Persona Identification in E-Commerce with Scarce Labels and In-Context Graph Learning

Anjali Gupta*
Indian Institute of Technology Delhi
Computer Science and Engineering
New Delhi, Delhi, India
anjali@cse.iitd.ac.in

Prashant Kumar*
Indian Institute of Technology Delhi
School of Infomation Technology
New Delhi, Delhi, India
prashantk.nan@gmail.com

Aniket Mishra
Indian Institute of Technology Delhi
Computer Science and Engineering
New Delhi, Delhi, India
aniketmishra1002@gmail.com

Abhishek Singh
Flipkart Internet Pvt Ltd
Data Science
Bangalore, Karnataka, India
abhishek.kumars@flipkart.com

Surender Kumar
Flipkart Internet Pvt Ltd
Data Science
Bangalore, Karnataka, India
surender.k@flipkart.com

Muthusamy Chelliah
Flipkart Internet Pvt Ltd
Data Science
Bangalore, Karnataka, India
chelgeetha@gmail.com

Abhijnan Chakraborty
Indian Institute of Technology Kharagpur
Computer Science and Engineering
Kharagpur, West Bengal, India
abhijnan@cse.iitkgp.ac.in

Sayan Ranu
Indian Institute of Technology Delhi
Computer Science and Engineering
New Delhi, Delhi, India
sayanranu@cse.iitd.ac.in

Abstract

User personas play a crucial role in shaping shopping behavior, making persona identification an essential task for e-commerce platforms to personalize user experiences. However, this problem remains highly challenging due to several key factors: (1) users often exhibit multiple personas simultaneously (e.g., a fashion enthusiast who is also a sports lover), (2) labeled data is scarce, as annotation requires extensive human supervision, (3) real-world user-product interaction data is inherently noisy, as accounts may be shared among multiple individuals, complicating persona assessment, and (4) user-product interactions form a dynamic, heterogeneous bipartite graph where product features are diverse, and personas evolve due to shifts in interests, seasons, and external events.

In this work, we study the problem of persona identification on an extensive real-world user-product interaction dataset spanning six months (August 2023 to January 2024), capturing user behavior influenced by real-world factors such as seasonal changes, festivals (e.g., Christmas, New Year), and major sales events. To model persona identification in this evolving interaction graph, we reformulate the multi-label node classification task as a link prediction problem, enabling a structured decoupling of user and persona representations. To this end, we propose **TriPer**, a novel **TRI**partite graph neural network specifically designed to enhance multi-label **PER**sona classification along with *in-context inference* capabilities. Extensive evaluations on our real-world dataset demonstrate that TriPer achieves high predictive accuracy in user persona identification, exhibits strong generalization over time, and effectively learns from limited labeled data, outperforming state-of-the-art baselines.

*Denotes equal contribution

KDD '25, Toronto, ON, Canada

ACM ISBN 979-8-4007-1454-2/2025/08
https://doi.org/10.1145/3711896.3737080

CCS Concepts

• **Applied computing** → **Online shopping**; • **Computing methodologies** → *Lifelong machine learning*.

Keywords

E-commerce, Personalization, Heterogeneous Graphs, User Persona, Scarce Labels, Generalization, Graph Neural Networks

ACM Reference Format:
Anjali Gupta, Prashant Kumar, Aniket Mishra, Abhishek Singh, Surender Kumar, Muthusamy Chelliah, Abhijnan Chakraborty, and Sayan Ranu. 2025. Persona Identification in E-Commerce with Scarce Labels and In-Context Graph Learning . In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (KDD '25), August 3–7, 2025, Toronto, ON, Canada.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3711896.3737080

KDD Availability Link:
The source code of this paper has been made publicly available at https://doi.org/10.5281/zenodo.15543277.

1 Introduction

In the fast-evolving e-commerce landscape, businesses strive to understand customer preferences for tailored personalized experiences, leveraging customer data to decode behaviors and anticipate expectations [3, 4, 8, 11, 12, 33, 42]. In addition to personality, demographic factors like age, gender, and purchasing power also influence customer decisions [24]. For instance, younger shoppers may prioritize the latest trends, while older buyers value quality and materials. Education, occupation, and income may further shape purchasing behaviors. Given the strong link between personality, demographics, and user interests, integrating these aspects into user embedding models can more accurately capture preferences and enhance personalization. *User personas* – fictional representations of customer segments based on demographics, behaviors, motivations, and data – are conceptualized to capture this phenomenon. In simpler terms, a persona refers to a label that encapsulates a distinct customer type with a focus on specific product categories or shopping behaviors. These personas help e-commerce businesses segment their customer base and create personalized marketing strategies, product offerings, and user experiences to cater to each group's unique needs and preferences.

1.1 Challenges

Technically, given a user-product interaction graph, the problem of persona identification maps to a *multi-class node classification* problem, where each user is assigned one class (persona) from a predefined set of personas. However, several problem-specific challenges arise which the generic multi-class node classification does not address:

- **Multiple personas per user:** User preferences in e-commerce are often complex and dynamic, so a single user might not fit neatly into one persona. For example, a user may purchase high-end fashion items (Luxury Shopper) but may also frequently look for eco-friendly brands (Sustainable Shopper). Thus, we must identify multiple personas for each user, reflecting different facets of their shopping behavior, making the problem a *multi-label classification* problem.
- **Label scarcity:** Labeling users based on their extensive purchase behavior into persona classes is labor, cost, and time intensive. It additionally requires *expert domain knowledge* of a persona and the wide range of products that represent it. Hence, the algorithm must be designed to learn from low-volume labeled data and demonstrate strong generalizability. Furthermore, due to the dynamic nature of personas, even existing labeled users may require re-labeling at periodic intervals to reassess and adjust a model to the time-varying nature of personas.
- **In-context generalization to unseen personas:** Given the volume and dynamic nature of user preferences, e-commerce businesses must continuously adapt to seasonal trends, emerging product categories, and major global events. For instance, during a global sports event like the Olympics, a retailer might observe a surge in demand for sports merchandise and introduce a new "Olympic Enthusiast" persona to better capture the preferences of users engaging with related products. However, traditional node classification models, which rely on a softmax layer over predefined class labels, are inherently constrained by their dependency on a fixed set of persona definitions. Introducing new labels in such models would require retraining from scratch, making them ill-suited for rapidly evolving personalization needs.

To overcome this limitation, we leverage *in-context* examples—small sets of users explicitly labeled with new personas—allowing the model to infer these novel personas for a broader user base without retraining. In the Olympics example, providing in-context examples could involve annotating a small subset of users who have recently purchased sports jerseys, fitness accessories, or event tickets with the "Olympic Enthusiast" persona. By integrating these labeled users into the inference graph, our approach propagates persona information based on shared behavioral patterns, enabling the model to generalize effectively. This in-context learning capability ensures that businesses can dynamically refine their personalization strategies and respond swiftly to changing market conditions and customer behaviors.

1.2 Contributions

In this paper, we introduce TriPer which is built on the following core innovations to address the aforementioned challenges:

- **Novel problem formulation:** We formulate the problem of persona identification from user-product interaction graphs with two key requirements embedded in the formulation: (1) the model must predict both the number and types of personas associated with each user based on their purchase data, reflecting that individual users may exhibit multiple distinct personas, and (2) the model must inductively generalize to unseen personas without requiring retraining from scratch.
- **Algorithm design:** We map persona identification to a link prediction problem on a tripartite graph connecting users, products, and personas. To enable accurate link prediction on this graph, we introduce TriPer, a **TRI**partite Gnn designed specifically for multi-label **PER**sona identification on heterogeneous graphs. This novel approach of modeling the data as a tripartite graph and framing the problem as link prediction between user and persona nodes offers two key advantages: (1) it enables persona identification without forcing personas to compete with each other, which would occur in traditional node classification with a softmax layer, and (2) by modeling personas as nodes rather than labels, the model's parameter size remains independent of the number of personas, allowing in-context generalization to new personas introduced over time.
- **Extensive evaluation on real-world dataset:** In this work, we utilize a comprehensive six-month dataset of user-product purchase information, enhanced with expert-annotated persona labels. We evaluate TriPer's performance against established techniques for node classification on heterogeneous graphs. Our experiments demonstrate TriPer's superior performance in persona identification tasks and generalization robustness across six consecutive months without retraining. To our knowledge, this represents the first study of persona identification conducted on a real-world dataset of this scale.

2 Problem Formulation and Preliminaries

In this section, we present the key concepts underlying our work, formally define the problem of in-context persona identification, and introduce the large-scale dataset that serves as the foundation for our empirical evaluation and insights.

2.1 E-commerce Graph

We define the e-commerce graph over user-product interactions as a heterogeneous bipartite graph $G = (\mathcal{U}, \mathcal{P}, \mathcal{E}, \mathbf{X})$, where $\mathcal{U} = \{u_i\}_{i=1}^{m}$

represents user nodes and $\mathcal{P} = \{p_j\}_{j=1}^{n}$ are product nodes. The edge set $\mathcal{E} \subseteq \mathcal{U} \times \mathcal{P}$ denotes interactions between users and products, such as purchases, ratings, or clicks. Each user node u_i is associated with a feature vector $\mathbf{x}_{u_i} \in \mathbb{R}^{d_u}$, where d_u is the dimensionality of the user attributes (e.g., age, gender, residence, marital status, parental status). Similarly, each product node p_j is associated with a feature vector $\mathbf{x}_{p_j} \in \mathbb{R}^{d_p}$, where d_p is the dimensionality of the product attributes (e.g., price, category, brand, weight, color), along with textual description, which we embed into a dense vector s_{p_j} using any off-the-shelf text embedding encoder. (See Figure 8 in the Appendix as an example). Let $\mathbf{X}_{\mathcal{U}} = \{[\mathbf{x}_{u_i}]\}_{i=1}^{m}$ represent the combined feature attributes of all users in $\mathcal{U}$, and $\mathbf{X}_{\mathcal{P}} = \{[\mathbf{x}_{p_j}, \mathbf{s}_{p_j}]\}_{j=1}^{n}$ represent the combined feature attributes of all products in $\mathcal{P}$. The complete feature attributes of the graph are denoted as $\mathbf{X} = \mathbf{X}_{\mathcal{U}} \cup \mathbf{X}_{\mathcal{P}}$.

2.2 Problem Formulation

Our problem has two primary objectives. First, we aim to develop a persona-identification model trained on a persona-annotated user-product interaction graph. Second, as outlined in § 1, the model must exhibit in-context generalization to unseen personas. Specifically, when provided with in-context examples of users labeled with new personas, the model should be capable of annotating unseen users within a joint persona space that includes both previously seen personas and newly introduced ones with requiring to retrain the model. We formally define these objectives as follows.

Problem 1 (Persona Identification). *We use the notation $\mathbf{Y} \in \{0,1\}^{m \times k}$ to denote the persona labels for each user, where k is the total number of predefined personas at the time of training. Each user $u \in \mathcal{U}$ is associated with a subset of these personas, which we denote as $\mathbf{Y}[i] \subseteq \{c_1, \ldots, c_k\}$.*

Given a training set $G = (\mathcal{U}, \mathcal{P}, \mathcal{E}, \mathbf{X})$ and corresponding ground-truth persona labels $\mathbf{Y}$, our objective is to learn a model $\mathcal{M}$, parameterized by Θ, that can predict the multi-label set of personas for any unseen user node u_i. Formally, we aim to minimize a suitable multi-label classification loss function $\mathcal{J}$ such that:

$$\min_{\Theta} \mathcal{J}\left(\mathcal{M}(G, u; \Theta), \mathbf{Y}[i]\right) \tag{1}$$

$$\text{s.t.} \quad \widehat{\mathbf{Y}}[i] \approx \mathbf{Y}[i] \quad \text{where,} \quad \mathcal{M}(G, u_i; \Theta) = \widehat{\mathbf{Y}}[i]. \tag{2}$$

Here, $\widehat{\mathbf{Y}}[i]$ represents the predicted set of persona labels for user u_i, and $\mathcal{J}$ denotes the loss function, which we will formally define later.

The problem of in-context learning is formally stated as follows:

Problem 2 (In-context generalization to unseen personas). *Let $G_T = (\mathcal{U}_T, \mathcal{P}_T, \mathcal{E}_T, \mathbf{X}_T)$ be the training graph, where $\mathbf{Y}_T \in \{0,1\}^{|\mathcal{U}_T| \times k}$ represents the ground-truth persona labels used to train the model $\mathcal{M}$. For in-context learning, we are given:*

- *An **inference** graph $G_I = (\mathcal{U}_I, \mathcal{P}_I, \mathcal{E}_I, \mathbf{X}_I)$, where the goal is to infer persona labels for users in $\mathcal{U}_I$.*
- *An **in-context example graph** $G_C = (\mathcal{U}_C, \mathcal{P}_C, \mathcal{E}_C, \mathbf{X}_C)$, where users are annotated with new persona labels $\mathbf{Y}_C \in \{0,1\}^{|\mathcal{U}_C| \times k'}$, $\mathbf{Y}_C \setminus \mathbf{Y}_T \neq \emptyset$.*

The objective is to infer persona labels for each user in $\mathcal{U}_I$ within the joint persona space of $\mathbf{Y}_T \cup \mathbf{Y}_C$, without requiring retraining of $\mathcal{M}$.

To illustrate the application of our proposed approach, consider an e-commerce platform that has curated a training graph G_T, where users are categorized into personas such as *Luxury Shopper*, *Tech Enthusiast*, and *Sustainable Consumer*. A model $\mathcal{M}$ is trained on this graph to infer the personas of unannotated users. Now, suppose the platform wants to introduce a new persona category, *Gaming Enthusiast*, to better capture users who frequently purchase gaming consoles, high-performance PCs, and accessories. Traditionally, updating the model would require retraining $\mathcal{M}$ from scratch, incorporating new data, and adjusting learned representations—a computationally expensive process.

Instead, we adopt an *in-context learning approach*: rather than retraining, we provide an *in-context example graph* G_C, which contains a small set of users explicitly labeled as Gaming Enthusiasts. These users serve as reference points for the model to recognize behavioral patterns associated with the new persona. The objective is for $\mathcal{M}$ to generalize from these examples and accurately infer personas for unseen users in the *inference graph* G_I, expanding its predictions to the joint persona space—including both previously known personas and the newly introduced Gaming Enthusiast category. For instance, if a new user in G_I frequently purchases gaming accessories alongside high-end smartphones, the model should infer that they likely belong to both the *Gaming Enthusiast* and *Tech Enthusiast* personas, even though the Gaming Enthusiast persona was not part of the original training set. This approach allows the platform to dynamically adapt to evolving user behaviors without costly retraining, ensuring that persona classification remains flexible and responsive to emerging trends.

2.3 Dataset Description

Flipkart is one of the world's largest e-commerce companies by sales volume. In this work, we utilize annotated data collected over a six-month period (August 2023 to January 2024) by tracking orders in the Fashion and Lifestyle category. This dataset provides a monthly snapshot of users, products, and order information. A summary of its structure is presented in Table 2. The dataset provides a rich and comprehensive view of user behavior in the Fashion and Lifestyle category over a six-month period, capturing both large-scale trends and seasonal dynamics. Spanning both summer and winter months, it enables the analysis of seasonal effects on shopping patterns. Notably, the dataset covers key shopping periods, including a major sale event in October and early November, reflected in a sharp increase in sales volume. It also encompasses the festive season, including Christmas and New Year's eve, where a surge in sales is observed in December, followed by a lull in January as shopping activity declines. These temporal variations make the dataset particularly valuable for understanding user personas in different contexts, revealing how shopping preferences shift in response to seasonal, festive, and promotional influences.

Dataset Statistics: To provide a more comprehensive view of the dataset and platform, we include additional statistics that illustrate user and persona distributions. The dataset comprises users categorized into nine persona types, with relatively balanced representation. The most common personas are Budget Shoppers (16.61%), Fashion Enthusiasts (16.02%), and Casual/Comfort Shoppers (16.02%), while Luxury Shoppers (10.16%) and Adventure Shoppers (10.74%) are the least represented. This diversity supports a wide range of product-persona associations. In terms of user demographics,

Persona Label	Definition	Characteristics
Fashion Enthusiast	Represents customers who are passionate about the latest fashion trends. They frequently explore new collections, follow fashion influencers, and prefer staying ahead of the style curve.	High interest in designer brands, fashion blogs, and social media trends.
Budget Shopper	Represents customers who are highly price-conscious, seeking the best deals and discounts. They are motivated by value for money and often purchase during sales or use coupons.	Visits sales sections, uses price comparison tools and is sensitive to promotions and deals.
Sport Shoppers	Represents customers who are focused on sportswear, activewear, and gear for physical activities such as fitness, yoga, running, and outdoor sports.	Interested in performance-focused clothing, accessories, and equipment for their active lifestyles.
Luxury Shoppers	Represents customers who prefer premium, high-end brands and exclusive products. They seek superior quality and are willing to spend more on luxury items.	Focused on exclusivity, brand reputation, and the highest quality materials.
Professional Attire Shoppers	Represents customers who are primarily interested in clothing and accessories for professional settings, such as office wear, formal attire, and business casual clothing.	Purchases are driven by the need for sophisticated, professional-looking outfits for work environments.
Casual/Comfort Shoppers	Represents customers who prioritize comfort and practicality in their clothing choices. They often seek casual, everyday wear that is easy to style and comfortable.	Focused on relaxed, functional clothing like loungewear, athleisure, or casual weekend outfits.
Adventure Shoppers	Represents customers who seek clothing and gear for outdoor adventures and activities such as hiking, camping, and travel.	Interested in rugged, versatile apparel and equipment designed for outdoor exploration.
Children Clothing Shoppers	Represents customers who frequently purchase children's clothing, including infant, toddler, and school-age categories. These shoppers prioritize comfort, durability, and style for kids.	Looks for easy-to-care, safe, and stylish options for children, often shopping based on age or specific events like school season or holidays.
Ethnic Wear Shoppers	Represents customers who shop for ethnic attire, either for themselves or as part of festive/wedding parties. This persona may focus on bridal dresses, bridesmaids' dresses, ethnic suits, and other ceremonial outfits.	Focused on high-quality, ethnic wear suited for festive occasions, often involving personalization and custom fittings.

Table 1: Persona Labels in E-Commerce Lifestyle Categories

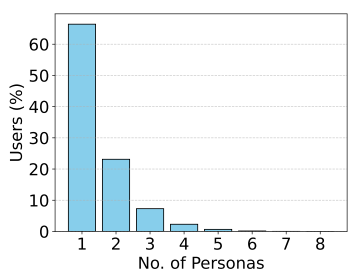


Figure 1: Distribution of personas across users in our data.

59.59% of users are male and 40.41% are female. Age distribution is grouped into four buckets, with the majority falling into bucket 2.0 (52.65%) and 1.0 (32.79%), suggesting that the platform primarily engages a younger to middle-aged demographic.

These statistics highlight the heterogeneity of users and personas in the dataset, reinforcing the need for a model that can effectively learn from diverse behavioral patterns.

Month	# User Nodes	# Product Nodes	# User-Product Edges	# Product-Persona Edges
August	10,000	62,723	78,479	10,068
September	16,757	59,230	77,737	11,973
October	29,097	103,929	161,299	19,873
November	36,003	93,216	147,109	44,295
December	57,438	126,815	223,253	27,234
January	17,286	46,919	65,322	8,492

Table 2: Summary statistics of the collected data. Product-persona edges are not provided to us as raw data. We create them using labeled information (refer to § 3.1).

Persona Classes: In collaboration with in-house business experts, the e-commerce platform has identified nine distinct personas within the Fashion and Lifestyle category. A detailed description of these personas is provided in Table 1. Each user in the dataset is assigned one or more persona labels based on their shopping behavior. The distribution of personas per user is illustrated in Figure 1. Manually annotating so many users over a six-month period is an immense challenge, yet it is essential for verifying the efficacy of the proposed technique. Previous efforts at this scale have not been feasible due to the substantial time and effort required to curate such a dataset.

3 TriPer: Proposed Methodology

TriPer maps the problem of persona identification to a link prediction task on a tripartite graph. As we will discuss next, the proposed data augmentation—transforming the data representation from a bipartite to a tripartite graph—plays a crucial role in enabling in-context learning to infer unseen personas without retraining.

3.1 Tripartite Graph Construction

We extend the persona-annotated bipartite graph into a tripartite structure by introducing a new node type—*persona* nodes—corresponding to each distinct persona in the labeled data $\mathbf{Y}$, denoted as $\mathcal{C} = \{c_i\}_{i=1}^{k}$. These persona nodes connect to both users and products, forming a richer representation of interactions (see Figure 2). The edges in this tripartite graph carry the following semantics:

User-persona edges: For each persona $c \in \mathcal{C}$, we identify the set of users associated with it:

$$\mathcal{U}^c = \{u \in \mathcal{U} \mid \mathbf{Y}[u][c] = 1\}. \quad (3)$$

An undirected edge is then added between user u and persona c if $u \in \mathcal{U}^c$, effectively linking users to their assigned persona labels.

Note that the user-persona edges are *not* available during inference. Hence, these edges are used as ground truth to only train our model.
Product-persona edges: A directed edge is created from a product $p \in \mathcal{P}$ to a persona $c_k \in \mathcal{C}$ if the product has been purchased at least θ times by users belonging to $\mathcal{U}^c$, i.e.,

$$|\{(u,p) \in \mathcal{E} \mid u \in \mathcal{U}^c\}| \geq \theta. \quad (4)$$

Here, θ is a hyper-parameter. This construction ensures that products strongly associated with a persona—based on user purchasing behavior—are explicitly captured within the graph structure.

3.2 Persona Identification as Link Prediction

With the integration of persona nodes into our graph, we reformulate the persona identification problem (see Prob. 1) as a link prediction task, where the goal is to determine whether an edge exists between a user u and a persona c. Note that in our proposed Gnn framework, user-persona edges are excluded from the message-passing process and are used solely as ground truth during training. This ensures that the model learns to infer persona associations based on user-product and product-persona relationships.

We adopt this formulation for several key reasons. Traditional node classification methods typically rely on a softmax activation over class affinity scores, which enforces a single-label assignment by forcing competition among classes. While a sigmoid activation can accommodate multi-label scenarios through independent predictions for each persona, as we will see in our empirical evaluation (§ 4), it falls short in effectively segregating users' diverse purchasing behaviors into distinct persona types.

In contrast, our link prediction framework explicitly models the relationship between users and personas by leveraging rich contextual information from the graph. Each persona node is embedded based on the products associated with it (through product-persona edges), and each user is embedded according to their purchase history. This common basis – products – anchors the embedding of both users and personas, and link prediction is performed as a function of the similarity between these embeddings.

Most importantly, this design enables in-context learning for personas not seen during training. *When in-context examples are provided, new personas are embedded using the model weights already learned, thereby ensuring generalizability to novel persona labels.* This is a significant advantage over traditional node classification frameworks, where the model parameter size is tightly coupled with the number of class labels. In our link prediction formulation, the model size remains independent of the number of users, products, or personas, as we demonstrate in the next section.

3.3 Tripartite Gnn for Link Prediction

Our goals are three-fold:

(1) **User embeddings:** Learn user representations $\mathbf{z}_u \in \mathbb{R}^{d_u}$ for each user $u \in \mathcal{U}$ that encapsulate their purchase behavior and are therefore predictive of their personas.
(2) **Persona embeddings:** Learn embeddings $\mathbf{z}_c \in \mathbb{R}^{d_c}$ corresponding to each personal node $c \in \mathcal{C}$ that characterize the types of products representative of each persona.

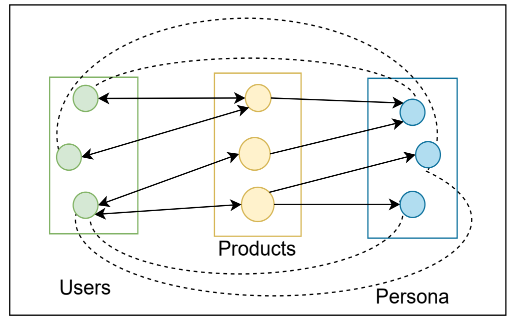


Figure 2: Tripartite graph schema to represent user-product-persona interactions. In this graph, only the solid edges are used for message passing in TriPer.

(3) **Persona identification as link prediction:** Use the learned node and persona embeddings to predict their association. Specifically, we formulate this as a link prediction task; *Iff* user u belongs to persona-type c, then the learned model should predict an edge between u and c, with high probability. Hence, we seek to minimize the following loss function.

$$\mathcal{J}_\Theta(G,\mathbf{Y}) = \sum_{u\in\mathcal{U}}\sum_{c\in\mathcal{C}} BCE\left(\sigma(\theta^T(\mathbf{z}_u \| \mathbf{z}_c)), \mathbf{Y}[u][c]\right) \quad (5)$$

Here, $\theta \in \mathbb{R}^{d_u+d_c}$ is a learnable weight vector, σ is sigmoid activation, $\|$ denotes the concatenation operation, and BCE indicates binary cross entropy loss.

To achieve our objective, we design a Tripartite Graph Neural Network, called TriPer, that performs a 3-phased message passing in *each* layer.Fig. 3 illustrates the pipeline.

- **Phase 1 - User to product:** During this phase, products aggregate messages from their purchasers (users). The goal is to develop product embeddings that reflect user purchasing patterns. As a result, products bought by similar users will have comparable embeddings, effectively capturing the relationships between products based on consumer behavior.
- **Phase 2 - Product to user:** In this phase, each user aggregates messages from the products they have purchased. The objective is to create user embeddings that capture shopping preferences and behaviors. Consequently, users who have bought similar products will have comparable embeddings, effectively representing the relationships between users based on their purchase history.
- **Phase 3 - Product to persona:** Finally, persona nodes collect messages from the products, constructing persona embeddings that capture the characteristic purchasing patterns of each persona. This step is a fundamental component of TriPer, ensuring that persona nodes integrate signals from all key products that strongly represent their respective personas.

After L layers of 3-phased message passing, the updated user and persona node embeddings are used for link prediction, and the gradients are backpropagated to update the model parameters Θ. During inference, a single forward pass is performed.

It is important to note that the user-persona edges are never used during message passing. This is a conscious choice since for unseen

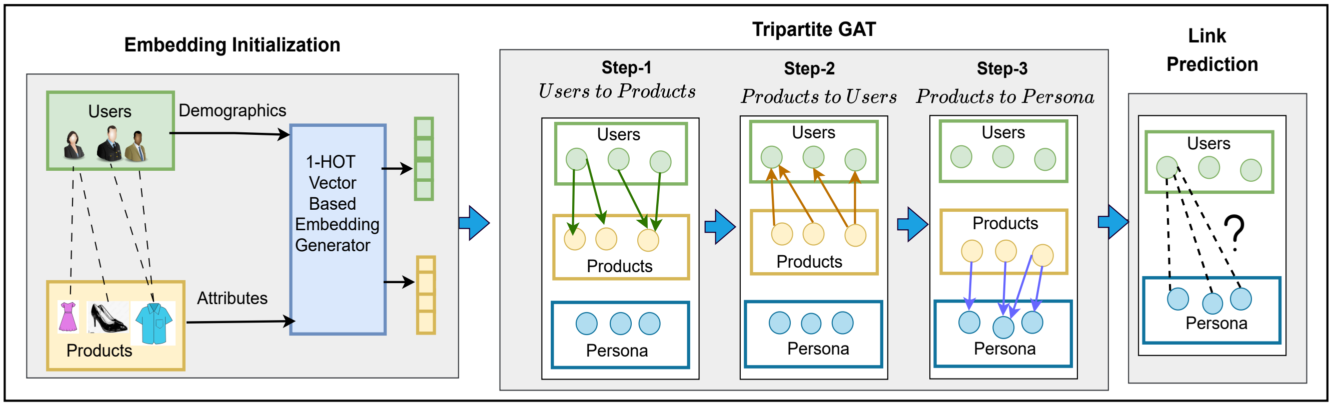


Figure 3: The figure illustrates the overall pipeline of TriPer.

users, their personas are not known. Hence, we aim to infer these edges based on user-product and product-persona edges, both of which are available for e-commerce vendors. Next, we detail each of the phases in further depth.

3.4 Initialization of Node Embeddings

For user and product nodes, we initialize them with their raw attributes defined in § 2.1. Specifically, $\mathbf{h}_p^0 = \mathbf{x}_p \| \mathbf{s}_p$ where $\|$ represents the concatenation operation. Similarly, $\mathbf{h}_u^0 = \mathbf{x}_u$.We initialize persona nodes using Xavier Initialization [14].

3.5 Tripartite Message Passing

• **Phase 1 - User to product**: Let p and u denote the product and the user nodes and $\mathbf{h}_p^\ell$ and $\mathbf{h}_u^\ell$ denote their embeddings, respectively, at layer ℓ. $\forall p \in \mathcal{P}$, we update the product embeddings using *attention-weighted* message passing as below as shown below:

$$\textbf{Receive messages: } \mathbf{m}_p^\ell(u) = \text{ReLU}\left(\mathbf{W}_\mathcal{P}^\ell \mathbf{h}_p^{\ell-1} \middle\| \mathbf{W}_\mathcal{P}^\ell \mathbf{h}_u^{\ell-1}\right)$$

$$\textbf{Compute attention: } \alpha_{pu} = \frac{\exp\left({\mathbf{a}_\mathcal{P}^\ell}^T \mathbf{m}_p^\ell(u)\right)}{\sum_{u' \in \mathcal{N}_p^{user}} \exp\left({\mathbf{a}_\mathcal{P}^\ell}^T \mathbf{m}_p^\ell(u')\right)},$$

$$\textbf{Update: } \mathbf{h}_p^\ell = \left(\mathbf{W}_\mathcal{P}^\ell \mathbf{h}_p^{l-1} + \sum_{u \in \mathcal{N}_p^{user}} \alpha_{pu} \mathbf{W}_\mathcal{P}^\ell \mathbf{h}_u^{\ell-1}\right)$$

Here, all weights matrices and vectors of the form $\mathbf{W}_\mathcal{P}^\ell \in \mathbb{R}^{d\times d}$ and $\mathbf{a}_\mathcal{P}^\ell \in \mathbb{R}^{2d}$ respectively are trainable parameters in the ℓ^{th} layer. $\|$ represents the *concatenation* operator and $\mathcal{N}_p^{user}$ represents the *user* neighbors of the product node p. α_{pu} refers to the attention coefficient between the product p and user u.

• **Phase 2 - Product to user**: In this phase, the *updated product embeddings* pass on the message to the user nodes. The updated product embeddings have information about all users who bought the product after the first level of message passing. The second phase of message passing from p to u ensures that all users' embeddings have information of all other users who bought similar products. This allows users buying similar sets/subsets of products (that belong to a specific persona) to have similar embeddings. Towards that objective, $\forall u \in \mathcal{U}$, we update the user node embeddings as shown below.

$$\textbf{Receive messages: } \mathbf{m}_u^\ell(p) = \text{ReLU}\left(\mathbf{W}_\mathcal{U}^\ell \mathbf{h}_u^{\ell-1} \middle\| \mathbf{W}_\mathcal{U}^\ell \mathbf{h}_p^{\ell-1}\right)$$

$$\textbf{Attention head: } \alpha_{up} = \frac{\exp\left({\mathbf{a}_\mathcal{U}^\ell}^T \mathbf{m}_u^\ell(p)\right)}{\sum_{p' \in \mathcal{N}_u^{product}} \exp\left({\mathbf{a}_\mathcal{U}^\ell}^T \mathbf{m}_u^\ell(p')\right)},$$

$$\textbf{Update: } \mathbf{h}_u^\ell = \left(\mathbf{W}_\mathcal{U}^\ell \mathbf{h}_u^{l-1} + \sum_{p \in \mathcal{N}_u^{product}} \alpha_{up} \mathbf{W}_\mathcal{U}^\ell \mathbf{h}_p^{\ell-1}\right)$$

All weights matrices and vectors of the form $\mathbf{W}_\mathcal{U}^\ell \in \mathbb{R}^{d\times d}$ and $\mathbf{a}_\mathcal{U}^\ell \in \mathbb{R}^{2d}$ respectively are trainable parameters in ℓ^{th} layer. $\mathcal{N}_u^{product}$ represents *product* neighbours of user node u. α_{up} refers to the attention coefficient between the user u and product p.

• **Phase 3 - Product to persona**: This phase ensures that: **(a)** persona node embeddings reflect the type of products it is frequently associated with and **(b)** the type of users buying those products. To capture these objectives, $\forall c \in \mathcal{C}$, we update the persona node embeddings as shown below.

$$\textbf{Receive messages: } \mathbf{m}_c^\ell(p) = \text{ReLU}\left(\mathbf{W}_\mathcal{C}^\ell \mathbf{h}_c^{\ell-1} \middle\| \mathbf{W}_\mathcal{C}^\ell \mathbf{h}_p^{\ell-1}\right)$$

$$\textbf{Attention head: } \alpha_{cp} = \frac{\exp\left({\mathbf{a}_\mathcal{C}^\ell}^T \mathbf{m}_c^\ell(p)\right)}{\sum_{p' \in \mathcal{N}_c^{product}} \exp\left({\mathbf{a}_\mathcal{C}^\ell}^T \mathbf{m}_c^\ell(p')\right)},$$

$$\textbf{Update: } \mathbf{h}_c^\ell = \left(\mathbf{W}_\mathcal{C}^\ell \mathbf{h}_c^{l-1} + \sum_{p \in \mathcal{N}_c^{product}} \alpha_{cp} \mathbf{W}_\mathcal{C}^\ell \mathbf{h}_p^{\ell-1}\right)$$

All weights matrices and vectors of the form $\mathbf{W}_\mathcal{C}^\ell \in \mathbb{R}^{d\times d}$ and $\mathbf{a}_\mathcal{C}^\ell \in \mathbb{R}^{2d}$ respectively are trainable parameters in ℓ^{th} layer. $\mathcal{N}_c^{product}$ represents *product* neighbours of persona node c. α_{cp} refers to the attention coefficient between the persona c and product p.

Prediction of user-persona links: Let $\mathbf{z}_u = \mathbf{h}_u^L$ and $\mathbf{z}_c = \mathbf{h}_c^L$ be the embeddings of user u and persona c in the final Gnn layer L.

As defined in the loss function (Eq. 5), the probability of user u belonging to persona c is computed as:

$$P(\langle u, c\rangle) = \sigma\left(\theta^T(\mathbf{z}_u \| \mathbf{z}_c)\right) \tag{6}$$

where $\theta \in \mathbb{R}^{d_u+d_c}$ is a learnable weight vector, and σ denotes the sigmoid activation function.

3.6 Inference and In-context Generalization

Inference: Given an inference graph $G_I = (\mathcal{U}_I, \mathcal{P}_I, \mathcal{E}_I, \mathbf{X})$, we first transform it into a tripartite graph as discussed in § 3.1. Since persona labels are unknown, all user-persona edges are initially absent, and our task is to predict them. To achieve this, we perform a forward pass on the tripartite graph, applying L layers of the 3-phase message passing mechanism to compute user and product embeddings, leveraging the weights learned during training. Finally, we apply Eq. 6 to obtain predictions for each user-persona pair $\langle u, c\rangle \in \mathcal{U} \times \mathcal{C}$.

Inference with in-context examples: Given an inference graph $G_I = (\mathcal{U}_I, \mathcal{P}_I, \mathcal{E}_I, \mathbf{X}_I)$ along with in-context example graph $G_C = (\mathcal{U}_C, \mathcal{P}_C, \mathcal{E}_C, \mathbf{X}_C)$, where users are annotated with new persona labels $\mathbf{Y}_C \in \{0, 1\}^{|\mathcal{U}_C| \times k'}$, we first construct the tripartite graph corresponding to G_I. In this graph, we augment the existing persona nodes with the newly introduced personas from $\mathbf{Y}_C$. Additionally, we establish product-persona edges for these new persona nodes by applying Eq. 4 on G_C. We then proceed with inference on this enhanced graph as outlined above.

This augmentation enables in-context reasoning by allowing the model to incorporate and leverage new persona information dynamically. Since the embeddings of users and personas are derived through message passing over product interactions, the newly added persona nodes immediately integrate into the learned representation space. The model can infer relationships between unseen personas and users based on shared product affinities, without requiring retraining. This facilitates adaptive reasoning, enabling the system to generalize to novel persona types solely from a small set of in-context examples.

Furthermore, as detailed in § 3.5, the number of parameters in TriPer remains independent of the number of users, products, or personas. Consequently, the addition of new personas does not hinder the ability of the model to perform inference.

4 Experimental Evaluation

In this section, we benchmark TriPer and establish:

- **Prediction accuracy:** TriPer outperforms a suite of baselines, including heterogeneous GNNs and LLM-powered recommendation engines in persona identification.
- **In-context generalization:** TriPer provides e-commerce vendors unprecedented flexibility in refining persona categorization strategies post-training through robust in-context learning.
- **Robustness:** TriPer demonstrates strong resilience in learning under low-data regimes.
- **Generalization over time:** TriPer maintains high predictive performance across temporal shifts, demonstrating its ability to capture meta-level behavioral patterns that remain relevant even as user preferences and market dynamics evolve.

Our codebase is available at https://github.com/idea-iitd/persona.git

Model	F1(↑)	Jaccard(↑)
MLNC	0.28±0.00	0.16±0.00
SlotGAT	0.41±0.00	0.28±0.00
SeHGNN	0.73±0.01	0.58±0.01
RpHGNN	0.44±0.01	0.34±0.01
LLM_Enhance	0.67±0.01	0.52±0.01
TAPE	0.74±0.00	0.60±0.01
EasyRec	0.48±0.00	0.32±0.00
BiPer	0.74±0.00	0.60±0.00
TriPer	**0.89±0.00**	**0.82±0.00**

Table 3: Performance of benchmarked algorithms.

4.1 Empirical Setup

We use a system running on Intel Xeon 6248 processor with 96 cores and 1 NVIDIA A100 GPU with 40GB memory for our experiment. For our training, we use hyperparameters mentioned in Table 4. The details of hyper-parameters are outlined in Table 4 in the Appendix.

4.1.1 Train-validation-test splits. For a specific month, we randomly select $7K$ users for the training set and $2K$ users for the validation set, along with their associated purchase data and annotated persona labels from the data provided for each month. The remaining users are used for testing. This methodology guarantees that the test set comprises entirely of users unseen during the training phase, allowing us to effectively assess the model's performance on new user profiles.

4.1.2 Metrics. We use F1-score and Jaccard similarity to quantify accuracy. The Jaccard similarity for a user is computed by measuring the overlap between the true and predicted persona sets, defined as: $J(y_{\text{true}}, y_{\text{pred}}) = \frac{|y_{\text{true}} \cap y_{\text{pred}}|}{|y_{\text{true}} \cup y_{\text{pred}}|}$ We report average Jaccard similarity across all users in the test set.

4.1.3 Baselines. We consider 7 state-of-the-art baselines spaning algorithms for multi-label node classification on heterogeneous graphs and LLM-empowered recommendation engines. These techniques include MLNC [34], SlotGAT [43], SeHGNN [40], RpHGNN [21], LLMs-Enhancers [5], TAPE [19] and EasyRec [31]. A detailed description of these baselines is provided in Appendix A.2.

In addition, we construct a simplified version of TriPer, called BiPer, which is designed primarily for ablation studies (Appendix A.3). Specifically, we only perform phase 1 and phase 2 of the message passing scheme. Then, we pass the user embeddings through an MLP followed by sigmoid activation.

4.2 Prediction Quality

First, we evaluate the performance of TriPer and the baselines only on the month of August 2023. This presents a relatively easier scenario since user purchase behavior is unlikely to shift significantly in the same month and thereby maintaining good alignment between the distribution of the train and test sets.

Table 3 presents the accuracy achieved by the various algorithms. TriPer outperforms all baselines by a significant margin. This demonstrates that the various design choices made in TriPer —including modeling the data as a tripartite graph, implementing a 3-phased message passing scheme, and transforming the *multi-label* *node classification problem into link prediction* — are highly effective on real-world e-commerce data.

We also note that BiPer emerges as the second-best performer, alongside TAPE, indicating that even the 2-phased message passing approach is good enough to outperform the majority of the baselines. However, the noticeable gap between BiPer and TriPer reveals that the third phase of message passing from products to personas, and the formulation of the problem as link prediction, significantly contribute to the accuracy.

Finally, it is particularly interesting that the most comparable baselines, TAPE and SeHGNN, despite being transformer-based architectures, fail to surpass the performance of TriPer. This observation suggests that having the correct inductive biases in the data representation—in the form of the tripartite graph architecture along with its three-phased message passing—is more important than training a more powerful neural architecture on the raw data.

We further demonstrate the performance of TriPer on another dataset - Amazon Review Dataset [20] in Appendix A.4.

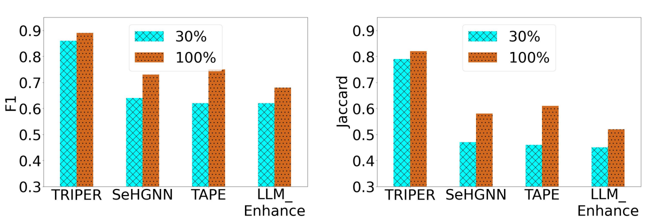
4.3 Training and Test Data Efficiency

Train data efficiency: *How much training data do we need to predict accurately?* Fig. 4 studies this question where we compare the performance of training on full dataset vs. training only on 30% of the edges chosen uniformly at random from the August month data. While we notice a reduction in accuracy with reduction in training data across all techniques, the reduction is substantially lower in TriPer, indicating efficacy of the inductive biases in TriPer enabling it learn from low volumes of data.

Test data efficiency: *What is the impact of limited purchase data on prediction accuracy?* To investigate, we simulated this scenario by retaining only 50% of the edges in the test set and evaluated prediction accuracy. The results, presented in Fig. 5, show an expected decline in accuracy. Notably, however, TriPer exhibits remarkable robustness, with a negligible drop in accuracy, whereas other baseline models suffer significant declines. This outcome highlights the benefits of TriPer's 3-phased message passing scheme and tripartite graph schema, which inject inductive biases that enhance performance even with limited data.

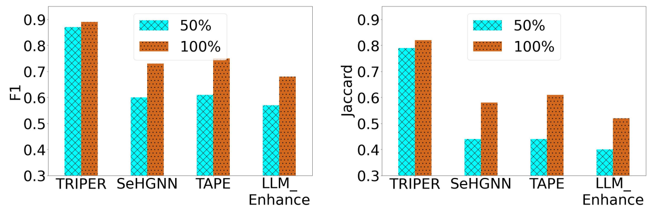
4.4 Generalization with In-Context Learning

The annotated data contains a total of 9 personas. To analyze the in-context learning efficacy of TriPer, we vary the number of personas included in the training set, as shown in Fig. 6, starting from 3. All of



(a) F1 **(b)** Jaccard

Figure 4: Compares models trained on the full dataset versus 30% of the train edges. TriPer exhibits a smaller accuracy drop, highlighting effective learning from reduced data.



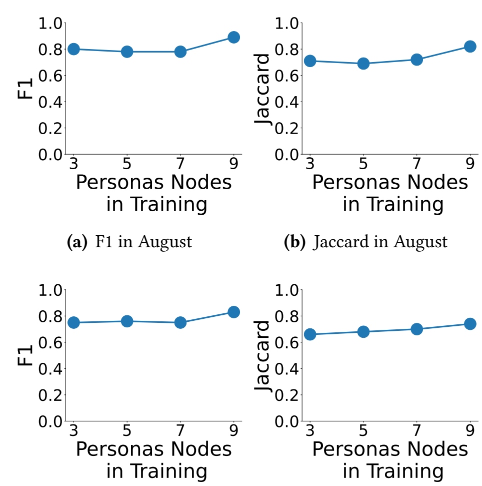
(a) F1 **(b)** Jaccard

Figure 5: Presents the results of reducing test data by retaining only 50% of the edges. Here, TriPer maintains a high level of accuracy, while significant drops are observed in baseline models.

the remaining personas are provided solely as in-context examples. We then study how the prediction accuracy improves as these in-context personas are progressively included in the training set.

As expected, performance improves with their inclusion; however, this improvement is less than 10% when shifting from 3 personas in the training set to all 9. This relatively small improvement highlights that TriPer effectively generalizes to unseen personas without requiring explicit training on them. The model's ability to leverage in-context examples suggests that its persona embeddings capture transferable knowledge, allowing it to recognize and associate new personas based on user-product interactions rather than requiring predefined label supervision.

It is important to note that none of the baselines offer in-context learning in their node-classification approach. In these baselines, the neural network's final layer (MLP) reduces user embeddings to a k-dimensional space, where k represents the number of personas



(a) F1 in August **(b)** Jaccard in August

(c) F1 in December **(d)** Jaccard in December

Figure 6: In-Context Generalization evaluated on the months of August and December, 2023. We examine the impact of reducing the number of personas in the training set and instead providing them as in-context examples on the accuracy of TriPer.

in the training set. This design inherently limits their ability to adapt to new personas.

4.5 Generalization over Time

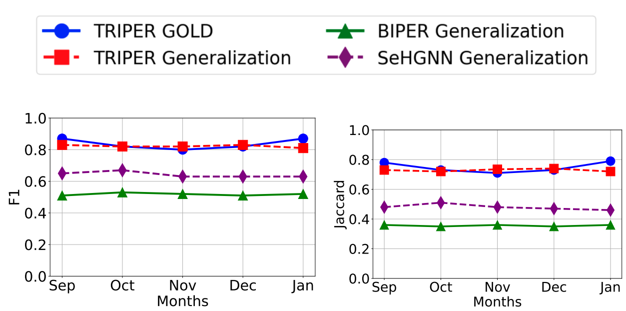
Can TriPer generalize effectively to data collected over subsequent months? Fig. 7 investigates this question by training TriPer on August data and evaluating its performance on the following five months, up to January. To isolate the model's long-term generalizability, we do not use any in-context examples in this analysis. Ensuring long-term generalizability is crucial, as annotating users with personas based on their shopping behavior and retraining the model with new data throughout the year is costly, labor-intensive, time-consuming, and often infeasible in real-world applications.

To assess the robustness of TriPer in generalizing over time, we compare its generalization performance against the GOLD standard for each subsequent month. In Fig. 7, GOLD refers to the accuracy obtained by training TriPer on a randomly selected subset of 7k users and their purchase behavior from the specific evaluation month, followed by testing on the remaining data from that same month. Ideally, perfect generalization would imply that the accuracy of the model trained on August data matches the accuracy of the model trained on data from the respective month under study.

Fig. 7 demonstrates that TriPer, despite being trained only on August data, generalizes well over the subsequent months. This is evident from the small gap between the generalization and the GOLD performance across all months. Notably, even amidst major Sale events (in October and November), seasonal shifts from summer to winter, and festive periods like Christmas, TriPer continues to exhibit strong generalization capabilities over time. Consistent with previous trends, the closest baselines, BiPer and SehGnn, continue to perform noticeable worse.

4.6 Ablation Studies

We perform ablation studies to observe the effect of **(a)** variation in θ, a parameter introduced in Section 3.1 **(b)** variation in number of GNN layers **(c)** Phase 3 of the message passing **(d)** Sensitivity of TriPer to noise in the embeddings. We present these in Section A.3 in the Appendix.



(b) F1 **(c)** Jaccard

Figure 7: Analyzes the generalization ability of the model trained on August over the next five months. TriPer GOLD refers to the performance of the model when trained on the same month as the test set.

5 Related Works

Heterogeneous Graph Neural Network (HGNN): HGNNs have been developed to effectively capture semantic information, driving significant progress in heterogeneous graph representation learning [9, 18, 27, 36, 39] with applications in social network analysis [26], recommendation systems [10, 37], and knowledge graph reasoning [2, 35]. HGNNs are typically categorized as Relation-based and Representation-based: Relation-based HGNNs [13, 41] aggregate node features along predefined metapaths to generate semantic-specific embeddings, which are fused into the final node representation; whereas, Representation-based HGNNs [28, 36, 44] build upon the principles of traditional GNNs by aggregating messages from a node's local neighborhood to capture both structural and semantic information. HGNNs distinguish themselves through the use of attention mechanisms, allowing them to effectively integrate the semantics of different node and edge types.

Persona identification in E-commerce: While there have been a long line of research focusing on issues with online commerce [1, 6, 7, 16, 17, 22, 23, 30, 32, 38], there has only been limited attempts on persona characterization in the e-commerce domain [8, 15, 25]. McIntosh et al. [29] utilized hard-coded rules to extract personas based on age in order to collect the views of people from different age groups about e-commerce applications. Whereas, our goal is to learn the personas of users based on their product interaction.

While substantial progress has been made in Heterogeneous Graph Neural Networks and, to some extent, for persona identification in e-commerce, prior works addressing both of these areas together is limited. In this paper, we address this gap.

6 Conclusion

Delivering personalized e-commerce experiences, ranging from landing pages and product recommendations to marketing campaigns and checkout processes, is critical for user retention and revenue generation. In this work, conducted in collaboration with one of the world's largest e-commerce companies, we tackled the problem of persona identification. This task poses unique challenges: users may embody multiple personas, persona definitions can evolve over time, and models should dynamically adapt without necessitating full retraining.

To address these challenges, we developed TriPer, a novel framework that integrates a 3-phased message-passing scheme within a Graph Neural Network and employs a tripartite graph schema to effectively model user-product-persona relationships. Additionally, TriPer leverages in-context learning to enhance adaptability without retraining. We rigorously evaluated TriPer on an extensive six-month e-commerce dataset, where each user was meticulously annotated by human experts. Our results demonstrate that TriPer not only significantly outperforms existing baselines but also exhibits strong generalization to subsequent scarce labeled data and maintains robustness even in low-data regimes. We hope that this work will spawn multiple future works on in-context graph learning not only in persona identification but also in several other application areas.

References

[1] Archit Bansal, Kunal Banerjee, and Abhijnan Chakraborty. 2022. These Deals Won't Last! Longevity, Uniformity and Bias in Product Badge Assignment in E-Commerce Platforms. *arXiv preprint arXiv:2204.12552* (2022).

[2] Trapit Bansal, Da-Cheng Juan, Sujith Ravi, and Andrew McCallum. 2019. A2N: Attending to neighbors for knowledge graph inference. In *Proceedings of the 57th annual meeting of the association for computational linguistics*. 4387–4392.

[3] Oren Barkan, Yonatan Fuchs, Avi Caciularu, and Noam Koenigstein. 2020. Explainable recommendations via attentive multi-persona collaborative filtering. In *Proceedings of the 14th ACM Conference on Recommender Systems*. 468–473.

[4] Ciro Bologna, Anna Chiara De Rosa, Alfonso De Vivo, Matteo Gaeta, Giuseppe Sansonetti, and Valeria Viserta. 2013. Personality-Based Recommendation in E-Commerce.. In *UMAP Workshops*.

[5] Zhikai Chen, Haitao Mao, Hang Li, Wei Jin, Hongzhi Wen, Xiaochi Wei, Shuaiqiang Wang, Dawei Yin, Wenqi Fan, Hui Liu, et al. 2024. Exploring the potential of large language models (llms) in learning on graphs. *ACM SIGKDD Explorations Newsletter* 25, 2 (2024), 42–61.

[6] Abhisek Dash, Abhijnan Chakraborty, Saptarshi Ghosh, Animesh Mukherjee, and Krishna P Gummadi. 2024. Investigating Nudges toward Related Sellers on E-commerce Marketplaces: A Case Study on Amazon. *Proceedings of the ACM on Human-Computer Interaction* 8, CSCW2 (2024), 1–31.

[7] Abhisek Dash, Saptarshi Ghosh, Animesh Mukherjee, Abhijnan Chakraborty, and Krishna P Gummadi. 2024. Sponsored is the New Organic: Implications of Sponsored Results on Quality of Search Results in the Amazon Marketplace. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, Vol. 7. 316–327.

[8] Lucky Dhakad, Mrinal Das, Chiranjib Bhattacharyya, Samik Datta, Mihir Kale, and Vivek Mehta. 2017. SOPER: Discovering the influence of fashion and the many faces of user from session logs using stick breaking process. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. 1609–1618.

[9] Yuxiao Dong, Nitesh V Chawla, and Ananthram Swami. 2017. metapath2vec: Scalable representation learning for heterogeneous networks. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*. 135–144.

[10] Wenqi Fan, Yao Ma, Qing Li, Yuan He, Eric Zhao, Jiliang Tang, and Dawei Yin. 2019. Graph neural networks for social recommendation. In *The world wide web conference*. 417–426.

[11] Ignacio Fernández-Tobías, Matthias Braunhofer, Mehdi Elahi, Francesco Ricci, and Iván Cantador. 2016. Alleviating the new user problem in collaborative filtering by exploiting personality information. *User Modeling and User-Adapted Interaction* 26 (2016), 221–255.

[12] Bruce Ferwerda, Mark P Graus, Andreu Vall, Marko Tkalcic, and Markus Schedl. 2016. The influence of users' personality traits on satisfaction and attractiveness of diversified recommendation lists. In *4th Workshop on Emotions and Personality in Personalized Syetems*. 43–47.

[13] Xinyu Fu, Jiani Zhang, Ziqiao Meng, and Irwin King. 2020. Magnn: Metapath aggregated graph neural network for heterogeneous graph embedding. In *Proceedings of the web conference 2020*. 2331–2341.

[14] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. 249–256.

[15] Ishika Goswami and Monomoy Goswami. 2021. Combined Use of Selected UX Research Techniques and Creation of User Persona for Design and Evaluation of Sustainable e-Commerce Apps—A Case Study. In *International Conference of the Indian Society of Ergonomics*. Springer, 1481–1493.

[16] Anjali Gupta, Shreyans J Nagori, Abhijnan Chakraborty, Rohit Vaish, Sayan Ranu, Prajit Prashant Nadkarni, Narendra Varma Dasararaju, and Muthusamy Chelliah. 2023. Towards fair allocation in social commerce platforms. In *Proceedings of the ACM Web Conference 2023*. 3744–3754.

[17] Anjali Gupta, Rahul Yadav, Ashish Nair, Abhijnan Chakraborty, Sayan Ranu, and Amitabha Bagchi. 2022. Fairfoody: Bringing in fairness in food delivery. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 11900–11907.

[18] Shubham Gupta, Sahil Manchanda, Sayan Ranu, and Srikanta J Bedathur. 2023. Grafenne: learning on graphs with heterogeneous and dynamic feature sets. In *International Conference on Machine Learning*. PMLR, 12165–12181.

[19] Xiaoxin He, Xavier Bresson, Thomas Laurent, Adam Perold, Yann LeCun, and Bryan Hooi. 2024. Harnessing Explanations: LLM-to-LM Interpreter for Enhanced Text-Attributed Graph Representation Learning. In *Twelfth International Conference on Learning Representations*.

[20] Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. 2024. Bridging Language and Items for Retrieval and Recommendation. *arXiv preprint arXiv:2403.03952* (2024).

[21] Jun Hu, Bryan Hooi, and Bingsheng He. 2024. Efficient heterogeneous graph learning via random projection. *IEEE Transactions on Knowledge and Data Engineering* (2024).

[22] Manas Joshi, Arshdeep Singh, Sayan Ranu, Amitabha Bagchi, Priyank Karia, and Puneet Kala. 2021. Batching and matching for food delivery in dynamic road networks. In *37th international conference on data engineering*. IEEE.

[23] Manas Joshi, Arshdeep Singh, Sayan Ranu, Amitabha Bagchi, Priyank Karia, and Puneet Kala. 2022. FoodMatch: Batching and Matching for Food Delivery in Dynamic Road Networks. *ACM Transactions on Spatial Algorithms and Systems* 8, 1 (2022), 1–25.

[24] Samira Khodabandehlou, S Alireza Hashemi Golpayegani, and Mahmoud Zivari Rahman. 2021. An effective recommender system based on personality traits, demographics and behavior of customers in time context. *Data Technologies and Applications* 55, 1 (2021), 149–174.

[25] Kang Li, Vinay Deolalikar, and Neeraj Pradhan. 2015. Mining lifestyle personas at scale in e-commerce. In *International Conference on Big Data*. IEEE.

[26] Yongkang Li, Zipei Fan, Jixiao Zhang, Dengheng Shi, Tianqi Xu, Du Yin, Jinliang Deng, and Xuan Song. 2022. Heterogeneous hypergraph neural network for friend recommendation with human mobility. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 4209–4213.

[27] Xingxing Liang, Yang Ma, Guangquan Cheng, Changjun Fan, Yuling Yang, and Zhong Liu. 2022. Meta-path-based heterogeneous graph neural networks in academic network. *Intl. Journal of Machine Learning and Cybernetics* (2022).

[28] Qingsong Lv, Ming Ding, Qiang Liu, Yuxiang Chen, Wenzheng Feng, Siming He, Chang Zhou, Jianguo Jiang, Yuxiao Dong, and Jie Tang. 2021. Are we really making much progress? revisiting, benchmarking and refining heterogeneous graph neural networks. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*. 1150–1160.

[29] Jennifer McIntosh, Xiaojiao Du, Zexian Wu, Giahuy Truong, Quang Ly, Richard How, Sriram Viswanathan, and Tanjila Kanij. 2021. Evaluating age bias in e-commerce. In *2021 IEEE/ACM 13th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE)*. IEEE, 31–40.

[30] Ashish Nair, Rahul Yadav, Anjali Gupta, Abhijnan Chakraborty, Sayan Ranu, and Amitabha Bagchi. 2022. Gigs with Guarantees: Achieving Fair Wage for Food Delivery Workers. In *Thirty-First International Joint Conference on Artificial Intelligence*. 5122–5128.

[31] Xubin Ren and Chao Huang. 2024. EasyRec: Simple yet Effective Language Models for Recommendation. *arXiv preprint arXiv:2408.08821* (2024).

[32] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. 2001. Item-Based Collaborative Filtering Recommendation Algorithms. In *10th International Conference on World Wide Web*. 285–295.

[33] Gkika Sofia, Skiada Marianna, Lekakos George, and Kourouthanasis Panos. 2016. Investigating the role of personality traits and influence strategies on the persuasive effect of personalized recommendations. In *4th Workshop on emotions and personality in personalized systems (EMPIRE)*, Vol. 9.

[34] Yingtong Sun, Ziyi Liu, Bryan Hooi, Yizhou Yang, Rizal Fathony, Jiyan Chen, and Bingsheng He. 2025. Multi-Label Node Classification with Label Influence Propagation. In *Thirteenth International Conference on Learning Representations*.

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

[36] 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*.

[37] Qitian Wu, Hengrui Zhang, Xiaofeng Gao, Peng He, Paul Weng, Han Gao, and Guihai Chen. 2019. Dual graph attention networks for deep latent representation of multifaceted social effects in recommender systems. In *The world wide web conference*. 2091–2102.

[38] Wen Wu and Li Chen. 2015. Implicit acquisition of user personality for augmenting movie recommendations. In *23rd International Conference on User Modeling, Adaptation and Personalization*. 302–314.

[39] Tianchi Yang, Linmei Hu, Chuan Shi, Houye Ji, Xiaoli Li, and Liqiang Nie. 2021. HGAT: Heterogeneous graph attention networks for semi-supervised short text classification. *ACM Transactions on Information Systems (TOIS)* 39, 3 (2021), 1–29.

[40] Xiaocheng Yang, Mingyu Yan, Shirui Pan, Xiaochun Ye, and Dongrui Fan. 2023. Simple and efficient heterogeneous graph neural network. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 37. 10816–10824.

[41] Chuxu Zhang, Dongjin Song, Chao Huang, Ananthram Swami, and Nitesh V Chawla. 2019. Heterogeneous graph neural network. In *25th ACM SIGKDD international conference on knowledge discovery & data mining*. 793–803.

[42] Yong Zheng. 2016. Adapt to Emotional Reactions in Context-aware Personalization.. In *EMPIRE@ RecSys*. 1–8.

[43] Ziang Zhou, Jieming Shi, Renchi Yang, Yuanhang Zou, and Qing Li. 2023. SlotGAT: slot-based message passing for heterogeneous graphs. In *International Conference on Machine Learning*. PMLR, 42644–42657.

[44] Shichao Zhu, Chuan Zhou, Shirui Pan, Xingquan Zhu, and Bin Wang. 2019. Relation structure-aware heterogeneous graph neural network. In *IEEE international conference on data mining*. IEEE, 1534–1539.

A Appendix

A.1 Empirical Setup

Hyperparameter	Value
Optimizer	Adam
Weight Decay	1×10^{-5}
Initial Learning Rate	1×10^{-3}
Learning Rate Scheduler	Cosine Annealing
No. of GNN layers	1
θ	2
LLM	Gemini 1.0 Pro

Table 4: Hyperparameters used for training TriPer.

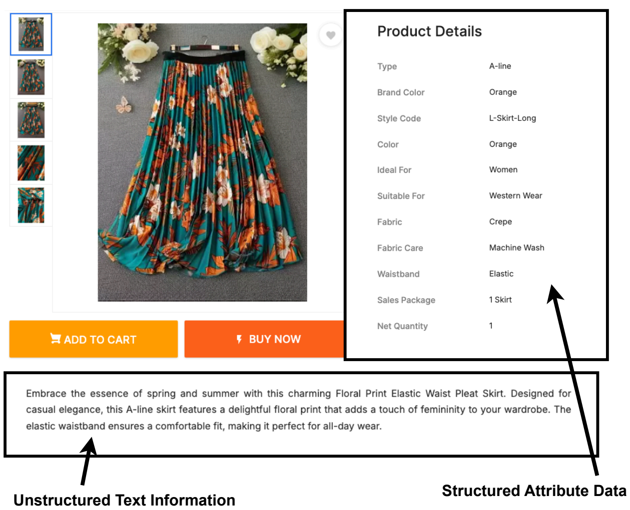


Figure 8: The figure illustrates the product description page.

A.2 Baselines

We consider the following baselines:

- **MLNC** [34] is a GNN-based approach designed for multi-label node classification. It models and propagates label dependencies through a label influence mechanism on graphs.
- **SlotGAT** [43] is a heterogeneous graph neural network (HGNN) that separately passes messages in slots, one for each node type, to maintain the representations in their own node-type feature spaces in order to avoid entangling the semantics in different node types into their representation.
- **SeHGNN** [40] is another type of HGNN that adopts the single-layer structure with long metapaths to extend the receptive field, as well as a transformer-based semantic fusion module to fuse features from different metapaths.
- **RpHGNN** [21] is a hybrid pre-computation-based HGNN that uses Random Projection Squashing and a Relation-wise Neighbor Collection with an Even-odd Propagation Scheme.
- **LLMs-Enhancers** [5] proposes an architecture that leverages LLMs to enhance nodes' text attributes with their massive knowledge and then generates predictions through GNNs. We adapt their framework for heterogeneous graphs and use HGNN instead of GNN.
- **TAPE** [19] is a representation learning framework for text-attributed graphs (TAGs) that leverage LLMs to capture textual information as features, which can be used to boost GNN performance on downstream tasks. We adapt their framework for heterogeneous graphs and use HGNN ([40]) instead of GNN.
- **EasyRec** [31] integrates LLMs to enhance recommendation tasks and combine collaborative language model tuning with the transformative capabilities of contrastive learning.
- **BiPer**: This is a simplified version of TriPer, designed primarily for ablation studies. Specifically, we only perform phase 1 and phase 2 of the message passing scheme. Then, we pass the user embeddings through an MLP followed by sigmoid activation.

A.3 Ablation Studies

A.3.1 Variation in Parameter θ: θ is a parameter introduced in Section 3.1. It helps in the creation of the product-persona edges. While a higher θ increases confidence in persona-product edges, it also sparsifies the graph. Persona classification performance initially improves but deteriorates beyond a point (Table 5).

Threshold (θ)	1	2	3	4	5	6
F1 (↑)	0.78	**0.89**	0.85	0.80	0.76	0.76

Table 5: Effect of increasing threshold θ on F1 score.

A.3.2 Variation in number of GNN layers (L): We show the effect of increasing the number of GNN layers in TriPer in Table 6. Performance improves up to $L = 3$, beyond which it degrades due to over-smoothing.

	$L = 1$	$L = 2$	$L = 3$	$L = 4$
F1 (↑)	0.89	0.90	**0.91**	0.78
Jaccard (↑)	0.82	0.83	**0.85**	0.72

Table 6: Performance metrics across different values of the number of GNN layers (L).

A.3.3 Effect of Phase 3 Message Passing: To understand the effect of the Phase 3 (product to persona) stage of message passing, we perform an experiment where we do not include the third-phase. We name this as BiPer. Comparison of BiPer againset TriPer is shown in Table 3.

A.3.4 Sensitivity to noise: Section 4.3 studies the impact of incomplete data during both training set and inference, mimicking cold-start scenarios. We further supplement the robustness evaluation of TriPer on noisy scenarios in Table 7. We add noise to user and product features, by sampling from a unit normal distribution within various values of standard deviation. We note some decrease in accuracy, which is expected since the inference set, where noise is added, goes out-of-distribution compared to train set. However, there is no scope of noise on interactions since these correspond to actual product purchases by users.

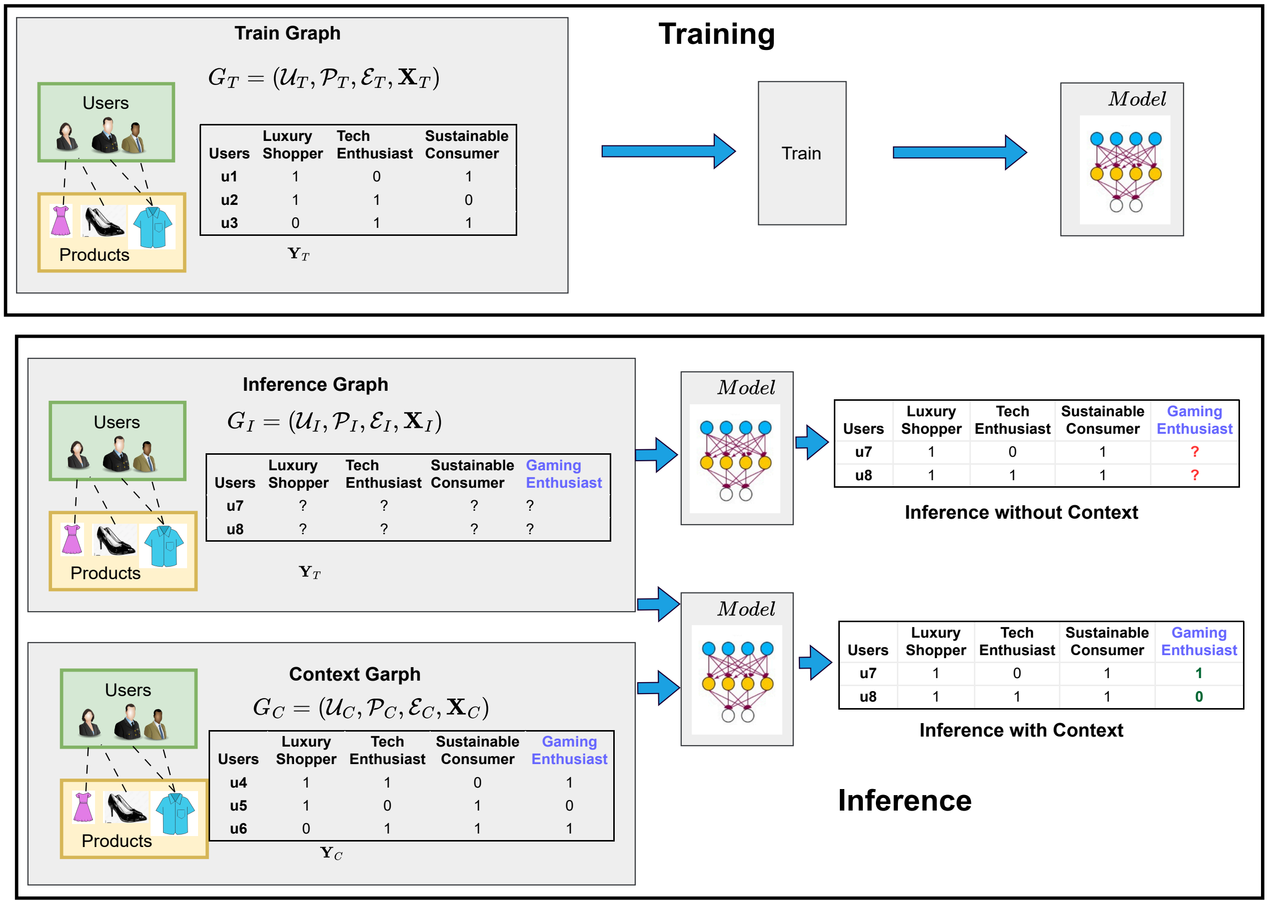


Figure 9: Illustration of In-Context Learning Formulation (Section 3.6, 4.4)

Level of Noise	$\sigma = 0$	$\sigma = 0.1$	$\sigma = 0.25$	$\sigma = 0.5$
F1	0.89	0.78	0.75	0.75
Jaccard	0.82	0.68	0.66	0.64

Table 7: Sensitivity of TriPer to noise.

	w/o Persona	with Persona
F1	0.64	**0.66**

Table 8: Recommendation performance comparison with and without persona information.

A.3.5 Evaluation on downstream recommendation task: We demonstrate the impact on product recommendation (modeled as a link prediction task on a user-product bipartite graph), where persona of a user is known vs. when its hidden. As we observe in Table 8, recommendation performance is better with persona information.

A.4 Evaluating TriPer on Amazon Review Dataset

To further validate the efficacy and generalizability of TriPer, we evaluate it on a newly curated e-commerce dataset based on Amazon product reviews [20]. The annotation process resulted in a labeled dataset comprising 1,470 users, who collectively reviewed 12,404 products, yielding 15,127 user-product edges in total.

We preprocessed the dataset to conform to our input format, where nodes represent users and products, and edges indicate a review interaction. Table 9 shows that TriPer attains the highest F1 score on the Amazon dataset, followed by SeHGNN and BiPer, demonstrating the robustness of TriPer in capturing nuanced persona-based patterns.

Metric	SeHGNN	BiPer	TriPer
F1 (↑)	0.83	0.44	**0.92**

Table 9: Comparison of F1 scores on the Amazon Review Dataset.