

# Intention Nets: Psychology-Inspired User Choice Behavior Modeling for Next-Basket Prediction

Shoujin Wang,<sup>1</sup> Liang Hu,<sup>2,3,\*</sup> Yan Wang,<sup>1,\*</sup> Quan Z. Sheng,<sup>1</sup> Mehmet Orgun,<sup>1</sup> Longbing Cao<sup>2</sup>

<sup>1</sup>Department of Computing, Macquarie University

<sup>2</sup>Advanced Analytics Institute, University of Technology Sydney

<sup>3</sup>University of Shanghai for Science and Technology

{shoujin.wang, yan.wang}@mq.edu.au, rainmilk@gmail.com

## Abstract

Human behaviors are complex, which are often observed as a sequence of heterogeneous actions. In this paper, we take user choices for shopping baskets as a typical case to study the complexity of user behaviors. Most of existing approaches often model user behaviors in a mechanical way, namely treating a user action sequence as homogeneous sequential data, such as hourly temperatures, which fails to consider the complexity in user behaviors. In fact, users' choices are driven by certain underlying intentions (e.g., feeding the baby or relieving pain) according to Psychological theories. Moreover, the durations of intentions to drive user actions are quite different; some of them may be persistent while others may be transient. According to Psychological theories, we develop a hierarchical framework to describe the goal, intentions and action sequences, based on which, we design *Intention Nets (IntNet)*. In IntNet, multiple *Action Chain Nets* are constructed to model the user actions driven by different intentions, and a specially designed *Persistent-Transient Intention Unit* models the different intention durations. We apply the IntNet to next-basket prediction, a recent challenging task in recommender systems. Extensive experiments on real-world datasets show the superiority of our Psychology-inspired model IntNet over the state-of-the-art approaches.

## Introduction

Human behaviors are full of complexity and uncertainty (Cao 2015), and psychologists have been trying to understand the influential factors in the process of forming *intentions* and performing actions (Fishbein et al. 2000). For example, shopping a basket of products is a typical user behavior in daily life. In this paper, we take the next-basket prediction as a representative problem to study the complexity of user behavior, where the sequences of user choices in sequences of baskets correspond to user action sequences.

According to the psychological theory (Albarracin and Wyer Jr 2000), user actions are driven by a set of heterogeneous intentions (e.g., feeding the baby and relieving pain in basket  $b_2$  shown in Figure 1) to achieve certain goals. Moreover, the durations of intentions may be quite different, i.e.,

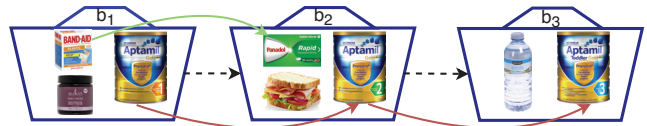


Figure 1: An example of a sequence of three baskets ( $b_1, b_2, b_3$ ). Traditional methods simply model the transitions between baskets (dashed arrow lines) while Psychological research indicates user choices in baskets are driven by different intentions accomplished by different action chains (arrow lines of different colours).

some of them may drive a long persistent action chain containing more actions while others may drive a short one with fewer actions. For example, as shown in Figure 1, baby feeding is a persistent intention which drives a long action chain (red lines) to buy milk powder of different stages across the whole basket sequence, while relieving pain is a transient intention driving a short one. Furthermore, the psychological research (Carrera et al. 2012) has indicated that user behavior can be described as a *hierarchical framework*, where the goal at the top level (e.g., filling shopping baskets) is driven by multiple underlying intentions (e.g., feeding baby and relieving pain) and each intention results in a chain of relevant choices across baskets with different durations.

However, most of existing next-basket prediction methods (Quadrana et al. 2017; Guidotti et al. 2018) mechanically model a user action sequence as homogeneous data points in time series just like the hourly temperatures, without considering human's heterogeneous intentions behind. For instance, the Markov chain-based (Rendle and et al. 2010) and the recurrent neural networks (RNN) (Wan et al. 2015; Yu et al. 2016) based methods are two current representative approaches to model user basket sequences in this way. Such mechanical treatment does not reflect the complexity and heterogeneity of human behaviors in the decision-making process (Chen, Lu, and Wang 2017; Hu et al. 2017). As a result, these approaches have trouble in creating an in-depth understanding of behaviors (including their heterogeneity and intentions) and properly predicting a user's next-basket choices (Cao 2010; Wang, Hu, and Cao 2017).

\*Corresponding author

Copyright © 2020, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

In this paper, inspired by the psychological perspective on user behavior formation and development, we propose Intention Nets (IntNet) to model the complex and heterogeneous user behaviour sequences based on the aforementioned hierarchical framework for next-basket prediction. First, Intention Recognition Nets (IRN) are proposed as a component of IntNet to disentangle the potential intentions from the observed user choices on items in each basket. Hence, the choices (chosen items) in each basket are integrated as one intention-driven action according to the disentangled intentions. A sequence of actions for the same intention from multiple baskets form an action chain, and finally multiple action chains are built due to the multiple intentions behind the baskets. Then, the Parallel Action Chain Nets (PACN) are designed to capture the transitions inside action chains where each Action Chain Net (ACN) models one intention accordingly. Note that different intentions may hold for different influence durations in practice. Accordingly, we specially design the Persistent-Transient Intention Units (PTIUs) to equip each ACN with precisely modeling the dependency between actions according to the duration of each intention. With this special structure inside, a PTIU is able to simultaneously model the persistent intention states that require more actions to accomplish and transit intention states that involve only a few actions. Consequently, based on the current state of each intention, ACN predicts the next action required to accomplish the corresponding intention, i.e., ranking the candidate items for the next choice. As a result, the predicted items for all intentions together accomplish the goal of next basket choices. The main contributions of this work are summarized below:

- We propose a Psychology-inspired hierarchical framework to model the complexity and heterogeneity of user behaviors and their driving intentions.
- With the hierarchical framework, we design the Intention Nets (IntNet) with two main components: the Intention Recognition Nets (IRN) to disentangle heterogeneous intentions, and the Parallel Action Chain Nets (PACN) to model the transition of actions for each intention.
- IntNet is illustrated by addressing the next-basket prediction problem in terms of modeling the sequence of purchasing actions and their intention dynamics.

We conduct extensive experiments for next-basket prediction to verify the effectiveness of IntNet. The results on two real-world datasets show that IntNet outperforms the best baseline by 3% to 19.89% w.r.t. the prediction accuracy.

## Related Work

Existing work on next-basket prediction can be generally categorized into sequence model-based approaches and neural model-based approaches (Wang et al. 2019b).

Sequential patterns are widely used to capture the sequential dependencies between baskets. (Guidotti et al. 2018) proposed Temporal Annotated Recurring Sequence to simultaneously capture different factors (e.g., co-occurrence and sequentiality) that influence user choices for personalizing next-basket predictions. Pattern-mining-based approaches may greatly reduce the recommendation accuracy

and diversity by only recommending those frequent items while ignoring less frequent ones. Markov chain models are another solution to next-basket prediction. (Rendle and et al. 2010) proposed a Factorized Personalized Markov Chains (FPMC) model to factorize the transition matrix over underlying Markov chains on items from adjacent baskets to model sequential behaviour for next-basket recommendation. However, FPMC is a first-order MC model which only captures the first-order dependencies while ignoring higher-order ones, leading to poor recommendation performance (Hu et al. 2018; Wang and et al. 2019; Yao et al. 2018).

Recent years have witnessed the potential of neural models in modeling complex dependencies. (Wang et al. 2015) built a hierarchical representation model (HRM) based on shallow neural networks to construct a hybrid representation of the last basket to predict the next basket. In this way, HRM only captures the first-order dependencies between two adjacent baskets. Compared with shallow neural models, deep neural models like RNN are more powerful in modeling complex relations due to their complex architectures. (Yu et al. 2016) proposed a Dynamic REcurrent bASket Model (DREAM) to both learn a dynamic representation of a user and capture global sequential features among baskets for next-basket recommendations. (Le and Lauw 2019) developed a sequence encoder built on RNN while incorporating the intra-basket correlations for more coherent basket recommendations. (Bai et al. 2018) incorporated the attention mechanism into RNN to emphasize those more relevant items for next-basket prediction. Although RNN-based approaches can capture higher-order dependencies among multiple baskets, they may generate false dependencies due to the employed rigid order assumption among any two adjacent baskets and bias to the recent baskets due to the memory decay (Wang et al. 2019a).

To sum up, all the aforementioned work mechanically models user behaviour as homogeneous sequences without thoroughly considering the complex and heterogeneous human intentions behind. The lack of understanding of human intentions would significantly reduce the next-basket prediction performance. Actually, the theories of reasoned action and planned behavior (Fishbein et al. 2000; Carrera et al. 2012) in Psychology have revealed that intentions are the driving factors of actions and one intention requires a series of relevant actions to accomplish. Moreover, the current intention state leads to the subsequent actions (Malle and Knobe 1997). Inspired by these theories, we propose the Intention Nets to deeply model the complex and heterogeneous user behaviour for next-basket prediction by tracking the intention states behind user actions.

## Problem Statement

Given a user transaction dataset,  $D = \{s_1, \dots, s_{|D|}\}$  denotes a set of sequences of shopping baskets (called baskets for short), where each sequence  $s = \{b_1, \dots, b_{|s|}\} (s \in D)$  consists of a sequence of baskets associated with a user. Here  $|D|$  denotes the total number of sequences in  $D$  while the subscripts of  $s$  indicate the order of baskets. Each basket  $b = \{v_1, \dots, v_{|b|}\} (b \in s)$  contains a collection of items

in one transaction event. It should be noted that the subscripts in  $b$  do not rigidly indicate the order of items occurring in the basket since the order in which items are put into a basket are usually random and does not make much sense in real-world cases (Wang et al. 2018). All the items occurring in all baskets constitute the universal item set  $V = \{v_1, \dots, v_{|V|}\}$ . For a target basket  $b_t (b_t \in s)$ , all the baskets that occurred prior to  $b_t$  in  $s$  together constitute the sequential context (called context for short) of  $b_t$  over  $s$ , denoted as  $C_{b_t} = \{b_1, \dots, b_{t-1}\}$  where each basket  $b \in C_{b_t}$  is a contextual basket. Accordingly, each item  $v \in b (b \in C_{b_t})$  in the contextual basket is a contextual item. Given a context  $C$  with precedent  $(t-1)$  baskets, a next-basket predictor aims to predict user choices for  $t^{\text{th}}$  basket, namely to generate a list of items that are most probably to appear in basket  $b_t$ .

## Intention Nets

The architecture of IntNet is illustrated in Figure 2 (a). IntNet is mainly composed of two components: (1) Intention Recognition Nets (IRN), and (2) Parallel Action Chain Nets (PACN). IRNs first detects the intention behind each chosen item in a basket, and then builds an action representation with the embeddings of items associated with a specific intention. This action embedding is then used as an input of the current time step into the corresponding ACN. Each ACN consists of Persistent-Transient Intention Units (PTIUs) associated with the action embedding of each time step to model their transitions and durations. The last intention state  $\mathbf{h}_{t-1,l}$  output from PTIUs at step  $(t-1)$  is used as the intention-specific context embedding  $\mathbf{c}_l$  of the  $l^{\text{th}}$  intention. Then,  $\mathbf{c}_l$  is used to predict the intention-specific items in the next basket. Finally, the predicted items for each intention are assembled as the next basket. Next, we demonstrate the technical details of these two components.

### Intention Recognition Nets

Given the items in a basket, IRNs aim to first detect the underlying intention driving a user to choose an item  $v_i$ . To be specific, each item  $v$  is projected 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, the embedding  $\mathbf{v}_i$  of item  $v_i$  is input into IRNs.  $\mathbf{W}_p \in \mathbb{R}^{K \times m}$  is the intention filtering matrix, where  $m$  is the number of possible latent intentions (the optimal number of intentions is tuned by cross validation). To determine the intention behind each user choice, we employ a softmax function to compute the probability associated with each intention. Specifically, the probability  $g_{i,k}$  of item  $v_i$  w.r.t. the  $k^{\text{th}}$  intention is given below:

$$g_{i,k} = \frac{\exp(\mathbf{v}_i^\top \mathbf{W}_p[:,k])}{\sum_{h=1}^m \exp(\mathbf{v}_i^\top \mathbf{W}_p[:,h])}, \quad k \in \{1, \dots, m\} \quad (1)$$

Further, we adopt the gumble-softmax trick (Jang, Gu, and Poole 2017) to assign each item with a specific intention based on the above probabilities. Formally,

$$y_{i,k} = \frac{\exp((\log(g_{i,k}) + \pi_k)/\tau)}{\sum_{h=1}^m \exp((\log(g_{i,h}) + \pi_h)/\tau)}, \quad k \in \{1, \dots, m\} \quad (2)$$

where  $\pi_1 \dots \pi_m$  are the corresponding Gumbel noises which are independent and identically distributed samples drawn from Gumbel  $(0, 1)$ <sup>1</sup> and  $\tau$  is the temperature parameter (Jang, Gu, and Poole 2017). When  $\tau \rightarrow 0^+$ , the vector  $\mathbf{y}_i = [y_{i,1} \dots y_{i,m}]$  approximates a one-hot vector where only one dimension has the value 1, i.e. the item  $v_i$  concentrates on a single intention. In this paper,  $\tau$  is empirically set to be 0.01 to achieve the best performance.

Then, the action embedding w.r.t. the  $k^{\text{th}}$  intention can be obtained by aggregating the embeddings of all items in the current basket according to their corresponding intention assignment vectors  $\mathbf{y}_i$ :

$$\mathbf{e}_{j,k} = \sum_{i=1}^{|b_j|} y_{i,k} \mathbf{v}_i \quad (3)$$

Therefore, we obtain an action embedding for each intention as the input of each time step for the Action Chain Nets.

### Parallel Action Chain Nets

The parallel action chain nets (PACN) consists of  $m$  action chain nets (ACNs), and one for each intention as shown in Figure 2 (a). Each ACN is composed of  $(t-1)$  sequentially connected Persistent-Transient Intention Units (PTIUs) to model the sequential dependencies over the intention-driven actions, i.e., choices, in the  $(t-1)$  contextual baskets. As stated before, the durations of different intentions may be quite different. The design of PTIUs aims to address this problem in an adaptive way, i.e., PTIUs are capable of learning the durations of heterogeneous intentions from data.

**Persistent-Transient Intention Units.** Due to the heterogeneity of intentions, some intentions are observed as persistent action chains while others are observed as transient ones. Traditional RNN cells like Long Short-Term Memory (LSTM) (Xingjian et al. 2015) or Gated Recurrent Unit (GRU) (Chung et al. 2014) are incapable of representing the heterogeneous durations of intention-driven actions due to the identical state updating operation at each time step. To this end, inspired by the great power of ordered neurons LSTM (Shen et al. 2019) in modeling hierarchical semantic meaning of sentences and paragraphs, we design the PTIUs to serve as the cells of each ACN. Specially, PTIUs introduce an intention duration detection module to weigh the persistence and transience of an intention and apply different update strategies to update the intention states accordingly.

The specific structure of a PTIU is depicted in Figure 2 (b), where a persistent intention gate and a transient intention gate are used to softly specify the duration of an action. Correspondingly, a persistent-intention updating module and a transient-intention updating module are applied to update the intention states in different strategies. Below, we formulate the state updating details.

First, the current intention state  $\tilde{\mathbf{h}}_j$  in preparation for the subsequent state updating is calculated below:

$$\mathbf{r}_j = \sigma_s(\mathbf{W}_r[\mathbf{h}_{j-1}, \mathbf{e}_j] + \mathbf{b}_r) \quad (4)$$

<sup>1</sup>Gumbel (0,1) distribution is sampled using inverse transform sampling by drawing  $u \sim U(0, 1)$  and computing  $\pi = -\log(-\log(u))$ .

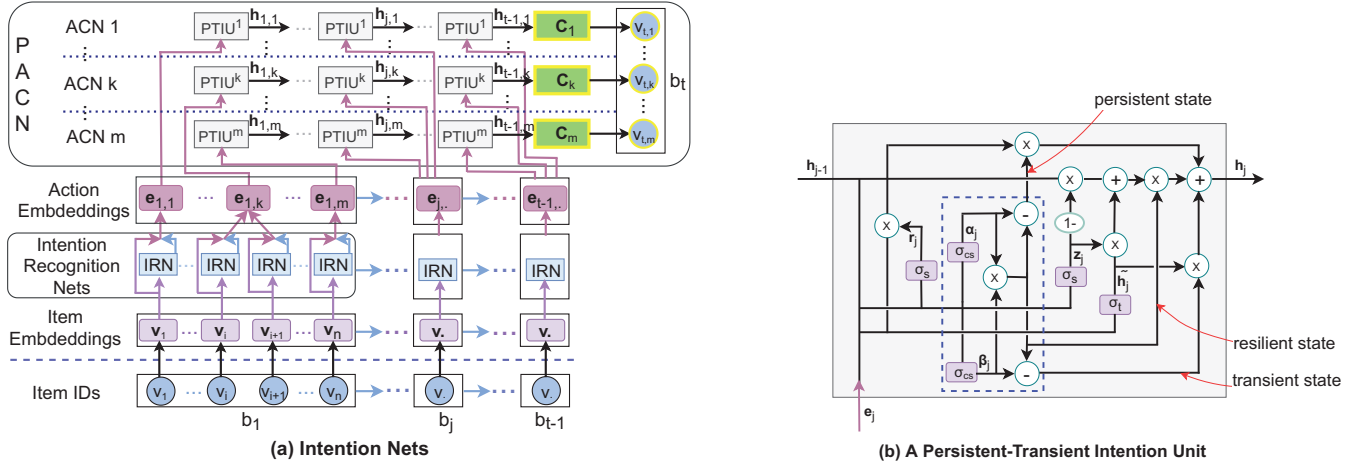


Figure 2: (a) The Intention Nets model consist of two main components: Intention Recognition Nets and Parallel Action Chain Nets (PACN); (b) The Persistent-Transient Intention Unit (PTIU) introduces a persistent gate and transient gate (see the blue dash line square) to respectively determine the persistent part and transient part of the intention state and update them accordingly.

$$z_j = \sigma_s(\mathbf{W}_z[\mathbf{h}_{j-1}, \mathbf{e}_j] + \mathbf{b}_z) \quad (5)$$

$$\tilde{\mathbf{h}}_j = \sigma_t(\mathbf{W}_h[\mathbf{r}_j * \mathbf{h}_{j-1}, \mathbf{e}_j] + \mathbf{b}_h) \quad (6)$$

where  $\mathbf{r}_j$  and  $\mathbf{z}_j$  are the vectors of reset gate and update gate respectively.  $\sigma_s$  and  $\sigma_t$  are activation functions and are specified as sigmoid and tanh respectively in this work.  $\mathbf{W}_r$ ,  $\mathbf{W}_z$  and  $\mathbf{W}_h$  are the weight matrices while  $b_r$ ,  $b_z$  and  $b_h$  are the corresponding biases.

Then, the persistent intention gate and the transient intention gate are designed to weigh the persistence and transience according to the intention state:

$$\alpha_j = \sigma_{cs}(\mathbf{W}_\alpha[\mathbf{h}_{j-1}, \mathbf{e}_j] + \mathbf{b}_\alpha) \quad (7)$$

$$\beta_j = \mathbf{1} - \sigma_{cs}(\mathbf{W}_\beta[\mathbf{h}_{j-1}, \mathbf{e}_j] + \mathbf{b}_\beta) \quad (8)$$

where  $\sigma_{cs}$  is the activation function, which is specified to cumulative softmax (Shen et al. 2019) to predict the cut-off points of persistent and transient intentions in the state vector  $\mathbf{h}_{j-1}$  considering the cumulative effect of each dimension in  $\mathbf{h}_{j-1}$ . To be specific, first a softmax function is used to transfer each dimension of the input vector into a probability value and then cumulative sum (cumsum) is performed on the output vector of the softmax function.

$$\text{cumsum}(x_1, x_2, \dots, x_n) = [x_1, x_1 + x_2, \dots, x_1 + x_2 + \dots + x_n] \quad (9)$$

This is based on the assumption that the information for persistent intention is encoded in a particular part with some adjacent dimensions of  $\mathbf{h}_{j-1}$  and will be passed to the similar dimensions of the next states continuously to play long-term impact. In contrast, the information for transient intention is stored in a different part with other adjacent dimensions of  $\mathbf{h}_{j-1}$  and will be updated by the current intention in the current state to reduce its impact on the future choices of items.

Moreover, we introduce the resilient gate as an adaptive tradeoff between persistence and transience:

$$\gamma_j = \alpha_j \circ \beta_j \quad (10)$$

where  $\circ$  denotes the element-wise multiplication.

Therefore, the vector of last intention state  $\mathbf{h}_{j-1}$  consists of three parts: the persistent state encoding part  $(\alpha_j - \gamma_j)$ , the resilient state encoding part  $\gamma_j$ , and the transient state encoding part  $(\beta_j - \gamma_j)$  as illustrated in 2 (b). As a result, PTIUs respectively update the information from the last state to the current state with three different strategies: (1) the states associated with the persistent state encoding part are completely copied from the last states to the current ones to maintain the persistent impact; (2) the states corresponding to the transient part are completely updated with the information from the current action; and (3) the resilient states combine the information from the last state and current input as a tradeoff. Following these strategies, we update intention state  $\mathbf{h}_j$  of the current time step by taking the last state  $\mathbf{h}_{j-1}$  and the current candidate state  $\tilde{\mathbf{h}}_j$  as the input:

$$\mathbf{h}_j = (\alpha_j - \gamma_j) \circ \mathbf{h}_{j-1} + \gamma_j \circ ((\mathbf{1} - \mathbf{z}_j) \circ \mathbf{h}_{j-1} + \mathbf{z}_j \circ \tilde{\mathbf{h}}_j) + (\beta_j - \gamma_j) \circ \tilde{\mathbf{h}}_j \quad (11)$$

where  $\mathbf{z}_j$  and  $\tilde{\mathbf{h}}_j$  are the reset gate and candidate intention state at the current time step calculated in Eqs. 5 and 6 respectively.

Thanks to the special design of PTIU, which enables to effectively model the different state transitions for different intentions with heterogeneous durations.

### Next-basket Prediction

Once the sequence of the  $(t-1)$  contextual baskets has been input into the IntNet, the last intention states are used as the contextual embedding to predict the next intention-specific choices for filling the next-basket. We concatenate the contextual embedding vectors for all intentions to form a contextual embedding matrix for all intentions:

$$\mathbf{C} = [\mathbf{h}_{t-1,1}, \mathbf{h}_{t-1,2} \dots \mathbf{h}_{t-1,m}] \quad (12)$$

---

**Algorithm 1** Model parameter learning procedure

---

- 1:  $\mathcal{B} \leftarrow$  Get mini-batch from all context-target basket pairs
  - 2:  $\mathcal{N} \leftarrow$  Sample a set of negative items  $V_{v_t}^-$ , for each item  $v_t$  from each target basket  $v_t \in b_t (b_t \in \mathcal{B})$
  - 3: Compute mini-batch loss using Eq. 16:  
 $\mathcal{L}_{\mathcal{B}} \leftarrow \frac{1}{|\mathcal{B}|} \sum_{\langle v^+, V^- \rangle \in \langle \mathcal{B}, \mathcal{N} \rangle} \mathcal{L}(v^+, V^-)$
  - 4: Update parameters:  $\Theta \leftarrow \Theta - \Gamma_{Adam}(\nabla_{\Theta} \mathcal{L}_{\mathcal{B}})$
- 

Given a candidate item  $v_t$ , we extract its specific contextual embedding from the above contextual embedding matrix according to the intention inferred for  $v_t$ :

$$\mathbf{C}_{\mathbf{v}_t} = \mathbf{y}_t^{\top} \mathbf{C}^{\top} \quad (13)$$

where  $\mathbf{y}_t$  is the intention vector of item  $v_t$  and each of its elements is calculated via Eq. 2. Suppose that a candidate item  $v_t$  is associated with the  $1^{st}$  intention, and its corresponding intention vector  $\mathbf{y}_t = [1, 0, 0 \dots 0]$ . Namely, it will be treated as the  $1^{st}$  intention-driven choice for the next basket.

Then the inner product is applied as a score to quantify the relevance degree between the candidate item  $v_t$  and the given context  $C$ :

$$\delta(v_t, C) = \mathbf{C}_{\mathbf{v}_t} \mathbf{v}_t \quad (14)$$

Accordingly, the conditional probability of a user to choose the item  $v_t$  under the context  $C$  is obtained according to the above score in terms of logistic function:

$$p(\hat{v}_t | C; \Theta) = \frac{1}{1 + e^{-\delta(v_t, C)}} \quad (15)$$

where  $\Theta$  is the set of model parameters to be learned over all the sequences of baskets in a dataset.

Finally, without loss of generality (Wan et al. 2018) and according to Eq. 15, the top- $n$  items with the highest probabilities are selected to fill the next basket.

### Loss Function and Parameter Learning

Taking the negative log over the conditional probability  $p(v|C)$  (Eq. 15), we obtain the loss function:

$$\mathcal{L}(v^+, V^-) = -[\log(p(v^+|C)) + \sum_{v^- \in V^-} \log(1 - p(v^-|C))] \quad (16)$$

where  $v^+$  and  $V^-$  are the positive and negative sample sets for each item in the next basket. Here, we adopt the negative sampling strategy (Goldberg and Levy 2014) due to the huge number of items. Given context  $C$  and the corresponding true next basket  $b_t$  including  $|b_t|$  chosen items, we build  $|b_t|$  contrastive pairs for training the model. Each contrastive pair takes one item from  $b_t$  as the positive sample  $v^+$  while randomly sample  $|V^-|$  negative items from the item set  $V \setminus b_t$  to form the negative sample set  $V^-$ . The loss of each contrastive pair  $\langle v^+, V^- \rangle$  is composed of the loss  $-\log(p(v^+|C))$  of one positive sample and the loss  $-\log(1 - p(v^-|C))$  for each negative sample. The model parameters are updated by minimizing the loss  $\mathcal{L}(v^+, V^-)$  of all contrastive pairs in one mini-batch each time.

Our model is implemented using Tensorflow 1.4. We only present a brief scheme of the learning procedure on a mini-batch in Algorithm 1 due to the limited space. We use Adam (Kingma and Ba 2015) for gradient learning, as illustrated by  $\Gamma_{Adam}$  in Algorithm 1. The initial learning rate is empirically set to 0.001 and the batch size is set to 50.

## Experiments and Evaluation

### Data Preparation

Two real-world online transactional datasets are used for the experiments: (1) Tmall<sup>2</sup> released by IJCAI-15 competition, which recorded the purchased baskets from each anonymous user on Tmall platform (The Chinese version of Amazon) in six months. Each basket is associated with a purchase date with no timestamp for each item inside transaction; and (2) Tafeng<sup>3</sup> released on Kaggle, which contains the transactional data of a Chinese grocery store generated in four months, whose format is similar to Tmall. Both datasets are commonly used to test the performance of next-basket prediction (Yu et al. 2016; Guidotti et al. 2018).

First, a set of sequences is extracted from each dataset where each sequence consists of all the baskets purchased by a user. The baskets inside each sequence are ordered by the purchasing time. To feed the data into the model well, we follow a common manner to build sequence instance sets for training and test where each instance is in the form of  $\langle C, b_t \rangle$  ( $C = \{b_1, b_2, \dots, b_{t-1}\}$ ),  $C$  and  $b_t$  indicate context and target baskets respectively. Specifically, a fixed sequence length  $t$  (i.e., 5 and 8 for Tmall and Tafeng respectively) is set for each dataset according to the data characteristics. Sequence instances of length- $t$  are extracted from the original sequences by employing the sliding window technique (Tanbeer et al. 2009) on original sequences longer than  $t$  while padding and masking (Collins et al. 2012) on those shorter than  $t$  respectively. Second, we make three training-test splits on the sequence instance set by randomly selecting 20%, 30% and 40% of the instances whose target basket happens in the last 30 days respectively for test while others for training. The characteristics of experimental datasets are shown in Table 2. Our method consistently outperforms all the baselines on all proportions, and only the results w.r.t. the 30% split are reported due to the limited space.

### Experimental Settings

**Evaluation Metrics.** Four commonly used accuracy metrics are employed to evaluate the recommendation performance of our method and the baselines. They are Recall, F-1 Score, Hit-Ratio (HR) and normalized Discounted Cumulative Gain (nDCG) (Liu et al. 2018; Yang et al. 2018).

**Comparison Methods.** Besides **IntNet**, **IntNet-S** is a simplified version including a single ACN and is built on the homogeneous intention assumption. It is aimed to demonstrate the effectiveness of PACN of IntNet in handling different heterogeneous intentions. In addition, **IntNet-GRU**

<sup>2</sup><https://tianchi.aliyun.com/dataset/dataDetail?dataId=42>

<sup>3</sup><https://www.kaggle.com/chiranjivdas09/ta-feng-grocery-dataset>

Table 1: Prediction accuracy on two real-world datasets

	Tmall						Tafeng					
	F1@5	F1@20	HR@5	HR@20	nDCG@5	nDCG@20	F1@5	F1@20	HR@5	HR@20	nDCG@5	nDCG@20
TBP	0.0282	0.0312	0.0210	0.0488	0.0642	0.1002	0.0252	0.0310	0.0187	0.0379	0.0842	0.1000
FPMC	0.0614	0.0538	0.0876	0.2068	0.0645	0.1088	0.0618	0.0565	0.0668	0.1468	0.0564	0.1012
HRM	0.0848	0.0788	0.1010	0.2322	0.0854	0.1362	0.0847	0.0785	0.0800	0.1788	0.0786	0.1326
DERAM	0.1080	0.0816	0.1226	0.2551	0.1028	0.1600	0.1038	0.0824	0.0906	0.1956	0.1312	0.1628
NAM	0.1088	0.0819	0.1224	0.2498	0.1148	0.1715	0.1108	0.0736	0.0978	0.1739	0.1301	0.1507
Beacon	0.1200	0.0880	0.1268	0.2746	0.1262	0.1876	0.1100	0.0897	0.0988	0.2012	0.1304	0.1626
MCPRN	0.1202	0.0882	0.1224	0.2755	0.1267	0.1892	0.1090	0.0897	0.1014	0.2137	0.1328	0.1669
IntNet-S	0.1182	0.0787	0.1309	0.2449	0.1237	0.1613	0.1102	0.0810	0.0942	0.1920	0.1315	0.1623
IntNet-GRU	0.1202	0.0880	0.1220	0.2752	0.1263	0.1888	0.1104	0.0824	0.0946	0.1956	0.1316	0.1625
IntNet	<b>0.1238</b>	<b>0.0934</b>	<b>0.1378</b>	<b>0.2857</b>	<b>0.1416</b>	<b>0.2018</b>	<b>0.1161</b>	<b>0.0971</b>	<b>0.1078</b>	<b>0.2283</b>	<b>0.1395</b>	<b>0.2001</b>
Improve (%) <sup>4</sup>	3.00	5.87	5.30	3.70	11.76	6.66	4.78	8.25	6.31	6.83	5.05	19.89

Table 2: Statistics of datasets

Statistics	Tmall	Tafeng
#Sequences	135,014	10,453
#Baskets	399,008	60,392
#Items	98,727	11,207
Avg. sequence length	2.96	5.78
Avg. basket size	3.08	8.42

is the version replacing the PTIUs in IntNet with general GRUs, which is compared with IntNet to justify the efficacy of PTIUs in handling intentions with different durations. The following representative and state-of-the-art approaches built on various models from sequential patterns to mixture models are selected as baselines.

- **TBP**: a next-basket predictor based on the temporal annotated recurring sequence to capture different factors influencing the user decision process (Guidotti et al. 2018).
- **FPMC**: a Markov chain-based approach which factorizes the transition matrix between items from adjacent baskets for next-basket prediction (Rendle and et al. 2010).
- **HRM**: a hierarchical representation model to predict the next basket based on the representations of a user and his/her last basket (Wang et al. 2015).
- **DERAM**: an RNN-based model which learns a dynamic representation of a user based on the historical baskets to predict the next basket (Yu et al. 2016).
- **NAM**: a model which incorporates the attention mechanism into RNN to track a user’s evolving appetite for items for next-basket prediction (Bai et al. 2018).
- **Beacon**: a state-of-the-art next-basket predictor using RNN to encode the basket sequence while incorporating the intra-basket correlations (Le and Lauw 2019).
- **MCPRN**: a next-item recommender using multiple channels to model different types of items in a basket (Wang et al. 2019a). We modified it for next-basket prediction.

<sup>4</sup>Improvement over the best performance achieved on all compared methods.

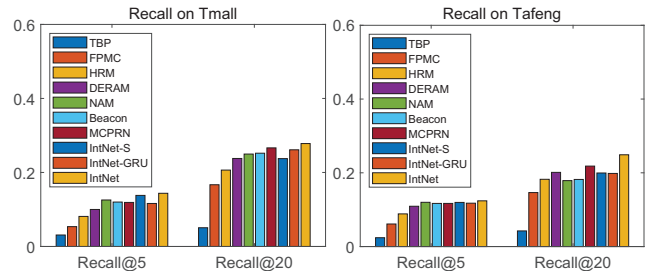


Figure 3: Recalls of Intention Nets and other compared methods.

**Parameter Settings.** For a fair comparison, we initialize each baseline model with the parameter settings in the original papers and then tune them on our datasets for best performance. In our model, the dimensions of item embeddings and intention states are empirically set to 100. The number of channels  $m$  is set to 3 by tuning on the validation set.

### Performance Evaluation

We conduct extensive experiments to evaluate our model by answering the following questions:

**Q1:** How does our model perform compared with the baseline approaches in terms of prediction accuracy?

**Q2:** How does PACN for modeling multiple heterogeneous intentions perform compared with a single ACN for modeling only a homogeneous intention?

**Q3:** How does the PTIU perform in handling intentions with different durations?

**Reply to Q1: IntNet vs. Baselines.** We compare the prediction accuracy of IntNet with those of the seven baselines and present the results in Table 1. In TBP, we set the the minimum item occurrence times in the whole dataset to 5 and the minimal number of baskets per user to 10. TBP is based on frequent-pattern mining which is usually biased to frequent items. Moreover, TBP can only handle users with sufficient data due to its user-centralized design, performing the worst. The number of factors is set to 50 in FPMC for the best performance. Compared with TBP, FPMC can cover much more items and thus can achieve better performance. However, FPMC assumes strict first-order dependencies be-

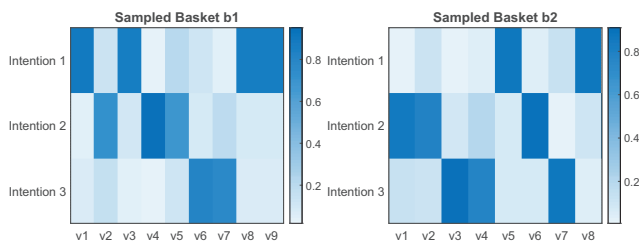


Figure 4: Intention assignment visualization of the items in two sampled baskets on Tafeng.

tween any adjacent baskets, which may not be always true in the real life. The sizes of item embedding and state vectors (if there is any) are empirically set to 100 in the other five baselines for best performance. HRM relaxes the strict assumption in FPMC by learning latent representations of both items and baskets for better performance. Both FPMC and HRM capture the first-order dependencies between adjacent baskets while ignoring higher-order dependencies, leading to limited accuracy. DERAM employs an RNN to model the higher-order sequential dependencies among multiple baskets and thus generates more accurate prediction. NAM incorporates the attention mechanism into RNN to emphasize those more relevant items and baskets for better prediction while Beacon incorporates intra-basket correlations into RNN for coherent basket prediction. MCPRN employs multiple channels to model dependencies among different types of items independently to avoid the noisy information from irrelevant items. However, it still cannot handle the complex behaviours driven by heterogeneous intentions holding different durations due to its oversimplified PSRU cells.

In contrast, by modeling the transitions of actions for different intentions with different ACNs and carefully updating different intention states respectively in each PTIU, IntNet can well model the complex and heterogeneous user behaviours for better next-basket prediction. Consequently, it achieves 3% to 19.89% improvements over the best-performing baseline w.r.t. all the metrics on both datasets (cf. the bottom row of Table 1). The recall also shows IntNet leads the baselines with a clear margin (cf. Figure 3).

**Reply to Q2: Heterogeneous Intention Modeling vs. Homogeneous Intention Modeling.** To demonstrate the efficacy of modeling multiple heterogeneous intentions with PACN, we compare the performance of IntNet with that of IntNet-S. It is clear that IntNet achieves much higher accuracy than IntNet-S as shown in Table 1 and Figure 3. Particularly, IntNet achieves at least 15% improvement on most metrics like F1@20, HR@20, nDCG@20 and Recall@20 on both datasets, which proves that user choices are driven by different heterogeneous intentions instead of a single homogeneous one assumed by most of the existing approaches.

**Reply to Q3: PTIU vs. GRU in Modeling Intentions with Different Durations.** To demonstrate the effectiveness of our PTIU cells equipped in IntNet in modeling intentions with different durations, we compare IntNet with IntNet-GRU. It is clear that IntNet performs much better in terms of

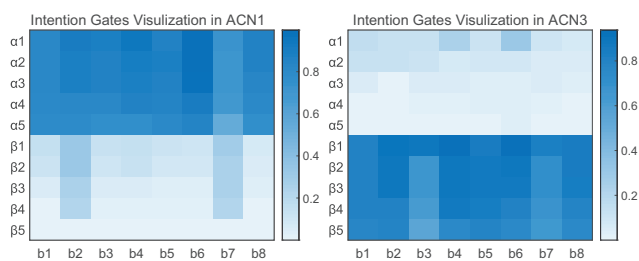


Figure 5: Persistent (denoted as  $\alpha$ ) and transient (denoted as  $\beta$ ) intention gate values of a sampled sequence of 8 baskets from Tafeng. Top-5 dimensions (e.g.,  $\alpha_1, \dots, \alpha_5$ ) with the largest values of each gate are shown for a clear view (a color version works better).

all four accuracy metrics on both datasets. Particularly, the HR@20 and nDCG@20 of IntNet are at least 10% higher than that of IntNet-GRU on Tafeng. This justifies the obvious advantages of PTIUs over GRUs in handling intentions with different durations.

### Visualization of the IntNet Working Mechanism

To interpret the working mechanism of IntNet, we sample baskets from Tafeng and visualize the intention assignments in terms of  $\mathbf{y}$  (cf. Eq. 2) and intention durations in terms of persistent-transient gates  $\alpha$  and  $\beta$  (cf. Eqs. 7 and 8).

**Intention Assignments.** Figure 4 describes the intention assignment of sampled baskets  $b_1$  and  $b_2$ . It is clear that the choices on different items (e.g.,  $v_1, v_2, v_6$ ) in one basket (e.g.,  $b_1$ ) are commonly driven by different intentions (e.g., intentions 1, 2 and 3) and most of the choices concentrate on one intention indicated by the darkest color. This essentially justifies the necessity and rationality of the parallel-network structure inside PACN in IntNet.

**Intention Gates.** Figure 5 depicts the intention gate values in two ACNs (ACN1 and ACN3) of a sampled sequence composed of 8 baskets. It is obvious that the persistent gate stably has much larger values than transient gate in ACN1 (the left sub figure) while the case is reversed in ACN3 (the right sub figure). This indicates that ACN1 mainly models the action transitions for persistent intentions while ACN3 mainly models those for transient intentions behind the sampled sequence, proving the efficacy of the designed PTIU to capture heterogeneous intentions with different durations.

### Conclusions

In this paper, we have proposed Intention Nets (IntNet) to effectively model the complex and heterogeneous user behaviours for accurate next-basket prediction, which cannot be well addressed by existing next-basket prediction works. IntNet utilizes intention recognition nets to disentangle the potential intentions from the observed user choices in baskets and parallel action chain nets to model the transitions inside each action chain to accomplish an intention in parallel. The empirical evaluation on the real-world datasets shows its superiority over the state-of-the-art approaches.

## Acknowledgements

This work was supported by Australian Research Council Discovery Project DP180102378.

## References

- Albarracín, D., and Wyer Jr, R. S. 2000. The cognitive impact of past behavior: influences on beliefs, attitudes, and future behavioral decisions. *Journal of Personality and Social Psychology* 79(1):5.
- Bai, T.; Nie, J.-Y.; Zhao, W. X.; Zhu, Y.; Du, P.; and Wen, J.-R. 2018. An attribute-aware neural attentive model for next basket recommendation. In *SIGIR*, 1201–1204.
- Cao, L. 2010. In-depth behavior understanding and use: the behavior informatics approach. *Information Sciences* 180(17):3067–3085.
- Cao, L. 2015. Coupling learning of complex interactions. *Information Processing & Management* 51(2):167–186.
- Carrera, P.; Muñoz, D.; Caballero, A.; Fernández, I.; and Albaracín, D. 2012. The present projects past behavior into the future while the past projects attitudes into the future: How verb tense moderates predictors of drinking intentions. *Journal of Experimental Social Psychology* 48(5):1196–1200.
- Chen, A.; Lu, Y.; and Wang, B. 2017. Customers’ purchase decision-making process in social commerce: a social learning perspective. *International Journal of Information Management* 37(6):627–638.
- Chung, J.; Gulcehre, C.; Cho, K.; and Bengio, Y. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS*, 1–9.
- Collins, J. E.; White, S.; Searle, S. M.; and Stemple, D. L. 2012. Incorporating rna-seq data into the zebrafish ensembl genebuild. *Genome Research* 22(10):2067–2078.
- Fishbein, M.; Triandis, H. C.; Kanfer, F. H.; Becker, M.; and Midlstedt, S. E. 2000. Factors influencing behavior and behavior change.
- Goldberg, Y., and Levy, O. 2014. Word2vec explained: deriving mikolov et al.’s negative-sampling word-embedding method. *CoRR* abs/1402.3722.
- Guidotti, R.; Rossetti, G.; Pappalardo, L.; Giannotti, F.; and Pedreschi, D. 2018. Personalized market basket prediction with temporal annotated recurring sequences. *IEEE Transactions on Knowledge and Data Engineering* 99(9):1–12.
- Hu, L.; Cao, L.; Wang, S.; Xu, G.; Cao, J.; and Gu, Z. 2017. Diversifying personalized recommendation with user-session context. In *IJCAI*, 1858–1864.
- Hu, L.; Jian, S.; Cao, L.; and Chen, Q. 2018. Interpretable recommendation via attraction modeling: Learning multilevel attractiveness over multimodal movie contents. In *IJCAI*, 3400–3406.
- Jang, E.; Gu, S.; and Poole, B. 2017. Categorical reparametrization with gumble-softmax. In *ICLR*, 1–13.
- Kingma, D. P., and Ba, J. 2015. Adam: A method for stochastic optimization. *CoRR* abs/1412.6980.
- Le, D.-T., and Lauw, H. W. 2019. Correlation-sensitive next-basket recommendation. In *IJCAI*, 2808–2814.
- Liu, Y.; Zhao, L.; Liu, G.; Lu, X.; and Gao, P. 2018. Dynamic bayesian logistic matrix factorization for recommendation with implicit feedback. In *IJCAI*, 3463–3469.
- Malle, B. F., and Knobe, J. 1997. The folk concept of intentionality. *Journal of Experimental Social Psychology* 33(2):101–121.
- Quadrana, M.; Karatzoglou, A.; Hidasi, B.; and Cremonesi, P. 2017. Personalizing session-based recommendations with hierarchical recurrent neural networks. In *RecSys*, 130–137.
- Rendle, S., and et al. 2010. Factorizing personalized markov chains for next-basket recommendation. In *WWW*, 811–820.
- Shen, Y.; Tan, S.; Sordoni, A.; and Courville, A. 2019. Ordered neurons: Integrating tree structures into recurrent neural networks. In *ICLR*, 1–14.
- Tanbeer, S. K.; Ahmed, C. F.; Jeong, B.-S.; and Lee, Y.-K. 2009. Sliding window-based frequent pattern mining over data streams. *Information Sciences* 179(22):3843–3865.
- Wan, S.; Lan, Y.; Wang, P.; Guo, J.; Xu, J.; and Cheng, X. 2015. Next basket recommendation with neural networks. In *RecSys*, 1–2.
- Wan, M.; Wang, D.; Liu, J.; Bennett, P.; and McAuley, J. 2018. Representing and recommending shopping baskets with complementarity, compatibility and loyalty. In *CIKM*, 1133–1142.
- Wang, S., and et al. 2019. A survey on session-based recommender systems. *arXiv preprint arXiv:1902.04864*.
- Wang, P.; Guo, J.; Lan, Y.; Xu, J.; Wan, S.; and Cheng, X. 2015. Learning hierarchical representation model for next basket recommendation. In *SIGIR*, 403–412.
- Wang, S.; Hu, L.; Cao, L.; Huang, X.; Lian, D.; and Liu, W. 2018. Attention-based transactional context embedding for next-item recommendation. In *AAAI*, 2532–2539.
- Wang, S.; Hu, L.; Wang, Y.; and et al. 2019a. Modeling multi-purpose sessions for next-item recommendations via mixture-channel purpose routing networks. In *IJCAI*, 3771–3777.
- Wang, S.; Hu, L.; Wang, Y.; and et al. 2019b. Sequential recommender systems: Challenges, progress and prospects. In *IJCAI*, 6332–6338.
- Wang, S.; Hu, L.; and Cao, L. 2017. Perceiving the next choice with comprehensive transaction embeddings for online recommendation. In *ECML-PKDD*, 285–302.
- Xingjian, S.; Chen, Z.; Wang, H.; Yeung, D.-Y.; Wong, W.-K.; and Woo, W.-c. 2015. Convolutional lstm network: a machine learning approach for precipitation nowcasting. In *NIPS*, 802–810.
- Yang, P.; Zhao, P.; Liu, Y.; and Gao, X. 2018. Robust cost-sensitive learning for recommendation with implicit feedback. In *Proceedings of the 2018 SIAM International Conference on Data Mining*, 621–629.
- Yao, Y.; Shen, F.; Zhang, J.; Liu, L.; Tang, Z.; and Shao, L. 2018. Extracting privileged information for enhancing classifier learning. *IEEE Transactions on Image Processing* 28(1):436–450.
- Yu, F.; Liu, Q.; Wu, S.; Wang, L.; and Tan, T. 2016. A dynamic recurrent model for next basket recommendation. In *SIGIR*, 729–732.