Implicit Session Contexts for Next-Item Recommendations

Sejoon Oh
soh337@gatech.edu
Georgia Institute of Technology
United States

Ankur Bhardwaj
ankurbhardwaj843@gmail.com
Georgia Institute of Technology
United States

Jongseok Han
jhan405@gatech.edu
Georgia Institute of Technology
United States

Sungchul Kim
sukim@adobe.com
Adobe Research
United States

Ryan A. Rossi
ryrossi@adobe.com
Adobe Research
United States

Srijan Kumar
srijan@gatech.edu
Georgia Institute of Technology
United States

ABSTRACT
Session-based recommender systems capture the short-term interest of a user within a session. Session contexts (i.e., a user’s high-level interests or intents within a session) are not explicitly given in most datasets, and implicitly inferring session context as an aggregation of item-level attributes is crude. In this paper, we propose ISCON, which implicitly contextualizes sessions. ISCON first generates implicit contexts for sessions by creating a session-item graph, learning graph embeddings, and clustering to assign sessions to contexts. ISCON then trains a session context predictor and uses the predicted contexts’ embeddings to enhance the next-item prediction accuracy. Experiments on four datasets show that ISCON has superior next-item prediction accuracy than state-of-the-art models. A case study of ISCON on the Reddit dataset confirms that assigned session contexts are unique and meaningful.

CCS CONCEPTS
• Information systems → Recommender systems.

KEYWORDS
Session-based Recommendation, Session Contextualization

1 INTRODUCTION
Session-based recommendation systems (SBRSs) [10, 11, 19, 20, 23, 24, 33, 37, 38, 41, 42] have been proposed to accurately model a user’s short-term and evolving interest, where a user’s session is defined as a sequence of its interactions with items occurring within a short time period [11, 34].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

CIKM '22, October 17–21, 2022, Atlanta, GA, USA
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-9236-5/22/10...
https://doi.org/10.1145/3511808.3557613

Context-aware recommendations [2, 7, 10, 23, 26, 28, 32, 33, 36] have also gained attention as contexts can represent high-quality knowledge about the user’s interests, which can enhance next-item prediction performance. In this paper, we define the contexts of a session as a user’s high-level interests and objectives during the session. Our goal in the paper is to predict session context, as it can serve as novel features to enhance the next-item prediction quality in the current session. Furthermore, since consecutive sessions are likely to have related user interests (i.e., complementary, similar, or supplementary) [19], the current session context can help make better recommendations in the next session, especially when the next session only has a few interactions.

Existing context-aware models have limitations in SBRSs that (1) they require session context information to be given explicitly, but in reality, the session contexts are often unavailable in the data or may need to be implicitly inferred, (2) they cannot incorporate session contexts to their models, (3) they employ inaccurate implicit session contexts when only the first few items of a session are observed, and (4) simple aggregation of item features to derive implicit session contexts can be inaccurate. Since session contexts are rarely presented explicitly, it is crucial to create methods that can assign precise and meaningful implicit contexts to sessions.

We propose a novel recommendation model called ISCON (Implicit Session CONtexts for next-item recommendations). ISCON first finds implicit contexts of sessions via graph-based session contextualization, which is more meaningful and precise compared to existing session contextualizations using simple item feature aggregation. ISCON trains a context predictor using the implicit contexts as labels to predict future sessions’ contexts accurately. Finally, ISCON utilizes the predicted contexts as novel features to enhance the next-item prediction accuracy. The main novelties of ISCON include that (1) ISCON develops an implicit session context predictor that estimates session contexts (even for sessions with few items and for future sessions), (2) a next-item predictor that leverages predicted session contexts and merges them with other features. Experimentally, ISCON outperforms 4 state-of-the-art SBRSs across 4 real-world datasets. A case study of ISCON on the Reddit dataset shows that the sessions are properly contextualized. Our dataset and code used in the paper are available here.

2 RELATED WORK
Context-aware recommender systems [2, 7, 10, 23, 26, 32, 33, 36, 39, 43] incorporate contextual information into their models for

1https://github.com/srijankr/iscon
capturing user preferences correctly [16]. A session context can imply various aspects such as temporal features [2, 21, 32] and graph-based features [26, 32, 39]; our context definition is high-level intents or interests of a user in a session, which is related to multi-interest extraction methods [3, 18, 27]. Many algorithms assume the context information is given [2, 7, 10, 14, 34] or perform user- or interaction-level contextualization [3, 18, 27, 39, 43], not session-level. Few methods [23, 33, 36, 39] are able to contextualize sessions, but their contextualizations can be inaccurate for sessions with few items or sessions that are not included in the training, since they do not utilize other session information or cannot inductively contextualize sessions without model retraining (e.g., In Table 2, ISCON outperforms CSRM that also contextualizes sessions).

For SBRs, attention mechanisms have been widely adopted [19, 20, 23, 24, 33, 39]. Graph neural network-based models [36–39, 42] also achieve superior performance by capturing complex transitions of items on a session-item graph. These models cannot assign implicit contexts to sessions accurately, predict session contexts, or use the session contexts for the next-item prediction.

3 PROPOSED APPROACH: ISCON

3.1 Contextualizing Sessions

We define the context of a session as a summary of interests expressed by a user’s interactions in a session. A session’s contexts can be used as prior knowledge to predict items that the user is likely to be interested in the session, and this can enhance the next-item prediction accuracy. Most public datasets do not include explicit contexts (i.e., interests specifically stated by a user) of a session as it is hard to gather; for example, it is intrusive and disruptive to ask users about their current session contexts/interests directly. Unlike explicit contexts, implicit contexts of a session can be inferred interests from the users’ interactions. Trivial approaches such as aggregating item features in a session cannot find proper implicit contexts when only the first few items of the session are observed or item features are uniformly distributed, which shows the need for a more sophisticated session contextualization method.

Our session contextualization approach consists of two steps: 1) generating session embeddings from a user-item multigraph, and 2) clustering sessions to identify session context clusters. To obtain session embeddings, we create a session-item bipartite multigraph, where its nodes are sessions and items, and its edges (undirected) represent membership between an item and a session. We apply a node embedding method called GraphSage [9] to the bipartite multigraph to obtain session embeddings. GraphSage [9] is used as its inductive capability that does not require retraining when computing the embeddings of future or test sessions. The key advantages of this graph-based technique include (1) it does not require pre-trained item embeddings, and (2) by considering multi-hop paths in the graph, this method can produce generalizable session embeddings that encode not only the items in the current session but also those in nearby sessions in the graph. Next, given the number of implicit session contexts |C| (hyperparameter; found by empirical searches), we cluster sessions by applying the K-means clustering [15] (K = |C|) to session embeddings. Each cluster represents similar sessions with the same implicit context. We define an implicit context of each session as the number of a cluster closest to the session embedding. We define trainable embeddings of all |C| implicit session contexts as session context embeddings.

3.2 Session Context Prediction

Predicting the context of a future session and updating the context of the session in real-time (as new items are observed) can help guide the next-item prediction task. However, using the context assignment method in Section 3.1 is not scalable, as multi-hop aggregation in the multigraph after observing every new item to re-generate session embeddings in real-time is expensive. Moreover, this method does not consider the relationships across consecutive sessions of a user. Naturally, the contexts of consecutive sessions of a user can be related (complementary, similar, or supplementary) [19]. The previous sessions’ context information can thus be used to predict the current/future session’s context better.

We train a novel real-time session context predictor using the user’s short-term (current session) and long-term (previous sessions) interest vectors as well as a trainable user embedding (see Figure 1). The predicted session context is updated dynamically whenever we observe a new item in a session. We train two Bi-directional LSTMs (Bi-LSTMs) [25] to get both vectors. Bi-LSTMs have shown superior performance in recommendation than LSTMs by utilizing both direction sequences [6, 45]. We have tried other architectures such as Transformer [30] or GRU [5] for ISCON, but they have empirically shown similar or worse prediction performance compared to the performance of the Bi-LSTM.

To derive the user’s long-term interest vector using a long-term (session-level) Bi-LSTM, we feed to it a sequence of previous L sessions’ features including the session embedding from Section 3.1 and metadata. The metadata of a session s includes the session duration $D_s$ (in seconds), the time interval $\Delta_s$ between sessions s and s + 1, and the number of items $M_s$ in the session. Given the current session s of a user u, the output $z_{long}$ from the long-term Bi-LSTM $\Theta_{long}$ is given as follows:

$$z_{long} = \Theta_{long}(F_{s-L}, \ldots, F_{s-1})$$

$$F_s = \text{concat}(E_i^s, D_s, \Delta_s, M_s)$$

where $L$ is the maximum sequence length (a hyperparameter), and $E_i^s$ is the session embedding of the session s from Section 3.1.

To obtain the short-term interest representation of a user using a short-term (item-level) Bi-LSTM $\Theta_{short}$, we input the sequence of embeddings of the observed items in the session. The short-term vector is updated whenever we observe a new item in the session to make our context prediction more accurate. Given the session s of a user u and observed items $i_1, \ldots, i_k$ in the session so far, the output
We note that contexts in Sections 3.1 and 3.2 are the same, and we use $y$ with the following Cross-Entropy loss and Adam optimizer on all contexts in Section 3.1 since the trainable ones show higher next-item prediction accuracy empirically.

Finally, we use a learnable user embedding $E_u$ as one of the input features for the next-item predictor. We expect improved personalization with $E_u$ since it is trained only with interactions of that particular user and is a representation of a user’s overall behavior.

We concatenate the three vectors – long-term interest, short-term interest, and a user embedding – and feed them to fully connected and Softmax layers. The output is the session context predictor and next-item predictor, since they are optimized to predict session contexts, not to predict next items.

Third, similar to the context predictor, we use a user embedding $E_u$ as one of the input features for the next-item predictor. A user embedding can personalize the predictions. Furthermore, a user embedding can ease the cold-start problem when zero or only a few items are observed in a session.

Finally, ISCON has a concatenation layer that concatenates the above representations ($z_{context}$, $z_{item}$, $E_u$) and feeds them to fully connected and Softmax layers to generate next-item recommendation probabilities $\hat{p} \in \mathbb{R}^{|I|}$. When $|I|$ is the number of items.

We train the next-item predictor $\Theta_{next-item} = \{\Theta_{item}, E_u, E^C, FC_2\}$ with the following Cross-Entropy loss and Adam optimizer on all training interactions $X_{train}$. For training in a supervised manner, we use the implicit contexts of the sessions derived in Section 3.1 as ground-truth labels. The training loss is given as follows:

$$L_{context}(\hat{y}) = -\sum_{i=1}^{|I|} y_i \log(\hat{y}_i)$$

where $y$ is a one-hot vector containing the implicit context assignment of a current session $s$ of a user $u$ (the output of Section 3.1)

Using the trained session context prediction model $\Theta_{context}$, ISCON generates a session context probability vector ($\hat{y} \in \mathbb{R}^{|I|}$).

### 3.3 Next-item Predictions with Session Contexts

The ultimate goal of ISCON is to enhance the next-item prediction accuracy by contextualizing sessions and utilizing those session contexts as indicators. Figure 2 shows the ISCON architecture, where it combines predicted session context embeddings $z_{context}$, an item-level interest $z_{item}$, and a user embedding $E_u$ for personalization.

First, to contextualize the predictions, we use the predicted session context information from Section 3.2. Specifically, for each session, we select the top-$K^3$ contexts with the highest probabilities predicted by the context predictor and concatenate their session context embedding vectors together. Since session contexts serve as a high-level summary of the session, the concatenation of the top-$K$ predicted context embeddings provides an accurate representation of the user’s interest in the current session. Taking $K$ contexts instead of only one context increases the breadth of predictions and prevents erroneous predictions due to wrong context predictions from Section 3.2.

Second, similar to the context predictor (Section 3.2), we summarize a user’s item-level current interest within a session via a Bi-LSTM. Given a session $s$ of a user $u$ and its predicted top-$K$ contexts $\hat{C}_1, \ldots, \hat{C}_K$ (ordered by context IDs), the predicted session context representation $\hat{z}_{context}$ is given as follows:

$$\hat{z}_{context} = \text{concat}(E_{\hat{C}_1}^C, \ldots, E_{\hat{C}_K}^C)$$

where $E_{\hat{C}^C}$ is a trainable embedding of a context $c$. All users share the same context embeddings.

Finally, ISCON has a concatenation layer that concatenates the above representations ($z_{context}$, $z_{item}$, $E_u$) and feeds them to fully connected and Softmax layers to generate next-item recommendation probabilities $\hat{p} \in \mathbb{R}^{|I|}$. When $|I|$ is the number of items.

We train the next-item predictor $\Theta_{next-item} = \{\Theta_{item}, E_u, E^C, FC_2\}$ with the following Cross-Entropy loss and Adam optimizer on all training interactions $X_{train}$.

$$L_{next-item}(\hat{p}) = -\sum_{i=1}^{|I|} p_i \log(\hat{p}_i)$$

where $p$ is a one-hot vector containing the ground-truth next-item ($i_{k+1}$) of the current session $s$ of the user $u$.

Using the trained $\Theta_{next-item}$, ISCON generates the next-item probability vectors ($\hat{p} \in \mathbb{R}^{|I|}$) for all test interactions $X_{test}$.

There are no shared model parameters or joint-training between the context predictor and next-item predictor, since they are optimized to solve different tasks. The next-item predictor only utilizes context prediction results of sessions from the context predictor.

### 4 EXPERIMENTS

**Datasets:** Table 1 lists the statistics of the datasets. We created sessions of all datasets with a 1-hour idle threshold since the datasets
Table 1: Summary of datasets and sessions used for experiments.

| Name          | Users | Items | Interactions | Sessions | Avg session length |
|---------------|-------|-------|--------------|----------|-------------------|
| Gowalla [4]   | 69,332| 10,000| 1,250,045    | 915,135  | 1.20              |
| LastFM [12]   | 954   | 1,000 | 258,620      | 167,382  | 1.54              |
| Foursquare [44]| 2,321 | 5,596 | 194,105      | 42,881   | 4.49              |
| Reddit [1]    | 8,640 | 966   | 134,489      | 55,698   | 2.03              |

Table 2: Next-item prediction performance of ISCON and baselines.

| Model / Dataset | Gowalla | Foursquare | Reddit | LastFM |
|-----------------|---------|------------|--------|--------|
| GRU4REC [11]    | 0.27724 | 0.06696    | 0.63536| 0.12587|
| TAGNN [42]      | 0.32614 | 0.11189    | 0.67666| 0.13375|
| COTREC [38]     | 0.17464 | 0.11119    | 0.45382| 0.09295|
| CSRM [33]       | 0.31326 | 0.12807    | 0.68894| 0.13190|
| ISCON (proposed)| 0.35979 | 0.17483    | 0.72665| 0.13838|

*a Mean reciprocal rank (MRR) of ISCON and baselines* 

- Gowalla [4, 40, 44] is a point-of-interest (POI) dataset collected in the US, represented as (user, location, timestamp).
- LastFM [8, 12, 24] includes the music playing history of users represented as (user, music, timestamp).
- Foursquare [40, 44] is a POI dataset collected from Singapore, which is represented as (user, location, timestamp).
- Reddit [1, 17] includes the posting history of users on subreddits represented as (user, subreddit).

**Baselines:** As a baseline, we use four state-of-the-art session-based recommender models: (1) GRU4REC [11]: it utilizes diverse ranking-based loss functions and additional data samples for higher accuracy, (2) TAGNN [42]: it uses a graph neural network and a target-aware attention module for prediction, (3) COTREC [38]: it combines self-supervised learning with graph co-training, and (4) CSRM [33]: it contextualizes the current and neighborhood sessions with inner and outer memory encoders, respectively.

**Experimental Setup:** Following [13, 22, 35], we use first 81%, mid-9%, last 10% of interactions (sorted by timestamps) of each dataset for training, validation, and test, respectively. The default hyperparameters of ISCON are set as follows: the number of session contexts is 40, the number of predicted contexts per session is 3, the user and item embedding sizes are 256, and the contextual embedding size is 32. Moreover, the maximum training epoch is 200, a learning rate is 0.001, the batch size is 1024, and the maximum sequence length per user is 50. For baselines, we use hyperparameters recommended in their original publications.

**4.1 Next-item Prediction Accuracy**

Table 2 shows Mean Reciprocal Rank (MRR) [31] and Recall@10 metrics of ISCON and the state-of-the-art methods on the four datasets. Among all methods, ISCON mostly shows the best next-item prediction performance, with statistical significance (p-values < 0.05), according to both the metrics across all four datasets. On the Foursquare dataset, which is the hardest-to-predict dataset for baselines, ISCON presents at least 36.5% and 34.0% improvements in MRR and Recall@10 metrics compared to the baselines.

**4.2 Session Contexts Evaluation**

ISCON finds the implicit contexts of sessions via clustering of session embeddings. Here, we verify the correctness of the derived session contexts. Since there is no ground-truth session context information available, we conduct a manual verification. We use the Reddit dataset as it includes information about the items (i.e., subreddits), such as the subreddit name, and text and title of posts (i.e., items). We first choose the top-10 clusters by size. For each cluster, we select the 100 sessions closest to the cluster center in the embedding space. After that, we analyze the items (posts) in the sessions of each cluster and manually verify if the sessions are semantically similar. If they are, we assign a context to the cluster.

Figure 3 shows a visualization of session contexts found on the Reddit dataset, where each data point is a session, and its color represents its cluster. Data points with a light gray color indicate randomly sampled sessions that are not in the top-10 clusters. We use t-SNE [29] to map and visualize the session embeddings to two-dimensional space. As shown in the figure, we find distinct contexts of session clusters like Trading, Music, and News topics. The dense clusters and their reasonable contextual meanings substantiate the correctness of the session contextualization approach of ISCON.

**5 CONCLUSION**

We showed that by assigning and predicting session contexts, next-item recommendations can be improved. This work can help recommendation engines better understand their users by identifying implicit contexts. Future works include handling cold-start users and items effectively with their metadata and optimizing different neural architectures (e.g., Transformer) as the backbone of ISCON.

**ACKNOWLEDGMENTS**

This research is supported in part by Georgia Institute of Technology, IDEaS, Adobe, and Microsoft Azure. S.O. was partly supported by ML@GT, Twitch, and Kwanjeong fellowships. We thank the reviewers for their feedback.
