# Modeling Multi-Purpose Sessions for Next-Item Recommendations via Mixture-Channel Purpose Routing Networks

**Shoujin Wang[1], Liang Hu[2,3], Yan Wang[1], Quan Z. Sheng[1], Mehmet Orgun[1]** and **Longbing Cao[2]**

[1]Department of Computing, Macquarie University
[2]Advanced Analytics Institute, University of Technology Sydney
[3]University of Shanghai for Science and Technology
{shoujin.wang, yan.wang}@mq.edu.au, rainmilk@gmail.com

## Abstract

A *session-based recommender system* (SBRS) suggests the next item by modeling the dependencies between items in a session. Most of existing SBRSs assume the items inside a session are associated with one (implicit) purpose. However, this may not always be true in reality, and a session may often consist of multiple subsets of items for different purposes (e.g., breakfast and decoration). Specifically, items (e.g., bread and milk) in a subset have strong purpose-specific dependencies whereas items (e.g., bread and vase) from different subsets have much weaker or even no dependencies due to the difference of purposes. Therefore, we propose a *mixture-channel* model to accommodate the *multi-purpose* item subsets for more precisely representing a session. To address the shortcomings in existing SBRSs, this model recommends more diverse items to satisfy different purposes. Accordingly, we design effective *mixture-channel purpose routing networks* (MCPRNs) with *a purpose routing network* to detect the purposes of each item and assign them into the corresponding channels. Moreover, a *purpose-specific recurrent network* is devised to model the dependencies between items within each channel for a specific purpose. The experimental results show the superiority of MCPRN over the state-of-the-art methods in terms of both recommendation accuracy and diversity.

## 1 Introduction

In real world, user requirements keep changing when the corresponding contexts evolve. This context-sensitive demand nature inspires the recent interest on session-based recommender systems (SBRSs) [Wang *et al.*, 2019], which tackle the gaps in the conventional static recommender systems (RSs) that only recommend homogeneous items without considering the demand dynamics across sessions. Accordingly, an SBRS recommends the next item for a session (as a context) to a user who may be interested in by modeling the dependencies between items in the session [Hidasi *et al.*, 2018].

Most of existing SBRSs (e.g., [Twardowski, 2016; Wang *et al.*, 2017; 2018]) make recommendations by forming a ses-
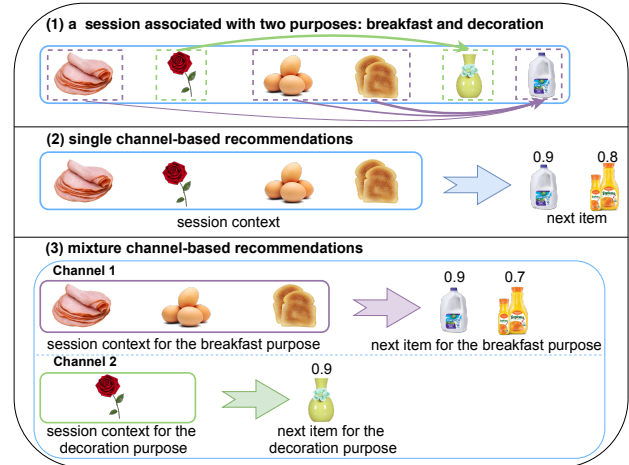


Figure 1: An example of two different approaches of SBRSs to model a multi-purpose session, where dashed squares with different colours indicate different purposes and thicker arrow lines show stronger dependencies between items.

sion with a single purpose or goal (e.g, shopping for food and beverage products in basket analysis). This often violates the reality that a session may involve multiple types of items where each type corresponds to one purpose. Taking the session illustrated by Row 1 in Figure 1 as an example, Janet first placed bacon into a cart for breakfast while she then was attracted by a piece of lovely rose for decoration and added it into the cart. She further chose eggs and bread for breakfast and a vase for fitting the rose. Finally, she ended this shopping session by picking up a bottle of milk. For this example, existing SBRSs would take a single-channel modeling approach to implicitly associate the items purchased in the session as a homogeneous item sequence, as shown in Row 2 in Figure 1. Obviously, such an approach fails to differentiate the purpose-specific item dependencies (e.g., items for breakfast vs. items for decoration) when the items in a session serve multiple purposes.

The above example reveals significant gaps in two representative types of state-of-the-art SBRSs, namely the single-channel recurrent neural networks (RNN) based SBRSs and attention-based SBRSs. RNN-based models [Hidasi *et al.*, 2018; Quadrana *et al.*, 2017] assume a rigid sequential dependency over any successive items within a session and hence:

(Gap 1) *it is easy to generate false dependencies as not all items depend on each other in a multi-purpose session.* In addition, due to the memory decay along with the time steps, those contextual items far from the target item will be overwhelmed by the near ones, as a result, (Gap 2) *an RNN-based SBRS tends to recommend items biased to the most recent purpose indicated by the nearest items.* To reduce the interference of irrelevant items, an attention mechanism has been recently incorporated into shallow networks [Wang *et al.*, 2018], RNN [Li and et al., 2017] or memory networks [Chen *et al.*, 2018] to build SBRSs. The attention model tends to assign salient weights on a very few significant items [Hu *et al.*, 2018], which makes the purpose of a session dominated by these few items. Consequently, (Gap 3) *an attention-based SBRS tends to recommend items to satisfy the dominant purpose while disadvantages others.*

This paper addresses the above three gaps by proposing *mixture-channel purpose routing networks* (MCPRN). MCPRN first automatically detects the possible multiple purposes in a session with a *purpose routing network* (PRN), and then models a session with a *mixture-channel recurrent networks* (MCRN) where each *channel* (i.e., a *purpose-specific recurrent network* (PSRN) built on *purpose-specific recurrent units* (PSRUs) models the dependencies between items for a specific purpose to build a purpose-specific embedding. Finally, a multi-purpose context (session) embedding is built by integrating all the channels to rank the candidate items for diverse purpose-sensitive recommendations.

Thanks to the PRN and MCRN, the different purposes in a session are detected and then their item dependencies are respectively modeled in different channels. As a result, the item dependencies are modeled over specific purposes instead of a coarse one, which enables our model to more concentratedly capture purpose-driven user behavior. Consequently, MCPRN leverages the current single-purpose SBRSs with a more effective and robust mechanism to represent multi-purpose sessions. By retaining and modeling the multiple purposes in a session, MCPRN can recommend diverse items satisfying different purposes, while existing methods tend to recommend items for a single purpose only, as shown in Row 3 and Row 2 in Figure 1 respectively. The main contributions of this work are summarized below:

- We propose multi-purpose modeling to capture user behavior in a session in a more reasonable way to fit real cases. Accordingly, *mixture-channel purpose routing networks* (MCPRN) are proposed to achieve this.

- In the MCPRN, the *purpose routing network* (PRN) is devised to infer the purposes of each item and route them into specific channels. Moreover, the *purpose-specific recurrent unit* (PSRU) is the key component to serve as the basic cells of the *mixture-channel recurrent networks* (MCRNs) w.r.t. each channel.

## 2 Related Work

A variety of SBRSs including rule/pattern-based RSs [Yap *et al.*, 2012], neighbourhood-based RSs [Jannach and Ludewig, 2017], Markov chain-based RSs [Feng and et al., 2015] and factorization machine-based RSs [Rendle *et al.*, 2010] have

been developed. We briefly review two representative state-of-the-art SBRSs: (1) RNN-based SBRSs, and (2) attention-based SBRSs, which are the most relevant ones to our work.

Due to its strength in handling sequential data, RNN is an intuitive choice to capture the complex intra-session dependency in SBRSs. The first RNN-based SBRS, GRU4Rec [Hidasi *et al.*, 2016], employed gated recurrent unit (GRU) to capture the long-term dependency within sessions. Later, the performance of GRU4Rec was improved significantly by introducing novel ranking loss functions tailored to RNNs in the recommendation setting [Hidasi *et al.*, 2018]. RNN is easy to generate false dependencies due to the employed rigid order assumption which assumes any adjacent items in a session are highly sequentially dependent. However, this may not be true in most real cases. In addition, RNN usually biases to recent items while missing much information of previous items in a session because of memory decay.

The attention mechanism [Vaswani *et al.*, 2017] has been employed to build more robust SBRSs by intensifying those relevant and important items in a session context. [Li and et al., 2017] proposed the Neural Attentive Recommendation Machine (NARM), in which a hybrid encoder with an attention mechanism is employed to model the user's sequential behavior and the user's main purpose in the current session by differentiating more and less important items. [Wang *et al.*, 2018] designed an attention-based transaction embedding model (ATEM) to build an attentive context embedding over the session context without order assumption. Further, the attention mechanism was used in a memory-augmented neural network for selectively reading out the external memory matrix for next item recommendations [Chen *et al.*, 2018]. However, the attention mechanism attempts to assign larger weights on few significant items while downplaying others, leading to bias to the main purpose indicated by those few items. As a result, attention-based SBRSs often only cater for the main purpose while ignoring others in a session.

In summary, all the aforementioned SBRSs are single channel-based, which are effective for one-purpose sessions in which all items within a session are dependent on each other. However, they cannot handle sessions with different types of items for multiple purposes well. Inspired by the great potential of mixture models in handling multiple kinds of relations [Shazeer *et al.*, 2017; Kang *et al.*, 2018], we devise a mixture-channel model for multi-purpose sessions.

## 3 Problem Statement

Given a session set $S = \{s_1, ..., s_{|S|}\}$, each session $s = \{v_1, ..., v_{|s|}\}(s \in S)$ consists of a sequence of items that are interacted (e.g., clicked or purchased) sequentially by an anonymous user in one transaction event. Here $|S|$ denotes the number of sessions in $S$ and the subscripts in $s$ indicate the order of item occurrence. All the items occurring in all sessions constitute the universal item set $V = \{v_1, ..., v_{|V|}\}$. For a target item $v_t \in s$, all the items that occurred prior to $v_t$ in $s$ together form the session context (called context for short) of $v_t$ over $s$, represented as $C_{v_t} = \{v_1, ..., v_{t-1}\}$. Each item in $C_{v_t}$ is called a contextual item.

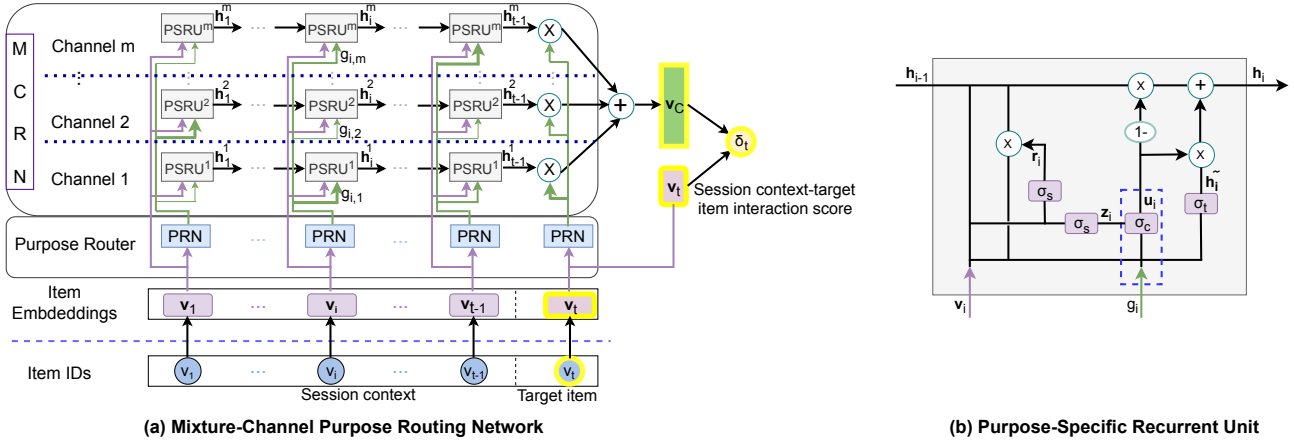Given a context $C$ with precedent $t - 1$ items, an SBRS

Figure 2: (a) The MCPRN model consists of two main modules: Purpose Router and Mixture-Channel Recurrent Networks; (b) The PSRU cell introduces a concentration gate (see the blue dash line square) to model purpose-specific transitions.

is built to recommend the $t^{th}$ item. Accordingly, MCPRN is trained as a probabilistic classifier model that learns to predict a conditional probability distribution $P(v_t|C)$. Once all the parameters of the model are learned, the MCPRN is able to recommend the next item by ranking all the candidate items in terms of their conditional probability over the given context.

## 4 Mixture-Channel Purpose Routing Networks

The architecture of MCPRN is illustrated in Figure 2 (a). MCPRN mainly consists of two modules: (1) Purpose Router, and (2) Mixture-Channel Recurrent Networks (MCRNs). The purpose router is employed to route each item to a channel of a specific purpose. Specially, we adopt a soft routing strategy to assign each item embedding to all channels with different purpose weights, so-called mixture channel networks. The soft routing strategy enables to conduct any gradient-based optimization for easing learning. In MCRNs, each channel is equipped with a PSRN built on PSRUs to model the dependencies between items for a specific purpose. The final hidden states $\mathbf{h}_{t-1}$ from different channels are selectively integrated as a multi-purpose context embedding $\mathbf{v}_C$. The embedding of target item $v_t$ conditional on $\mathbf{v}_C$ is used to calculate the probability of selection as the next item. Next, we present more technical details of these components.

### 4.1 Purpose Router

Given all the items in a session context, a purpose routing network (PRN) is employed in Purpose Router to extract the purposes of selecting an item $v_i$ without any human prior knowledge. First, we map each item $v$ into a $K$-dimensional embedding vector $\mathbf{v} \in \mathbf{W}_e$, where $\mathbf{W}_e \in \mathbb{R}^{K \times |V|}$ is the embedding matrix of all items. Then, we input the embedding $\mathbf{v}_i$ of item $v_i$ into the PRN to identify the purposes of selecting $v_i$. $\mathbf{W}_p \in \mathbb{R}^{K \times m}$ denotes the purpose filtering parameter of all purposes, where $m$ is the number of possible purposes (i.e., the number of channels) and it can be determined by cross-validation. Then, the concentration score $a_{i,j}$ of item

$v_i$ w.r.t. the $j^{th}$ purpose $p_j$ is computed as follows:

$$a_{i,j} = \mathbf{v}_i^{\top} \mathbf{W}_p[:,j], \quad j \in \{1, ..., m\} \tag{1}$$

where $\mathbf{W}_p[:,j]$ denotes the $j^{th}$ column of $\mathbf{W}_p$. Further, the normalized concentration weight w.r.t. $p_j$ can be obtained in terms of the following softmax function:

$$g_{i,j} = \frac{\exp(a_{i,j}/\tau)}{\sum_{h=1}^{m} \exp(a_{i,h}/\tau)} \tag{2}$$

where $\tau$ is a temperature parameter to tune. For high temperatures ($\tau \to \infty$), all purposes have nearly the same probability. For a low temperature ($\tau \to 0^+$), it tends to concentrate on a single purpose (nearly a hard routing). In experiments, we use $\tau = 0.1$ which can produce the best performance.

### 4.2 Mixture-Channel Recurrent Networks

Then, the item embedding and its purpose concentration weight are input into each channel of MCRN as shown by the green and purple arrow lines respectively in Figure 2 (a). Each channel is modeled by a PSRN composed of $t-1$ PSRU cells to model the purpose-specific sequential dependency over a sequence of $t - 1$ items. The working mechanism of each channel is similar to a normal RNN whereas the cells are equipped with our designed PSRU cells in PSRNs. The PSRU cells in one channel are identical while cells in different channels share an identical structure but different parameters. We present PSRU in detail in the following subsection.

#### Purpose-Specific Recurrent Units

RNN cells like long short-term memory (LSTM) or GRU do not consider the degree of membership of items in a sequence. In MCRNs, each channel contains all items in a session but their purpose concentration weights, i.e. the degree of membership, on this purpose are different. Hence, LSTM and GRU are not ready for modeling such purpose-specific dependencies in a channel given the concentration weights.

As a result, we designed the purpose-specific recurrent unit (PSRU) to serve as the cell for each PSRN. Different from GRU or its variants, which usually compute a gate value by only using the current input and the last hidden state without

any additional information, PSRU introduces a concentration gate to selectively integrate the item information in the transitions w.r.t. a specific purpose.

The structure of a PSRU cell is given in Figure 2 (b). Compared to the traditional GRU cell, an additional gate, namely concentration gate (see the blue dash line square), is added to decide to what extent the current state should be involved into the purpose-specific transitions according to the purpose concentration weight $g_{i,j}$. The PSRU cell at the $i^{th}$ time step in channel $c_j$ takes the last hidden state $h_{i-1}$, the embedding of the current item $v_i$ and the concentration weight $g_i$ (here subscript $j$ is omitted from $g_i$ when the channel $c_j$ is given) of $v_i$ as the input, and outputs the candidate hidden state $\widetilde{h}_i$ of current time step. More specifically, the candidate hidden state $\widetilde{h}_i$ can be computed as follows, with the reset gate vector $r_i$ and the update gate vector $z_i$:

$$r_i = \sigma_s(W_r[h_{i-1}, v_i]) \tag{3}$$

$$z_i = \sigma_s(W_z[h_{i-1}, v_i]) \tag{4}$$

$$\widetilde{h}_i = \sigma_t(W_h[r_i * h_{i-1}, v_i]) \tag{5}$$

where $\sigma_s$ and $\sigma_t$ are activation functions and are specified as sigmoid and tanh respectively. Further, we obtain the concentration gate vector $u_i$ as below:

$$u_i = \sigma_c(g_i, z_i; \epsilon) = \delta_c(g_i \geq \epsilon) * g_i * z_i \tag{6}$$

where the Delta function $\delta_c = 1$ if $g_i \geq \epsilon$ and 0 otherwise. $\epsilon$ is a threshold parameter to eliminate noisy items in modeling the transitions of a session. That is, it will bypass the item with a small concentration weight less than $\epsilon$. We empirically set $\epsilon = 0.01$ in our experiment.

As a result, the hidden state of current step, $h_i$, can be determined by $u_i$ in terms of previous state $h_{i-1}$ and current candidate state $\widetilde{h}_i$:

$$h_i = (1 - u_i) * h_{i-1} + u_i * \widetilde{h}_i \tag{7}$$

For example, if item $v_i$ majorly concentrate on a single purpose $p_1$, i.e., $g_{i,1} \approx 1$ while $g_{i,j} \approx 0$ $(j \neq 1)$. Due to $g_{i,j} < \epsilon$ for $(j \neq 1)$, item $v_i$ will be ignored in the transitions for all other channels except Channel 1, i.e. $h_i = h_{i-1}$.

### 4.3 The Probability of Selection

Once the session context $C$ has been input into MCPRN, the final hidden states $h_{t-1}$ of all $m$ channels are integrated together to build a multi-purpose context embedding $v_C$:

$$v_C = \sum_{j=1}^{m} \hat{g}_{t,j} * h_{t-1}^j \tag{8}$$

where $\hat{g}_{t,j}$ is the concentration weight of target item $\hat{v}_t$ w.r.t. purpose $p_j$. $\hat{g}_{t,j}$ weights target purpose to construct the context embedding $v_C$ that probably output the target item $\hat{v}_t$.

Then, we feed the context embedding $v_C$ together with the embedding of the candidate item into the output layer for the target item prediction. Specifically, a score that quantifies the relevance of the target item $\hat{v}_t$ w.r.t. the given context $C$ is computed using the inner product to capture the interaction between them.

$$\delta_t(C) = \hat{v}_t^\top v_C \tag{9}$$

---

**Algorithm 1** MCPRN parameter learning procedure

1: $\mathcal{B} \leftarrow$ Get mini-batch from all context-target item pairs
2: $\mathcal{N} \leftarrow$ Sampling $n$ negative items, $S_t^-$, for each target item $v_t \in \mathcal{B}$
3: Compute mini-batch loss using Eq. 11:
   $\mathcal{L}_{\mathcal{B}} \leftarrow \frac{1}{|\mathcal{B}|} \sum_{<s,S^->\in<\mathcal{B},\mathcal{N}>} \mathcal{L}(s^+, S^-)$
4: Update parameters: $\Theta \leftarrow \Theta - \Gamma_{Adam}(\nabla_\Theta \mathcal{L}_{\mathcal{B}})$

---

Finally, the conditional probability $q(\hat{v}_t|C; \Theta)$ to select the item $\hat{v}_t$ is obtained in terms of the score $\delta_t(C)$:

$$q(\hat{v}_t|C; \Theta) = \frac{1}{1 + e^{-\delta_t(C)}} \tag{10}$$

where $\Theta$ is the set of model parameters that need to be learned over all sessions.

### 4.4 Optimization and Training

We train our model by using a mini-batch gradient descent on the cross-entropy loss. Given the conditional probability $q$ of selection, the loss function is:

$$\mathcal{L}(s^+, S^-) = -[\log(q_{s^+}) + \sum_{s \in S^-} \log(1 - q_s)] \tag{11}$$

Given a session context $C$, we build a contrastive pair by taking the true next item $v_t$ as the positive sample $s^+$ and then randomly sample $n$ items from the item set $I \setminus v_t$ as the negative sample set $S^-$. The loss from the positive sample and each negative sample are $log(q_s)$ and $log(1 - q_s)$ respectively. They together form the loss of a contrastive pair $\langle s^+, S^- \rangle$. The model parameters $\Theta$ are learned by minimizing $\mathcal{L}(s^+, S^-)$. Specifically, negative sampling [Goldberg and Levy, 2014] is employed for efficient training.

Our model is implemented using Tensorflow. Due to the limited space, only a brief scheme of the learning procedure on a mini-batch is listed in Algorithm 1, where $\Gamma_{Adam}$ denotes Adam [Kingma and Ba, 2015]-based gradient descent optimizer and the batch size is set to 50.

## 5 Experiments and Evaluation

### 5.1 Data Preparation

Three real-world transaction datasets are used for experiments: (1) Yoochoose-buys[1] released by RecSys Challenge 2015, which records the purchased items in each session on Yoochoose.com; (2) Tmall[2] released by IJCAI-15 competition, which records the purchased items in each transaction on Tmall online shopping platform; and (3) Tafeng[3] released on Kaggle, which contains the transaction data of a Chinese grocery store.

First, a set of sessions is extracted from each original dataset by putting all items in one transaction together to form a session. Those sessions containing less than three items are removed since at least two items should be used as the context

---

[1]https://2015.recsyschallenge.com/challenge.html

[2]https://tianchi.aliyun.com/dataset/dataDetail?dataId=42

[3]https://www.kaggle.com/chiranjivdas09/ta-feng-grocery-dataset

| | Yoochoose-buy | | | | Tmall | | | | Tafeng | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | MRR@5 | MRR@20 | nDCG@5 | nDCG@20 | MRR@5 | MRR@20 | nDCG@5 | nDCG@20 | MRR@5 | MRR@20 | nDCG@5 | nDCG@20 |
| iGRU4Rec-BPR | 0.1150 | 0.1271 | 0.1314 | 0.1662 | 0.1849 | 0.1920 | 0.1994 | 0.2198 | 0.0101 | 0.0124 | 0.0135 | 0.0214 |
| iGRU4Rec-CE | 0.1538 | 0.1715 | 0.1776 | 0.2268 | 0.2247 | 0.2318 | 0.2424 | 0.2623 | 0.0145 | 0.0188 | 0.0185 | 0.0308 |
| NARM | 0.1514 | 0.1734 | 0.1768 | 0.2387 | 0.4213 | 0.4246 | 0.4532 | 0.4618 | 0.0754 | 0.0862 | 0.0882 | 0.1198 |
| MANN | 0.1653 | 0.1893 | 0.1986 | 0.2655 | 0.4166 | 0.4227 | 0.4520 | 0.4610 | 0.0171 | 0.0288 | 0.0318 | 0.0625 |
| Caser | 0.1604 | 0.185 | 0.1918 | 0.2444 | 0.2317 | 0.2513 | 0.2646 | 0.2735 | 0.0155 | 0.0206 | 0.0261 | 0.0335 |
| ATEM | 0.1212 | 0.1324 | 0.1768 | 0.2026 | 0.2012 | 0.2195 | 0.2238 | 0.2412 | 0.0216 | 0.0312 | 0.0282 | 0.0310 |
| MCPRN-S | 0.1680 | 0.1861 | 0.1859 | 0.2410 | 0.3552 | 0.3634 | 0.3907 | 0.4121 | 0.0446 | 0.0572 | 0.0581 | 0.0931 |
| MCPRN | **0.2191** | **0.2332** | **0.2547** | **0.2927** | **0.4657** | **0.4694** | **0.4947** | **0.5098** | **0.1017** | **0.1047** | **0.1112** | **0.1218** |
| Improvement (%) | 32.49 | 23.19 | 28.25 | 10.24 | 10.54 | 10.55 | 9.16 | 10.39 | 34.88 | 21.46 | 26.08 | 1.67 |

Table 1: Recommendation accuracy on three datasets

| Statistics | Yoochoose-buy | Tmall | Tafeng |
|---|---|---|---|
| #Sessions | 83,928 | 144,936 | 19,538 |
| #Items | 7,428 | 27,863 | 5,263 |
| Avg. session length | 5.13 | 4.09 | 7.41 |
| #Item category | n.a | 786 | 793 |

Table 2: Statistics of experimental datasets

to form a multi-purpose session context and one as the target item. Second, we do three training-test splits on the session set by randomly selecting 20%, 30%, 40% of the sessions from the last 30 days respectively for testing. Our method achieved similar performance on all the three splits and stably outperformed all the baselines and thus only the results from the 30% split is shown. Finally, to build the training-test instances of format $\langle C, v_t \rangle$, for one session $s$, each time one item from the third to the last item in $s$ is picked up as the target item $v_t$ and all those items before $v_t$ constitute its corresponding context $C$. Subsequently, for a session $s$ containing $|s|$ ($|s| \geq 3$) items, $|s| - 2$ instances are built in total. The characteristics of experimental datasets are shown in Table 2.

## 5.2 Experimental Settings

Three commonly used metrics: Mean Reciprocal Rank (MRR), normalized Discounted Cumulative Gain (nDCG) and Recall, are applied to evaluate the recommendation accuracy [Chen *et al.*, 2018; Wang *et al.*, 2018], together with a newly-designed diversity measure to evaluate the diversity.

**Comparison Methods**

To demonstrate the efficacy of mixture channels for multi-purpose modeling, we implemented two versions of our method: (1) full model of **MCPRN** proposed in this work; and (2) single channel network (**MCPRN-S**) with only one channel retained from MCPRN. The following representative state-of-the-art SBRSs built on various frameworks including RNN, attention model, memory network, convolutional neural network (CNN) and shallow network, are selected to be the baselines:

- **iGRU4Rec-BPR**: An improved version of a typical RNN-based SBRS which models a session using RNNs built on GRU. It takes Bayesian Personalized Ranking (BPR) as the loss function [Hidasi *et al.*, 2018].

- **iGRU4Rec-CE**: Similar to iGRU4Rec-BPR except that it replaces the loss function BPR with cross entropy [Hidasi *et al.*, 2018].

- **NARM**: A hybrid encoder with an attention mechanism to users' sequential behavior and capture users' main purpose in a session [Li and et al., 2017].

- **MANN**: A memory-augmented neural network which employs an attention model to read out historical information explicitly stored in the external memory matrix [Chen *et al.*, 2018].

- **Caser**: A convolutional sequence embedding model which embeds a sequence of items into an "image" and then learns sequential patterns as local features using convolutional filters [Tang and Wang, 2018].

- **ATEM**: A shallow and wide network with the attention mechanism incorporated to learn an attentive embedding of a session context [Wang *et al.*, 2018].

**Parameter Settings**

We initialize all the baseline models with the parameter settings in the corresponding papers and then tune them on our datasets for best performance for a fair comparison. The sizes of item embeddings and hidden states in PSRU are set to 128. The number of channels $m$ is set to 3 by tuning on the validation data. The initial learning rate for Adam is set to 0.001.

## 5.3 Recommendation Accuracy Evaluation

Extensive experiments are conducted to answer the following questions:

- **Q1:** How does our approach compare to the state-of-the-art SBRSs in terms of recommendation accuracy?

- **Q2:** How do the mixture-channel networks compare to the single channel network?

**Result 1: Comparison with Baselines w.r.t. Accuracy**

To answer question Q1, we compare the recommendation accuracy of our method MCPRN and those of the six state-of-the-art SBRSs. Table 1 reports the MRR and nDCG. The first two methods are built on GRU-based RNN, which can easily generate false dependencies due to the overly strong sequentially dependent-based assumption and bias to the most recent purpose. Therefore, they do not perform well on multi-purpose session data. NARM and MANN achieve better performance by incorporating the attention mechanism into an RNN-based and memory network-based SBRS respectively to emphasize the relevant information and reduce noise. But they are easy to bias to the main purpose in a session due to the attention weighting mechanism. Caser does not perform
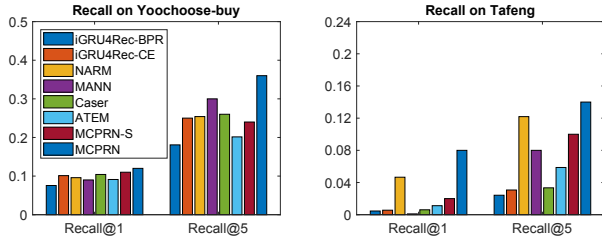
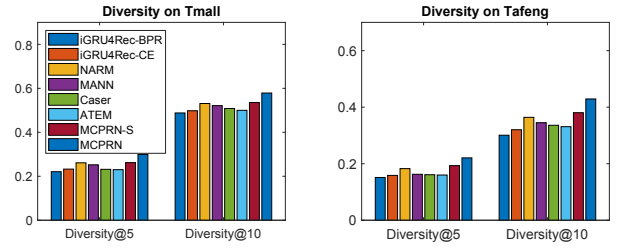Figure 3: Recalls of MCPRN and other compared methods



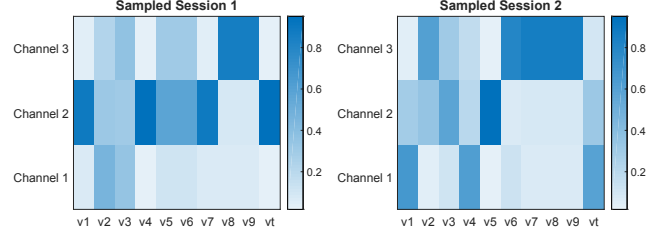Figure 4: Diversities of MCPRN and other compared methods



Figure 5: Purpose concentration weights w.r.t. two sampled sessions containing 10 items, where the first 9 items serve as the contextual items and the last one corresponds to the target item.

well either as it is hard for the pooling operation in CNN to capture the long-range dependencies. ATEM usually biases to the main purpose as other attention-based SBRSs do. By modeling each purpose independently with a recurrent channel, our MCPRN not only treats each purpose equally but also keeps the sequential dependencies within each purpose. As a result, MCPRN achieves the best performance on all datasets. Particularly, MCPRN demonstrates over 20% improvement over the best existing method in terms of MRR@5, MRR@20 and nDCG@5 on the last two datasets (cf. Table 1). The recall (cf. Figure 3) on two datasets (due to the limited space) shows MCPRN leads the baselines with a clear margin.

### Result 2: Mixture Channels vs. Single Channel

To demonstrate the efficacy of mixture-channel structure, we compared MCPRN with MCPRN-S. It is clear that MCPRN achieves much higher accuracy than MCPRN-S as shown in Table 1 and Figure 3. The MRR, nDCG and Recall@5 of MCPRN are at least 20% higher than that of MCPRN-S, proving the superiority of the mixture-channel architecture.

## 5.4 Recommendation Diversity Evaluation

Diversity evaluation has been introduced to make up the limitation of accuracy [Hu and et al., 2017; Yao *et al.*, 2019b]. Experiments are conducted to answer the following question:

- **Q3:** How does our approach compare to the state-of-the-art SBRSs in terms of recommendation diversity?

Intuitively, the more disperse distribution over the item categories of the recommendations means the larger diversity can be produced by the model. As a result, the category distribution of recommendations can be measured by the entropy to quantify the diversity, according to the information theory [Zhang *et al.*, 2016; Yao *et al.*, 2019a].

$$\mathcal{D}iversity@k = -\sum_{i=1}^{k} Pr_i \log_2 Pr_i \qquad (12)$$

where $k$ is the number of top-rank items in the recommendation list while $Pr_i$ is the probability of the category to which item $v_i$ belongs and is estimated by the frequency.

### Result 3: Comparison with Baselines w.r.t. Diversity

Figure 4 shows that MCPRN achieves higher diversity than compared methods, especially on the Tafeng dataset where MCPRN achieves up to 13.01% and 11.61% improvement w.r.t. Diveristy@5 and Diveristy@10 respectively compared to the best baseline method. The reasons are that the baseline methods usually recommend items that only satisfy either the recent purpose (e.g., RNN-based SBRSs) or the main purpose

(e.g., attention-based SBRSs) due to their single channel-based design. In contrast, MCPRN easily diversifies the recommendation list by modeling multiple purposes in a parallel way. Diversity evaluation is not applicable to Yoochoose-buy as its category information is unavailable (cf. Table 2).

## 5.5 Purpose Concentration Visualization

To get a deep insight into how the items for different purposes in a session are detected and accordingly routed into different channels in MCPRN, we visualize the concentration weights (cf. Eq. 2) of two sessions randomly sampled from the test set of Yoochoose-buy dataset in Figure 5.

Two observations can be made from Figure 5: (1) Items in one session are usually for multiple purposes and thus are put into different channels, as illustrated by the various distributions of different items on all the three channels; (2) The true target item $v_t$ may have purposes different from either the most recent purpose or the main purpose in a session.

## 6 Conclusions

To effectively detect and model the multiple purposes embedded within a session and thus to recommend corresponding diverse items to satisfy those different purposes, which cannot be addressed by existing session-based recommender systems, in this paper, we have proposed mixture-channel purpose routing networks (MCPRN). MCPRN harness a purpose routing network to detect the possible purposes of each item in a session and multiple channels to model the item dependencies within different purposes independently. Empirical evaluations on the real-world transaction data show the superiority of MCPRN in addressing the gaps in the state-of-the-art methods.

## Acknowledgements

# References

[Chen *et al.*, 2018] Xu Chen, Hongteng Xu, and et al. Sequential recommendation with user memory networks. In *Proceedings of the 11th ACM International Conference on Web Search and Data Mining*, pages 108–116, 2018.

[Feng and et al., 2015] Shanshan Feng and et al. Personalized ranking metric embedding for next new poi recommendation. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, pages 2069–2075, 2015.

[Goldberg and Levy, 2014] Yoav Goldberg and Omer Levy. word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *CoRR*, abs/1402.3722, 2014.

[Hidasi *et al.*, 2016] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. In *Proceedings of the Fourth International Conference on Learning Representations*, pages 1–10, 2016.

[Hidasi *et al.*, 2018] Balázs Hidasi, Alexandros Karatzoglou, and Alexandros Karatzoglou. Recurrent neural networks with top-k gains for session-based recommendations. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 843–852, 2018.

[Hu and et al., 2017] Liang Hu and et al. Diversifying personalized recommendation with user-session context. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 1858–1864, 2017.

[Hu *et al.*, 2018] Liang Hu, Songlei Jian, and et al. Interpretable recommendation via attraction modeling: Learning multilevel attractiveness over multimodal movie contents. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 3400–3406, 2018.

[Jannach and Ludewig, 2017] Dietmar Jannach and Malte Ludewig. When recurrent neural networks meet the neighborhood for session-based recommendation. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*, pages 306–310, 2017.

[Kang *et al.*, 2018] Wang-Cheng Kang, Mengting Wan, and Julian McAuley. Recommendation through mixtures of heterogeneous item relationships. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 1143–1152, 2018.

[Kingma and Ba, 2015] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015.

[Li and et al., 2017] Jing Li and et al. Neural attentive session-based recommendation. In *Proceedings of the 26th ACM International Conference on Information and Knowledge Management*, pages 1419–1428, 2017.

[Quadrana *et al.*, 2017] Massimo Quadrana, Alexandros Karatzoglou, and et al. Personalizing session-based recommendations with hierarchical recurrent neural networks. In *Proceedings of the 11th ACM Conference on Recommender Systems*, pages 130–137, 2017.

[Rendle *et al.*, 2010] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. Factorizing personalized markov chains for next-basket recommendation. In *Proceedings of the 19th International Conference on World Wide Web*, pages 811–820, 2010.

[Shazeer *et al.*, 2017] Noam Shazeer, Azalia Mirhoseini, and et al. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *Proceedings of the Fifth International Conference on Learning Representations*, pages 1–12, 2017.

[Tang and Wang, 2018] Jiaxi Tang and Ke Wang. Personalized top-n sequential recommendation via convolutional sequence embedding. In *Proceedings of the 11th ACM International Conference on Web Search and Data Mining*, pages 565–573, 2018.

[Twardowski, 2016] Twardowski. Modelling contextual information in session-aware recommender systems with neural networks. In *Proceedings of the 10th ACM Conference on Recommender Systems*, pages 273–276, 2016.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, and et al. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

[Wang *et al.*, 2017] Shoujin Wang, Liang Hu, and Longbing Cao. Perceiving the next choice with comprehensive transaction embeddings for online recommendation. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 285–302, 2017.

[Wang *et al.*, 2018] Shoujin Wang, Liang Hu, and et al. Attention-based transactional context embedding for next-item recommendation. In *32nd AAAI Conference on Artificial Intelligence*, pages 2532–2539, 2018.

[Wang *et al.*, 2019] Shoujin Wang, Longbing Cao, and Yan Wang. A survey on session-based recommender systems. *arXiv preprint arXiv:1902.04864*, 2019.

[Yao *et al.*, 2019a] Yazhou Yao, Fumin Shen, and et al. Extracting privileged information for enhancing classifier learning. *IEEE Transactions on Image Processing*, 28:436–450, 2019.

[Yao *et al.*, 2019b] Yazhou Yao, Jian Zhang, and et al. Towards automatic construction of diverse, high-quality image dataset. *IEEE Transactions on Knowledge and Data Engineering*, 2019.

[Yap *et al.*, 2012] Ghim-Eng Yap, Xiao-Li Li, and Philip Yu. Effective next-items recommendation via personalized sequential pattern mining. In *Database Systems for Advanced Applications*, pages 48–64, 2012.

[Zhang *et al.*, 2016] Xiao Zhang, Changlin Mei, Degang Chen, and Jinhai Li. Feature selection in mixed data: A method using a novel fuzzy rough set-based information entropy. *Pattern Recognition*, 56:1–15, 2016.