merchants contribute valuable insights and behaviors that enhance the richness of knowledge available to our model.

4.2.2 **Future Usage Prediction Performance.** For the future usage prediction task, we compare our method with variants of our model and a baseline (i.e., MLP+ pre-trained embeddings). Fig. 6 displays results for the fourth and fifth months after contracting. Notably, our model improves MAE at least by about 21.32% compared to the baseline. In addition, when we remove the restrictions in the future pattern discovery model (i.e., w/o MI), the performance deteriorates. When we implement different numbers of basic patterns (i.e., 2 patterns (w 2BP) and 4 patterns (w 4BP)), the performance varies. Those verify the effectiveness of our design and show the impact of varying the number of basic patterns.

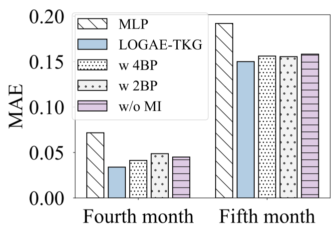

Fig. 6: Usage Prediction Performance

4.3 Ablation Study

The Effect of the Pre-trained Model. To understand the role of the temporal knowledge graph pre-trained model, we compare our model LOGAE-TKG with its two variants, as shown in Fig. 7. When the merchant-logistics relation prediction pre-training task is removed (i.e., w/o MLRP), the performance becomes worse, and when the whole temporal knowledge graph pre-trained model is removed (i.e., w/o P), the performance deteriorates even more. These results demonstrate that (i) the model can gain more comprehensive knowledge of logistics from the logistics knowledge graph with a large number of contracted and non-contracted merchants, thereby providing a robust knowledge base for contract prediction; (ii) two pre-training tasks are effective to capture the dynamics of intention of the merchants and the characteristics of the logistics companies.

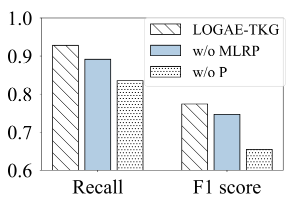

Fig. 7: The Effect of the Pre-training

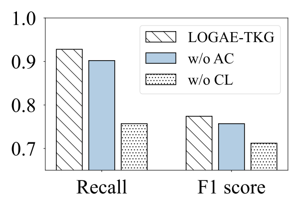

Fig. 8: The Effect of the Intention Learning Model

The Effect of Intention Learning Model. To get a better understanding of the intention learning model, we compare our model LOGAE-TKG with its two variants, as shown in Fig. 8. Initially, we remove the data augmentation-based comparison in the intention learning model (i.e., w/o AC), resulting in a decline in performance compared to our model. Subsequently, when we utilize a classifier (i.e., MLP) to replace the observed and data augmentation-based comparisons (i.e., w/o CL), the performance decreases significantly. These results illustrate that the intention comparison in the intention learning model plays a crucial role in capturing the differing intentions of merchants toward various logistics companies.

4.4 Online A/B Testing

We implement and deploy the LOGAE-TKG in three provinces at the JD e-commerce platform to improve the business performance of JD Logistics. Fig. 9 shows the heat map of the distribution of merchants involved in the deployment. To verify the benefits to the platform, we conduct online A/B testing experiments within three months. Specifically, we use different methods to select potential merchants for contracting, send out messages, and track the results. We use the same size candidates for two groups (i.e., the control group and the treatment group). The control group is a deployed model that adapts meta-path based heterogeneous graph learning [51], while the treatment group uses our designed LOGAE-TKG. To evaluate the performance online, we use the contract conversion rate as the metric. A higher contract conversion rate indicates a higher quality of identified potential merchants. The result shows that our LOGAE-TKG achieves a 13.6% improvement compared to the currently deployed model.

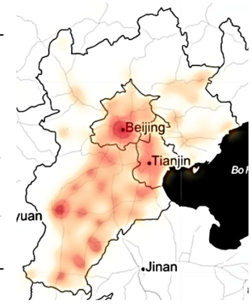

Fig. 9: Deployment Heatmap

5 RELATED WORK

Audience Expansion The audience expansion aims to identify more potential audiences in a specific campaign [35]. Existing works include similarity-based and model-based methods. In the similarity-based methods [4, 11, 26, 33, 34, 42], researchers use pre-defined similarity functions to measure the similarity of potential users and seed users. Some model-based methods [21, 39] directly train a customized model for a specific campaign without pre-training. Other model-based methods [8, 29, 55, 56] decompose the problem into including pre-training and expansion stage. These methods are not suitable for logistics audience expansion due to the multiple complex factors in the logistics scenario and the requiring long-term logistics service usage instead of one-time promotions.

Knowledge Graph Embedding. Representation learning for knowledge graphs has been extensively studied recently [20, 50], including static knowledge graph embedding and temporal knowledge graph embedding. To learn embeddings in the static heterogeneous knowledge graphs, tensor factorization-based methods [19, 37], relation-based methods [2, 7, 36, 44], and graph neural networks [16] based methods [41, 47, 51] are proposed. In order to capture temporal evolution in KGs, some static knowledge graph embedding methods are extended for temporal modeling [14, 15, 23]. In addition, some works [22, 25, 38] combine heterogeneous graph [28, 52] and temporal modeling to capture both structural and temporal dependencies.

6 CONCLUSION

This work aims to address audience expansion problems in the logistics system. We design a novel audience expansion model via temporal knowledge graph. We evaluate and deploy our method in the JD e-commerce platform. Both offline and online experiment results demonstrate the effectiveness of our model.

7 ACKNOWLEDGEMENT

This work was supported in part by Science and Technology Innovation 2030 - Major Project 2021ZD0114200.

REFERENCES

[1] Amazon. 2023. Amazon. https://www.amazon.com/.

[2] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems* 26 (2013).

[3] Yukuo Cen, Jianwei Zhang, Xu Zou, Chang Zhou, Hongxia Yang, and Jie Tang. 2020. Controllable multi-interest framework for recommendation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2942–2951.

[4] Gromit Yeuk-Yin Chan, Tung Mai, Anup B Rao, Ryan A Rossi, Fan Du, Cláudio T Silva, and Juliana Freire. 2021. Interactive Audience Expansion On Large Scale Online Visitor Data. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 2621–2631.

[5] Carlos Daganzo. 2005. *Logistics systems analysis*. Springer Science & Business Media.

[6] Songgaojun Deng, Huzefa Rangwala, and Yue Ning. 2020. Dynamic knowledge graph based multi-event forecasting. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1585–1595.

[7] Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2d knowledge graph embeddings. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 32.

[8] Stephanie deWet and Jiafan Ou. 2019. Finding users who act alike: transfer learning for expanding advertiser audiences. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2251–2259.

[9] Yi Ding, Ling Liu, Yu Yang, Yunhuai Liu, Desheng Zhang, and Tian He. 2021. From conception to retirement: a lifetime story of a 3-year-old wireless beacon system in the wild. *IEEE/ACM Transactions on Networking* 30, 1 (2021), 47–61.

[10] Yi Ding, Yu Yang, Wenchao Jiang, Yunhuai Liu, Tian He, and Desheng Zhang. 2021. Nationwide deployment and operation of a virtual arrival detection system in the wild. In *Proceedings of the 2021 ACM SIGCOMM 2021 Conference*. 705–717.

[11] Khoa D Doan, Pranjul Yadav, and Chandan K Reddy. 2019. Adversarial factorization autoencoder for look-alike modeling. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 2803–2812.

[12] YTO Express. 2023. YTO Express. https://www.yto.net.cn/English/en/index.html.

[13] FedEx. 2023. FedEx. https://www.fedex.com/en-us/home.html.

[14] Alberto García-Durán, Sebastijan Dumančić, and Mathias Niepert. 2018. Learning sequence encoders for temporal knowledge graph completion. *arXiv preprint arXiv:1809.03202* (2018).

[15] Rishab Goel, Seyed Mehran Kazemi, Marcus Brubaker, and Pascal Poupart. 2020. Diachronic embedding for temporal knowledge graph completion. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 3988–3995.

[16] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems* 30 (2017).

[17] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[18] Ziniu Hu, Yuxiao Dong, Kuansan Wang, and Yizhou Sun. 2020. Heterogeneous graph transformer. In *Proceedings of the web conference 2020*. 2704–2710.

[19] ByungSoo Jeon, Inah Jeon, Lee Sael, and U Kang. 2016. Scout: Scalable coupled matrix-tensor factorization-algorithm and discoveries. In *2016 IEEE 32nd International Conference on Data Engineering (ICDE)*. IEEE, 811–822.

[20] Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and S Yu Philip. 2021. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE transactions on neural networks and learning systems* 33, 2 (2021), 494–514.

[21] Jinling Jiang, Xiaoming Lin, Junjie Yao, and Hua Lu. 2019. Comprehensive audience expansion based on end-to-end neural prediction. In *CEUR Workshop Proceedings*, Vol. 2410. CEUR Workshop Proceedings.

[22] Woojeong Jin, Meng Qu, Xisen Jin, and Xiang Ren. 2019. Recurrent event network: Autoregressive structure inference over temporal knowledge graphs. *arXiv preprint arXiv:1904.05530* (2019).

[23] Julien Leblay and Melisachew Wudage Chekol. 2018. Deriving validity time in knowledge graph. In *Companion proceedings of the the web conference 2018*. 1771–1776.

[24] Chao Li, Zhiyuan Liu, Mengmeng Wu, Yuchi Xu, Huan Zhao, Pipei Huang, Guoliang Kang, Qiwei Chen, Wei Li, and Dik Lun Lee. 2019. Multi-interest network with dynamic routing for recommendation at Tmall. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 2615–2623.

[25] Zixuan Li, Xiaolong Jin, Wei Li, Saiping Guan, Jiafeng Guo, Huawei Shen, Yuanzhuo Wang, and Xueqi Cheng. 2021. Temporal knowledge graph reasoning based on evolutional representation learning. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 408–417.

[26] Haishan Liu, David Pardoe, Kun Liu, Manoj Thakur, Frank Cao, and Chongzhe Li. 2016. Audience expansion for online social network advertising. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 165–174.

[27] Kangzheng Liu, Feng Zhao, Guandong Xu, Xianzhi Wang, and Hai Jin. 2022. Temporal Knowledge Graph Reasoning via Time-Distributed Representation Learning. In *2022 IEEE International Conference on Data Mining (ICDM)*. IEEE, 279–288.

[28] Wei Liu, Yi Ding, Shuai Wang, Yu Yang, and Desheng Zhang. 2022. Para-Pred: Addressing Heterogeneity for City-Wide Indoor Status Estimation in On-Demand Delivery. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 3407–3417.

[29] Zhining Liu, Xiao-Fan Niu, Chenyi Zhuang, Yize Tan, Yixiang Mu, Jinjie Gu, and Guannan Zhang. 2020. Two-stage audience expansion for financial targeting in marketing. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2629–2636.

[30] JingDong Logistics. 2023. JingDong Logistics. https://www.jdl.com/.

[31] Wenjun Lyu, Haotian Wang, Zhiqing Hong, Guang Wang, Yu Yang, Yunhuai Liu, and Desheng Zhang. 2023. REDE: Exploring Relay Transportation for Efficient Last-mile Delivery. In *2023 IEEE 39th International Conference on Data Engineering (ICDE)*. IEEE, 3003–3016.

[32] Wenjun Lyu, Kexin Zhang, Baoshen Guo, Zhiqing Hong, Guang Yang, Guang Wang, Yu Yang, Yunhuai Liu, and Desheng Zhang. 2022. Towards Fair Workload Assessment via Homogeneous Order Grouping in Last-mile Delivery. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 3361–3370.

[33] Qiang Ma, Eeshan Wagh, Jiayi Wen, Zhen Xia, Robert Ormandi, and Datong Chen. 2016. Score look-alike audiences. In *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*. IEEE, 647–654.

[34] Qiang Ma, Musen Wen, Zhen Xia, and Datong Chen. 2016. A sub-linear, massive-scale look-alike audience extension system a massive-scale look-alike audience extension. In *Workshop on Big Data, Streams and Heterogeneous Source Mining: Algorithms, Systems, Programming Models and Applications*. PMLR, 51–67.

[35] Ashish Mangalampalli, Adwait Ratnaparkhi, Andrew O Hatch, Abraham Bagherjeiran, Rajesh Parekh, and Vikram Pudi. 2011. A feature-pair-based associative classification approach to look-alike modeling for conversion-oriented user-targeting in tail campaigns. In *Proceedings of the 20th international conference companion on World wide web*. 85–86.

[36] Maximilian Nickel, Volker Tresp, Hans-Peter Kriegel, et al. 2011. A three-way model for collective learning on multi-relational data.. In *Icml*, Vol. 11. 3104482–3104584.

[37] Namyong Park, Byungsoo Jeon, Jungwoo Lee, and U Kang. 2016. Bigtensor: Mining billion-scale tensor made easy. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*. 2457–2460.

[38] Namyong Park, Fuchen Liu, Purvanshi Mehta, Dana Cristofor, Christos Faloutsos, and Yuxiao Dong. 2022. Evokg: Jointly modeling event time and network structure for reasoning over temporal knowledge graphs. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*. 794–803.

[39] Yan Qu, Jing Wang, Yang Sun, and Hans Marius Holtan. 2014. Systems and methods for generating expanded user segments. US Patent 8,655,695.

[40] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2014. Bayesian personalized ranking from implicit feedback. In *Proc. of Uncertainty in Artificial Intelligence*. 452–461.

[41] Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *The Semantic Web: 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3–7, 2018, Proceedings 15*. Springer, 593–607.

[42] Jianqiang Shen, Sahin Cem Geyik, and Ali Dasdan. 2015. Effective audience extension in online advertising. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2099–2108.

[43] ShunFeng. 2023. ShunFeng. https://www.sf-express.com/chn/sc.

[44] Richard Socher, Danqi Chen, Christopher D Manning, and Andrew Ng. 2013. Reasoning with neural tensor networks for knowledge base completion. *Advances in neural information processing systems* 26 (2013).

[45] Taobao. 2023. Taobao. https://www.taobao.com.

[46] UPS. 2023. UPS. https://www.ups.com/us/en/Home.page.

[47] Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, and Partha Talukdar. 2019. Composition-based multi-relational graph convolutional networks. *arXiv preprint arXiv:1911.03082* (2019).

[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[49] Hai Wang, Shuai Wang, Yu Yang, and Desheng Zhang. 2023. GCRL: Efficient Delivery Area Assignment for Last-mile Logistics with Group-based Cooperative Reinforcement Learning. In *2023 IEEE 39th International Conference on Data Engineering (ICDE)*. IEEE, 3522–3534.

[50] Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. 2017. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering* 29, 12 (2017), 2724–2743.

[51] Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S Yu. 2019. Heterogeneous graph attention network. In *The world wide web conference*. 2022–2032.

[52] Hua Yan, Shuai Wang, Yu Yang, Baoshen Guo, Tian He, and Desheng Zhang. 2022. O2-siterec: Store site recommendation under the o2o model via multi-graph attention networks. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. IEEE, 525–538.

[53] Yu Yang, Yi Ding, Dengpan Yuan, Guang Wang, Xiaoyang Xie, Yunhuai Liu, Tian He, and Desheng Zhang. 2020. Transloc: transparent indoor localization with uncertain human participation for instant delivery. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*. 1–14.

[54] Zhi-Hua Zhou. 2021. *Machine learning*. Springer Nature.

[55] Yongchun Zhu, Yudan Liu, Ruobing Xie, Fuzhen Zhuang, Xiaobo Hao, Kaikai Ge, Xu Zhang, Leyu Lin, and Juan Cao. 2021. Learning to expand audience via meta hybrid experts and critics for recommendation and advertising. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 4005–4013.

[56] Chenyi Zhuang, Ziqi Liu, Zhiqiang Zhang, Yize Tan, Zhengwei Wu, Zhining Liu, Jianping Wei, Jinjie Gu, Guannan Zhang, Jun Zhou, et al. 2020. Hubble: An industrial system for audience expansion in mobile marketing. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2455–2463.