Graph Neural Network for Customer Engagement Prediction on Social Media Platforms

Tengteng Ma, Yuheng Hu, Yingda Lu, Siddhartha Bhattacharyya
University of Illinois at Chicago
{tma24, yuhenghu, yingdalu, sidb}@uic.edu

Abstract

Social media platforms such as Twitter and Facebook play a pivotal role in companies' strategy of engaging customers. How to target potential customers on social media effectively and efficiently is an important yet unsolved question. Predicting customer engagement on social media platforms is facing several challenges that cannot be solved by traditional methods. In this work, we design a framework that leverages individual behavior on Facebook together with network contextual information to predict customer engagement (like/comment/share) of a brand's posts. We first build a meta-path based Heterogeneous Information Network (HIN) to exploit large-scale content consumption information. We then design a Graph Neural Network (GNN) model combined with attention mechanism to learn structural feature representations of users to make the customer-brand engagement prediction. The proposed model is examined using a large-scale Facebook dataset and the result shows significant performance improvement compared with state-of-the-art baselines. Besides, the effectiveness of attention mechanism reveals the potential interpretability of the proposed model for the prediction results.

1. Introduction

Social media platforms, where users communicate with each other and generate content, play an important role in various online services. Brands are increasingly generating content on social media platforms to engage with their customers and boost new services and products. Users interact with brands by liking, commenting, or sharing brand posts, which reveal users' preferences, characteristics, and their attitude toward the brands. Various research has shown that engaging customers on social media has a significant impact on improving firms' sales, profitability and stock market performances [1, 2, 3], increasing customer visit frequency [4] and spreading brand knowledge [2]. Given such promises, firms often invest heavily in advertisements (e.g., Boost Your Post function on Facebook) to improve the exposure of their posts and grab customer attention on social media platform. However, the cost of such promotion services increases dramatically when the audience size increases. As such, it is important for firms to be able to accurately predict potential customer engagement, optimize their targeting and advertising strategies. This allows the firm to better target customers with a higher probability of engaging with brands, thus improve the efficiency of social media campaigns.

Predicting customer engagement faces several challenges. First, existing algorithms mainly rely on previous interaction information (i.e. which post focal user interacts with) to make predictions. For example, collaborative filtering is a classic method that provides recommendations based on the similarity between users or items (brand posts in our study). However, it only considers the binary interactions between users and items. Whereas information contained in the remaining network structure (i.e. which brand this post belongs to, which other posts this brand posts before, which users like other posts from this brand, etc.) is largely missing. We denote such information as contextual information. The second challenge is the large-scale online social network data. There are often millions of users who brands wish to engage, and these users often interact with millions of posts generated by hundreds of brands. However, traditional econometric-based prediction models such as ERGM model can only handle networks with hundreds of nodes. Thus to predict customer engagement in a social media platform, it is critical for the algorithm to be able to handle large-scale network efficiently. Third, traditional machine learning are facing challenge in predicting engagement as significant domain knowledge and efforts on feature engineering are needed. For deep learning algorithms, it can reduce demand on feature engineering at the cost of interpretability.

HłCSS

In this research, we propose a novel algorithm to predict engagement that addresses the challenges we mentioned above. We first define different types of entities in a social network that can help predict customer-brand engagement: customers, brand posts, brands, and brand categories. Based on this, we create a Heterogeneous Information Network (HIN) in which different types of entities are different types of nodes, and interaction among entities are edges. We further construct meta-paths [5] which capture the rich contextual information among various entities. Figure 1 shows a comprehensive view of the studied HIN. To tackle the large-scale network data, we apply Graph Neural Network (GNN) technique. One drawback of traditional HIN approaches is that no parameter sharing between nodes [6], and the growth of nodes leads to more parameters in the model, which increases computational complexity dramatically with large number of nodes. GNN based methods can share weights between nodes at each layer, which makes it possible to incorporate large-scale network data with high computational efficiency. To solve the challenge of interpretability, we leverage attention mechanism in the proposed model by adding an additional weighting matrix.

In this study, we build a model to predict customer engagement with brand posts by combining HIN with Graph Neural Network (GNN), that is, given a customer and a brand's post, whether the customer will either like/comment/share this post. Based on the HIN and meta-paths, we further incorporate GNN into our algorithm to learn the representation of nodes. We then use the learned representations of users and the pre-trained vectors of posts to perform the prediction task using deep neural networks. In addition, we leverage attention mechanism to provide interpretability of our deep learning framework.

The performance of the proposed model is evaluated on real-world data collected from Facebook. Extensive experiments have showed the outperformance of the proposed model compared with state-of-the-art baselines and verified the effectiveness of the proposed model.

To best of our knowledge, it is one of the first studies that develop a deep learning model using GNN from the perspective of HIN to predict user engagement on social media plat-forms. We conduct model evaluation on the Facebook dataset and the result shows dramatic performance improvement compared with baselines. The attention mechanism also provides interpretations of the recommendation results to some extent.
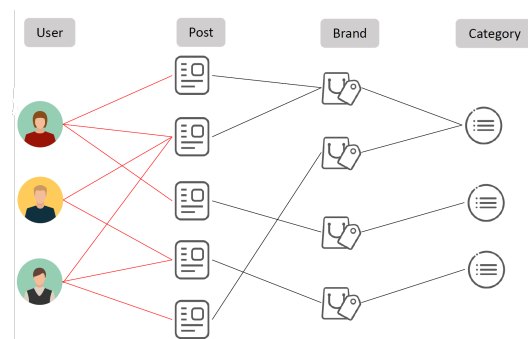

Figure 1. An example of heterogeneous customer engagement network on social media platforms. Different icons represent different types of entities.

2. Background

2.1. Online customer engagement

Online customer engagement has been studies in many academic disciplines. Online customer engagement involves customer interactions between the brand entity and other members in the online brand community [7]. [8] defines consumer-brand engagement as "a consumer's positively brand-related cognitive, emotional and behavioral activity during or related to focal consumer/brand interactions". Specifically, customer engagement on social media platforms mainly include three types: content reaction, content comment and content sharing [9]. For instance, Facebook users can like, comment or share brand posts, and Twitter has similar functions such as hearts, replies and retweets.

Previous literature on engagement mainly focus on exploring the causality of three types of mechanisms that influence user engagement: brand factors, peer customer factors and content factors. Customer engagement behavior can be influenced by brand identities, such as brand warmth, for-profit or non-profit status and communication strategy [9]. These factors depend on customers' perception of the brand. Besides, the content of post information also casts an impact on customer engagement behavior. [10] studied the main influential factors of users' sharing behavior on social media, and the result shows the content of a message, e.g., format and the sentiment, plays an important role in affecting users' engagement decision. Also, users have different social influence and bonding on social media platforms. Social ties are also related to engagement behavior [11] and has positive influence on recommendation [12].

2.2. Heterogeneous Information Network

Networks with different types of entities and links are called Heterogeneous Information Network (HIN). One popular technique to study HIN is network embedding, the objective of which is to learn a low-dimension representations of entities to reduce computational burden, meanwhile the learned representations preserve the structure of the network. Some studies focus on bipartite networks where there are only two types of nodes in the network [13, 14]. [5] built a meta-path based network embedding model for top-n recommendation. A model named metapath2vec is proposed in[15], in which heterogeneous random walks are formalized to leverage context information. [16] introduced a HIN based network embedding method for recommendation tasks. To deal with HIN structure, [17] categorized the edges in HIN into different types. [18] put forward a framework to preserve the meta-path based similarity in HIN. Unlike the existing methods, we also extract the post content features and incorporate the content information in the training model.

2.3. Graph Neural Network

Graph Neural Network (GNN), first proposed in [19], is an emerging technique which aims to leverage the deep neural networks to deal with graph-structured data. The key idea of GNN is to learn node embeddings based on local neighborhoods so that the embeddings of current nodes contain the information of its neighbors [6]. By aggregating the information and applying neural network, this mechanism is able to learn the representation of target nodes with all the information from their neighbors fused to it.

As a cutting-edge deep learning method, GNN is getting more attention from both industry and academia. GraphSAGE is proposed by [20] which concatenates self embedding and neighbor embedding. [21] proposed a GNN model to capture both node-level and semantic-level importance for heterogeneous information network. [22] introduced HetGNN, a model focusing on heterogeneous network. There are also some GNN applications in recommender system. [23] introduces a model named GraphRec for social recommendations. Graph Convolutional Neural Network is utilized for web-scale recommender systems in [24]. Different from these approaches, our proposed model employ attention mechanism to learn the representations of nodes efficiently and provide interpretations for the prediction results to some extent.

2.4. Interpretability using Attention Mechanism

Traditional machine learning interpretation is based on feature importance, which aims to map interpretable features to the output. Tree-based ensemble model is one of the examples. We can measure the contribution of each feature with various approaches, such as calculating the accuracy gain. Model-agnostic methods treat the models as black-boxes and try to explain predictions without accessing internal model structures and parameters [25]. [26] proposes a novel explanation technique, LIME, that explains the predictions of any classifier based on the assumption that there exists an interpretable model that can approximate the local predictions.

Instead of relying on the feature engineering or local approximation, attention mechanism provides understandable explanation for predictions by designing self-explanatory architectures. Attention mechanism involves the attention matrix that can be used to illustrate which parts of the input play more important roles in the training process. This technique is now being used to generate image descriptions [27], language translation [28], etc.

To sum, we have three methodological contributions. First, our model is one of the first studies that develop a deep learning model using GNN from the perspective of HIN to predict customer engagement on social media platforms. Besides, both network structure information and node-level information are leveraged in the model. Lastly, by applying attention mechanism, we provide interpretations of the prediction results to some extent to explore the underlying reason of customer behaviors.

3. The Proposed Model

3.1. Problem statement

The aim of the proposed method is to predict whether a target customer interacts (like/forward/reply) with a series of posts given historical interaction data of focal users. Specifically, let U, P denote a set of users and posts respectively, where a single user and a post can be represented by U_i and P_j. The objective is to predict the interaction between U_i and P_j, denoted by y_{ij} (0 or 1, 0 for no interaction between U_i and P_j, and 1 otherwise). Besides users and posts, there are some other types of entities, e.g., brands and categories, in the social media network, and are denoted as B and C respectively. Notice that there are also edges between P and B, as well as between B and C. To incorporate the complex contextual information in this HIN, we build meta-paths to represent the network in the social media

setting and leverage GNN and attention mechanism to learn a predict function for y_{ij}, i.e., whether the target user is interested in a specific post.

3.2. Meta-path based Heterogeneous Information Network

In our model, meta-paths are introduced to represent the heterogeneous information network structure and interactions between different types of entities. We define the HIN as $G = (V, E)$ where V, E represent nodes (entities) and edges (links) respectively, and a node type mapping function $\psi : V \rightarrow N$ and an edge mapping function $\phi : E \rightarrow \Gamma$ are associated with the HIN, where N and Γ represent the predefined node and edge types. Specifically, we study the social media network as a HIN where four types of nodes (users: U, posts: P, brands: B, categories: C) are included as well as various relationships among them. In this network, we define the schema of a meta-path schema as $\Phi : N_1 \xrightarrow{\Gamma_1} N_2 \xrightarrow{\Gamma_2} ... \xrightarrow{\Gamma_l} N_{l+1}$ where l indicates the lth step of a meta-path. Each meta-path schema includes different paths that can be denoted as $\phi : v_1 \rightarrow v_2 \rightarrow ...v_n \rightarrow ... \rightarrow v_m$, where $v_n \in V$. As we mentioned before, one of the challenges of tie prediction is the large number of users on social media platforms. To address this challenge, we use deep learning to learn the representation of each user, which can significantly reduce the dimension of the dataset. Thus we focus on the meta-paths starting at users. To be specific, we set the length of meta-path starting from user node to three, which means the number of nodes involved is fixed to four. Three meta-path schemas are defined to cover network structural information (Figure 2), they are: $User - Post - Brand - Category$ (this meta path can be denoted as $UPBC$), which means the engagement of the user to the post is related to the category that the brand belongs to; $User - Post - Brand - Post$ ($UPBP$), which indicates the user likes the post because of the brand; $User - Post - User - Post$ ($UPUP$), which shows the user interaction behavior is influenced by other users.
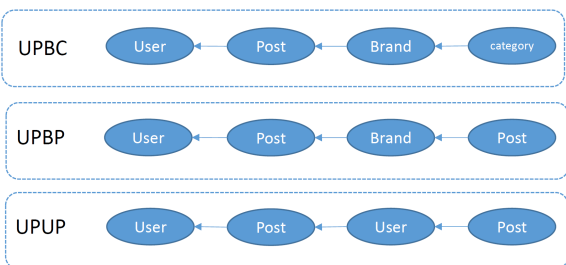

Figure 2. Meta-paths schema

Given a user node U_i and a meta-path Φ, the meta-path-based neighbors are the nodes included in the meta-path. Take Figure 3 as an example, if $N^i_{\Phi(u)}$ represents the ith neighbors of user U_i in a meta-path Φ, then $N^1_{UPUP(U_1)} = \{P_1, P_4\}$, $N^2_{UPUP(U_1)} = \{U_2, U_3\}$, $N^3_{UPUP(U_1)} = \{P_2, P_3\}$.
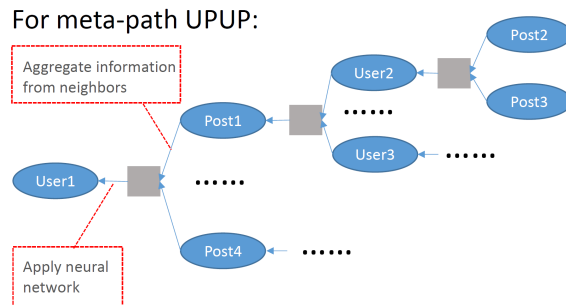

Figure 3. Meta-path example (UPUP)

In the proposed model, we exploit rich interaction information and learn representations of users from the HIN using GNN technique. The learned representations are applied to the prediction task. Figure 4 presents the framework of our model. First, we learn the representations of users for each meta-path respectively by aggregating all the information from neighbors, and weight the importance of representations from each meta-path using attention mechanism to get the optimized user representations. After that, the optimized user representations are concatenated with pre-trained post representations to predict whether users have interactions with posts. We further incorporate attention mechanism to provide interpretations of the prediction results.
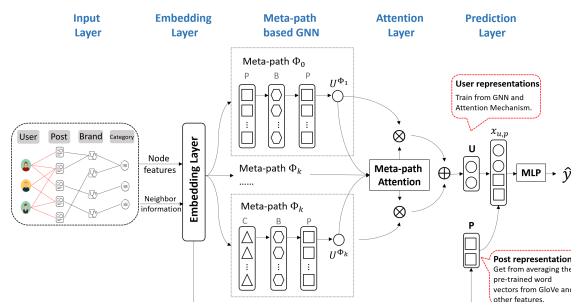

Figure 4. The framework of the model

3.3. Node embedding and vector initialization

The input to the model is node features and the HIN structure. For users, brands and categories, we

use the identity as the input feature. We set up an embedding layer to transport the initial one-hot vectors into low-dimensional vectors. Formally, given the one-hot vectors v_u, v_b and v_c that describe users, brands and categories respectively, the embedding vectors is formulated as:

$$h^u = O^T \cdot v_u$$
$$h^b = Q^T \cdot v_b$$
$$h^c = S^T \cdot v_c$$

Where $O \in R^{|U| \times d}$, $Q \in R^{|B| \times d}$ and $S \in R^{|C| \times d}$ denote the latent features of users, brands and categories, d is the dimension of the vectors. The obtained user, brand and category embedding vectors are used as latent features and will be updated during the training process.

The feature vectors of posts consist of two parts. First, we use the pre-trained word vectors from GloVe [29] to get the text features of posts. GloVe is a popular method for text mining which outperforms related models on word analogy, word similarity and named entity recognition tasks. Besides, textual data from social media platforms have some unique features, i.e., short length, unstructured phrased and abundant information, that are different from traditional linguistic data [30]. GloVe provides word vectors trained from social media platforms to represent elaborate textual information. We take the average of each word vector to represent textural features of each post. Then other post features such as post type are concatenated with text features to form the post representations:

$$h^p = ave(v_w, \forall w \in p) \oplus feature^p$$

where h^p represents post vectors, v_w means the vectors of word w that belongs to post p, and $feature^p$ is the other features of post p, e.g., content type, total number of like/reply/forward.

3.4. Meta-path based Graph Neural Network

Inspired by the idea of GraphSAGE [20], we develop a meta-path based GNN to learn the representations of users. We leverage meta-paths to obtain neighbors of users, and then aggregate the information of these neighbors to obtain the representations of users. Given a meta-path $\phi : v_1 \rightarrow v_2 \rightarrow ... v_n \rightarrow ... \rightarrow v_m$ where m indicates the step of nodes, the vectors of the current node v_n is:

$$h_{v_n} = \sigma(W_n \cdot h_{v_n}^0 \oplus g(\{h_{v_{n+1}}, \forall v_{n+1} \in N_p^1(v_n)\}_{n=1}^{m-1}) + b)$$

where σ is a non-linear function, W_n is the weight matrices, b is the bias vector, $h_{v_n}^0$ is the original

embedding vector of node v_n, $g(\cdot)$ is the aggregation function, and here we adopt max-pooling function based on the performance comparison between different aggregators.
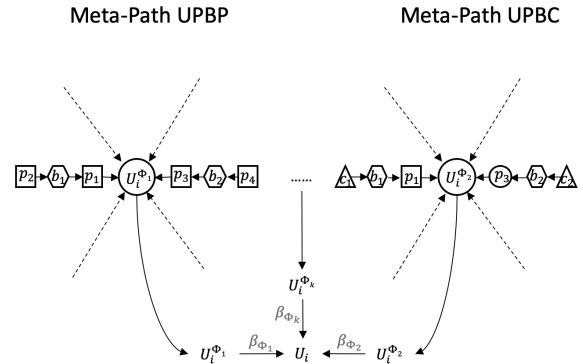

Figure 5. Example of aggregation process of GNN

Figure 5 illustrates a toy example of the GNN aggregation process. As shown in Figure 5, first we select a meta-path, saying meta-path $User - Post - Brand - Post$, then we can get $N_{UPBP}^1(u_i) = \{p_1, p_3\}$, $N_{UPBP}^2(u_i) = \{b_1, b_2\}$, $N_{UPBP}^3(u_i) = \{p_2, p_4\}$. The aggregation process starts from the nodes that are farthest from the central node u_i, i.e., $N_{UPBP}^3(u_i) = \{p_2, p_4\}$. The vectors of 2-nd step nodes $N_{UPBP}^2(u_i) = \{b_1, b_2\}$ are updated by aggregating the information of 3-rd step nodes ($N_{UPBP}^3(u_i) = \{p_2, p_4\}$) and a neural network is applied to it. The process is repeated along the path step by step until the representation of the central node u_i is obtained. Following the above process, the representations of same users from different meta-paths such as $UPBC$ and $UPUP$ are gained.

3.5. Attention layer

After getting the representations of users from different meta-paths, we have to solve another challenge: how to combine the representations from different meta-paths? Each meta-path can only reflect information from one single aspect. For example, meta-path $UPBC$ indicates whether the category of brands can influence users' interaction with posts, while meta-path $UPUP$ reveals the impact of other people's behavior on the users' engagement of brand posts. Just combining the user representations is not sufficient as the influence of different meta-paths may vary between each other.

To learn a more comprehensive representations of users, we implement attention mechanism [21] to figure out the importance of each meta-paths. That is, each meta-path is assigned a specific weighting which will be

updated during the process of training. By adding an attention layer, the model is able to learn the importance of different meta-paths automatically and fuse the information from each meta-path together effectively. Given a set of meta-paths $\{\Phi_1, \Phi_2, \cdots, \Phi_K\}$, the learned weights of each meta-path is:

$$(\beta_{\Phi_1}, \beta_{\Phi_1}, \cdots, \beta_{\Phi_K}) = attention(U_{\Phi_1}, U_{\Phi_2}, \cdots, U_{\Phi_K})$$

where $U_{\Phi_1}, U_{\Phi_2}, \cdots, U_{\Phi_K}$ are the representations of users learned from each meta-path, and $attention(\cdot)$ is the attention layer which performs the meta-path level attention.

To implement the attention mechanism, first we feed the representations learned from each meta-path to a one-layer non-linear neural network. Then an attention vector $\mathbf{q}$ is multiplied by node-level vectors and then we take the average to get the initial importance of each meta-path $\mathbf{w}_{\Phi_k}$:

$$\mathbf{w}_{\Phi_k} = \frac{1}{|U|} \sum_{i \in U} \mathbf{q}^T \cdot tanh(\mathbf{W} \cdot U_i^{\Phi_k} + \mathbf{b})$$

where $\mathbf{W}$ is the weight matrices and $\mathbf{b}$ is the bias vector. To keep the calculation consistent among meta-paths and ensure the weighting results are comparable, the parameters are shared between meta-paths.

Furthermore, the weights of meta-paths are normalized, and the final weights β_{Φ_k} represent the contribution of a specific meta-path Φ_k to the prediction task. After that, the final representations of users are obtained by combining the vectors from each meta-path with different weights.

$$\beta_{\Phi_k} = \frac{exp(\mathbf{w}_{\Phi_k})}{\sum_{k=1}^{K} exp(\mathbf{w}_{\Phi_k})}$$

$$U_i = \sum_{k=1}^{K} \beta_{\Phi_k} U_i^{\Phi_k}$$

where K is the total number of meta-paths and $U_i^{\Phi_k}$ is the representations learned from meta-path Φ_k.

3.6. Prediction and model training

The aim of our model is to predict the probability $\hat{y}_{i,j}$ of interaction between user u_i and post p_j which is in the range of [0,1]. Until now, we have the representations of all users U and all posts P, where U contains the features of users as well as the network

Table 1. Statistics of the dataset

Edges(A-B)	A	B	A-B
User-Post	32,535	1,730	1,247,318
Post-Brand	1730	7	1730
Brand-Category	7	1	7

structural information and P includes post features. To learn the probability of interactions between them, we concatenate the representations of users U and posts P together and feed the vectors into a Multi-Layer Perceptron component and map the prediction results between 0 and 1.

$$\hat{y}_{i,j} = sigmoid(MLP(U_i \oplus P_j))$$

For the MLP components, we apply three hidden layers and set ReLU as activation function.

In fact, the prediction task is also a binary classification task. We use the binary cross-entropy loss function to optimize the model:

$$Loss = \sum_{i,j \in Y} (y_{ij} log \hat{y}_{ij} + (1 - y_{ij}) log(1 - \hat{y}_{ij}))$$

where y_{ij} is the true labels, i.e., 0 or 1, and Y includes both positive and negative pairs.

4. Evaluation

4.1. Dataset

We collect the post information that was created between Jan.01, 2012 and Mar.19, 2016 from seven car brands on Facebook. For each post, we also have the list of users who liked/commented/shared the post. If a user u_i engages with a post p_j, then there is an edge between u_i and p_j. Users interacted with more than 1% posts are included. Table 1 shows the statistics of the Facebook dataset.

4.2. Baselines and evaluation metrics

To test whether our model if effective, we compared the performance with traditional recommender systems, deep learning based methods and graph embedding models. We selected state-of-art studies for each group and below is the detailed implementation.

Item-based Collaborative Filtering: It is a classic method of collaborative filtering that provides recommendations based on the similarity between items. We use the user-post adjacent matrix to obtain the cosine similarity between posts, and the predicted probability is calculated.

Matrix Factorization: It is a popular method that can capture the latent features of the nodes. The predicted probability is calculated by applying matrix factorization on user-post adjacent matrix.

MLP: We use Multilayer perceptron to build a deep neural network. A 3-layer MLP was built to perform the prediction task.

DeepWalk[31]: First, we train the embeddings of network nodes using DeepWalk, then feed the pre-trained embeddings to 3-layer MLP.

metapath2vec[15]: It is a state-of-art graph embedding model that aims to learn node representations for heterogeneous networks. This can be considered as HIN without GNN. We feed the pre-trained embeddings from metapath2vec to a 3-layer MLP.

We use Precision, Recall and F1 score as evaluation metrics. Precision expresses the proportion of the recommended posts that are indeed liked by the users, while recall reveals the model performance in finding all potential posts that are liked by the users, and F1 score is a blend of these two measurements. AUC provides an aggregate measure of performance across all possible classification thresholds.

4.3. Detailed implementation

The model is implemented on Tensorflow [32]. the dimension of the embedding layer is 64. Glorot normal initializer is used to initialize the parameters in the model. We choose Adaptive Moment Estimation (Adam) [33] as the optimizer because of its excellent performance compared with other optimizers. The batch size is 10000, the number of epoch is 5 and the learning rate is 0.0008. We split the whole dataset into the training dataset and test dataset with 80% and 20% respectively, and use the same dataset for the models to ensure the performances are comparable.

4.4. Performance evaluation

The performance of the proposed model and the baselines are reported in Table 2. As shown in the table, our proposed method outperforms the benchmark methods. Specifically, the precision is improved by 11.13%, and recall is 17.43% higher than metapath2vec. F1 score is improved from 0.8366 to 0.9237 (10.41%). Besides, the AUC is increased by 3.13%. This highlight the importance of incorporating GNN into HIN framework.

The improved performance indicates the proposed model achieves the best results by capturing the network structure and leveraging contextual information efficiently. The representations of users can be learned

Table 2. The performance of different methods of Facebook dataset

Method	Precision	Recall	F1	AUC
ItemCF	0.6941	0.0378	0.0716	0.5627
MF	0.5988	0.5001	0.5450	0.5849
MLP	0.6019	0.6458	0.6202	0.6258
DeepWalk+MLP	0.7473	0.7635	0.7553	0.8736
metapath2vec	0.8048*	0.8337*	0.8366*	0.926*
Our model	**0.8944**	**0.9790**	**0.9237**	**0.955**
Improvement	11.13%	17.43%	10.41%	3.13%

*The * means best performance among the baselines. The improvement is calculated on best performance baselines.*

effectively by fusing the structural information into the nodes. The combination of the learned user representations and the pre-trained post vectors provides impressive performance in the Graph Neural Network architecture.

4.5. Meta-path importance and interpretability

To find out whether the meta-path mechanism helps improve the prediction performance, we conduct several independent experiments by adding different meta-paths one by one to the model to observe the change of performance. Figure 6 shows the performance changing when a meta-path is added in the model. The prediction performance is improved by adding more meta-paths.
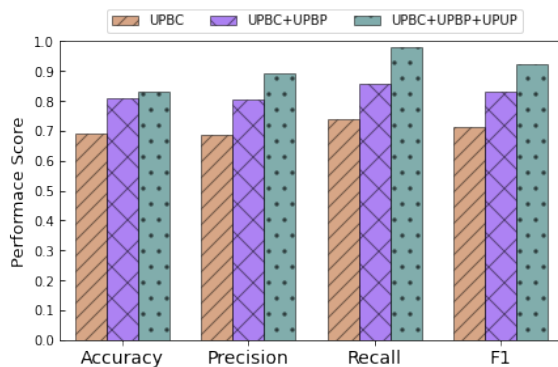

Figure 6. The performance change of addictive meta-paths

There are two reasons to adopt attention mechanism, one is to improve the efficiency of information aggregation, and the other one is to provide interpretations of the prediction result to some extent, which is also one of the major contributions of this study. By assigning weights among different meta-paths and update them during training process, attention mechanism can provide the importance of

each meta-path in the training process, in return, makes the prediction result interpretable. Since meta-paths contribution varies between each other, the learned meta-path weights can help us understand the underlying reason behind the engagement behavior.

After applying the user-level attention layer, we conduct one case to illustrate the interpretability of the proposed model. We randomly select two user nodes (U_{37} and U_{209}) that have a common interaction with one post P_{18}, and check the weights of each meta-path. As shown in Figure 7, the interaction between U_{37} and P_{18} is highly influenced by meta-path $UPBC$, while in the prediction between U_{209} and P_{18}, meta-path $UPBP$ plays the most import role. When we look into the dataset to try to find explanations of this, we notice that U_{37} has interactions with more than three car brands, while U_{209} only has engagement records with one brand. This gives evidence about why different meta-paths domain in different pair predictions.
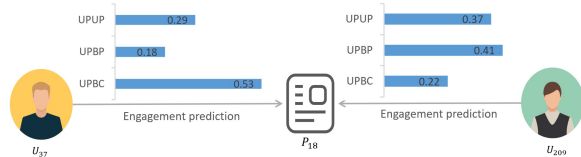

Figure 7. Meta-path importance on different pair prediction

4.6. The contribution of post text content

Leveraging both content and structural information is one of the novelties of the pro-posed model. To figure out how much did the post content contributes in the model training, we apply another model without the post content, i.e., using one-hot vector as post input (same with other types of nodes). The result is shown in Table 3.

Table 3. Performance comparison between with/without post content

Method	Precision	Recall	F1	AUC
Our model (without post content)	0.8237	0.9361	0.8763	0.9340
Our model (with post content)	0.8944	0.9790	0.9237	0.955
Improvement	8.58%	4.58%	5.41%	2.25%

As shown in Table 3, although only applying structural information can still achieve a good performance, the model with both content and structural information involved gives the best performance.

5. Conclusion

In this work, we learn representations of users that encoding both structural features and self-features from meta-path based context and propose a deep learning framework that characterizes the interaction between users and posts to perform personalized prediction. We propose efficient representations that can reserve the network structure well and are suitable for complicated deep learning models. We further combine GNN with HIN to increase the computational efficiency of our model. Lastly, we use the attention mechanism to aggregate information from different meta-paths effectively and provide interpretations of the prediction results to some extent. The proposed model is evaluated with Facebook data and the performance is improved significantly compared with baselines. We validate the efficiency and efficacy of the proposed model, and the mechanism can be generalized to various social media platforms. In the future, we would like to include more existing methods as benchmarks and extend the model to incorporate the dynamic of evolving heterogeneous social networks.

References

[1] H. Chen, P. De, and Y. J. Hu, "It-enabled broadcasting in social media: An empirical study of artists' activities and music sales," *Information Systems Research*, vol. 26, no. 3, pp. 513–531, 2015.

[2] V. Kumar and R. Mirchandani, "Increasing the roi of social media marketing," *MIT sloan management review*, vol. 54, no. 1, p. 55, 2012.

[3] Y. Yu, W. Duan, and Q. Cao, "The impact of social and conventional media on firm equity value: A sentiment analysis approach," *Decision Support Systems*, vol. 55, no. 4, pp. 919–926, 2013.

[4] R. Rishika, A. Kumar, R. Janakiraman, and R. Bezawada, "The effect of customers' social media participation on customer visit frequency and profitability: an empirical investigation," *Information systems research*, vol. 24, no. 1, pp. 108–127, 2013.

[5] B. Hu, C. Shi, W. X. Zhao, and P. S. Yu, "Leveraging meta-path based context for top-n recommendation with a neural co-attention model," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1531–1540, ACM, 2018.

[6] J. Zhou, G. Cui, Z. Zhang, C. Yang, Z. Liu, and M. Sun, "Graph neural networks: A review of methods and applications," *arXiv preprint arXiv:1812.08434*, 2018.

[7] R. J. Brodie, A. Ilic, B. Juric, and L. Hollebeek, "Consumer engagement in a virtual brand community: An exploratory analysis," *Journal of business research*, vol. 66, no. 1, pp. 105–114, 2013.

[8] L. D. Hollebeek, M. S. Glynn, and R. J. Brodie, "Consumer brand engagement in social media:

Conceptualization, scale development and validation," *Journal of interactive marketing*, vol. 28, no. 2, pp. 149–165, 2014.

[9] V. Barger, J. W. Peltier, and D. E. Schultz, "Social media and consumer engagement: a review and research agenda," *Journal of Research in Interactive Marketing*, 2016.

[10] J. Huang, S. Su, L. Zhou, and X. Liu, "Attitude toward the viral ad: Expanding traditional advertising models to interactive advertising," *Journal of Interactive Marketing*, vol. 27, no. 1, pp. 36–46, 2013.

[11] A. Stibe, H. Oinas-Kukkonen, and T. Lehto, "Exploring social influence on customer engagement: a pilot study on the effects of social learning, social comparison, and normative influence," in *2013 46th Hawaii International Conference on System Sciences*, pp. 2735–2744, IEEE, 2013.

[12] O. Oechslein and T. Hess, "The value of a recommendation: The role of social ties in social recommender systems," in *2014 47th Hawaii International Conference on System Sciences*, pp. 1864–1873, IEEE, 2014.

[13] L. Xu, X. Wei, J. Cao, and P. S. Yu, "Embedding of embedding (eoe): Joint embedding for coupled heterogeneous networks," in *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pp. 741–749, ACM, 2017.

[14] M. Gao, L. Chen, X. He, and A. Zhou, "Bine: Bipartite network embedding," in *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pp. 715–724, ACM, 2018.

[15] Y. Dong, N. V. Chawla, and A. Swami, "metapath2vec: Scalable representation learning for heterogeneous networks," in *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 135–144, ACM, 2017.

[16] C. Shi, B. Hu, W. X. Zhao, and S. Y. Philip, "Heterogeneous information network embedding for recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 2, pp. 357–370, 2018.

[17] S. Chang, W. Han, J. Tang, G.-J. Qi, C. C. Aggarwal, and T. S. Huang, "Heterogeneous network embedding via deep architectures," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 119–128, ACM, 2015.

[18] Z. Huang and N. Mamoulis, "Heterogeneous information network embedding for meta path based proximity," *arXiv preprint arXiv:1701.05291*, 2017.

[19] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61–80, 2008.

[20] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Advances in Neural Information Processing Systems*, pp. 1024–1034, 2017.

[21] X. Wang, H. Ji, C. Shi, B. Wang, Y. Ye, P. Cui, and P. S. Yu, "Heterogeneous graph attention network," in *The World Wide Web Conference*, pp. 2022–2032, ACM, 2019.

[22] C. Zhang, D. Song, C. Huang, A. Swami, and N. V. Chawla, "Heterogeneous graph neural network," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 793–803, ACM, 2019.

[23] W. Fan, Y. Ma, Q. Li, Y. He, E. Zhao, J. Tang, and D. Yin, "Graph neural networks for social recommendation," in *The World Wide Web Conference*, pp. 417–426, ACM, 2019.

[24] R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, and J. Leskovec, "Graph convolutional neural networks for web-scale recommender systems," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 974–983, ACM, 2018.

[25] M. Du, N. Liu, and X. Hu, "Techniques for interpretable machine learning," *arXiv preprint arXiv:1808.00033*, 2018.

[26] M. T. Ribeiro, S. Singh, and C. Guestrin, "" why should i trust you?" explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

[27] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International conference on machine learning*, pp. 2048–2057, 2015.

[28] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[29] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.

[30] X. Hu and H. Liu, "Text analytics in social media," in *Mining text data*, pp. 385–414, Springer, 2012.

[31] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 701–710, ACM, 2014.

[32] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, *et al.*, "Tensorflow: A system for large-scale machine learning," in *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pp. 265–283, 2016.

[33] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.