Supplementary Materials for OMBA: User-Guided Product Representations for Online Market Basket Analysis

Amila Silva, Ling Luo, Shanika Karunasekera, and Christopher Leckie

School of Computing and Information Systems
The University of Melbourne
Parkville, Victoria, Australia
{amila.silva@student., ling.luo@, karus@, caleckie@}unimelb.edu.au

Abstract. This is the supplementary material for OMBA: User-Guided Product Representations for Online Market Basket Analysis [2]. The structure of this material is as follows. In Section 1, we present detailed results of the empirical analysis conducted to verify users' repetitive purchasing behavior. Section 2 provides the derivation of the relationship between products' selling price and their frequency of occurrence in shopping baskets. Section 3 theoretically proves that the rules generated from OMBA-ARM have higher Lift with the correct selection of hyper-parameters. We provide detailed results for the hyper-parameter sensitivity of OMBA in Section 4. More results on intra-basket item retrieval tasks are presented in Section 5.

Keywords: Market Basket Analysis · Online Learning · Item Representations · Transaction Data

1 Empirical Analysis on Users' Purchasing Behavior

In this analysis, the following question is investigated: *Do users tend to buy the same items over and over again?*. If this is true, users exhibit repeating buying patterns and the historical transactions of a user could be useful to predict the user's future transactions. We conduct the following test to explore the aforementioned research question:

1. For each shopping basket b, represent the products in b by a TFIDF vector I_p^b (considering transactions as documents and products as words).
2. Randomly sample a user u, who have at least two transactions. Sample two transactions, b_x and b_y, from u's transaction history and calculate the similarity of b_x and b_y using cosine similarity between $I_p^{b_x}$ and the $I_p^{b_y}$
3. Perform Step 2 k times to construct the set of similarity scores, S_{same_users}, which denotes the similarity of the transactions of the same user.
4. Similarly, perform Step 2 iteratively k times with the tweet pairs from different users to construct S_{diff_users}.

Table 1: Descriptive statistics of the datasets

Datasets	# Users	# Items	# Transactions	# Baskets
Complete Journey (CJ)	2,500	92,339	2,595,733	276,483

5. Test hypothesis $H_0 : \overline{S_{same\_users}} \leq \overline{S_{diff\_users}}$ statistically (using one-tailed t-test) to verify the research question (H_0 should be rejected to verify the question), where $\overline{S_{same\_users}}$ and $\overline{S_{diff\_users}}$ denote the mean values of S_{same_users} and S_{diff_users} respectively.

We experiment with $k = 100,000$ using the finest product level category of Complete Journey (CJ), a publicly available real-world dataset (the statistics are shown in Table 1). For $k \geq 100,000$, $\overline{S_{same\_users}}$ and $\overline{S_{diff\_users}}$ remain stable around 0.0575 and 0.0043 respectively. This results in an almost zero p-value ($t_{stat} = 82.4665, p_{val} = 0.0$) for the aforementioned t-test, which verifies the users' tendency to buy same set of products repeatedly.

2 Product's Selling Price vs Frequency of Occurrence

In this section, we derive a relationship between the price of a product and its likelihood to appear in a transaction.

Initially, we assume that the number of appearances, $h(x)$, of product x follows a power-law distribution with respect to its selling price $SV(x)$ ($in\ dollars$) (see Figure 1a). Based on this assumption, a power-law formula is fitted to the curve in Figure 1a, which can be simplified to a linear regression problem as follows.

$$\begin{aligned} h(x) &= p * SV(x)^q \\ ln(h(x)) &= ln(p) + q * ln(SV(x)) \end{aligned} \tag{1}$$

By fitting Equation 1 using the curve in Figure 1a, p and q can be found as $5.4 * 10^6$ and -2.3 respectively (see Figure 1b for the fitted curve). For a given

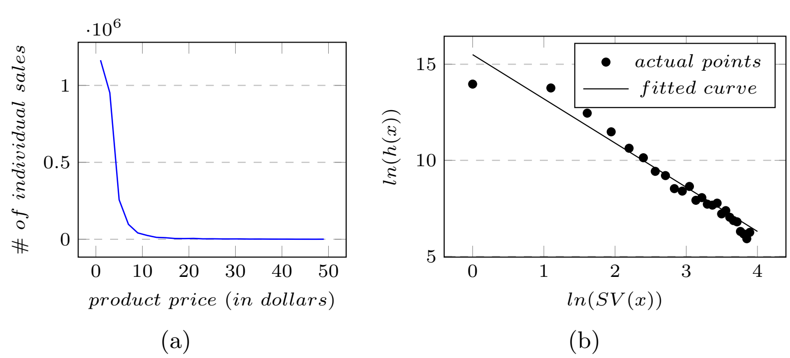


Fig. 1: (a) Number of products' sales with respect to the products' prices in CJ dataset; (b) Fitted linear curve, $ln(h(x)) = 15.5 - 2.3 * ln(SV(x))$, for Equation 1 assuming Figure 1a follows a power-law distribution

shopping basket b, the probability to have product x of price $SV(x)$ is computed as:

$$\begin{aligned} p(x \in b) &= \frac{5.4 * 10^6 * SV(x)^{-2.3}}{\int_0^{\inf} 5.4 * 10^6 * SV(x_0)^{-2.3} d(SV(x_0))} \\ &= 1.3 SV(x)^{-2.3} \end{aligned} \tag{2}$$

3 Finding Optimal $|F|$ and $|H|$ in OMBA-ARM

In this section, the optimal values for $|F|$ and $|H|$ (see Section 4.3 in [2] to learn about these parameters) are found such that they establish a theoretical relationship between the association rules generated from OMBA-ARM and Lift. Initially, we form two closed-form solutions to model the likelihood to have a strong association between product x and product y based on: (1) OMBA-ARM; and (2) Lift.

(1) Based on OMBA-ARM. products x and y should collide at least in a single hash table to have a strong association. Thus, the likelihood to have a strong association between product x and product y is:

$$p(x \Rightarrow y)_{omba} = 1 - (1 - p(sgn(v_x \cdot f) = sgn(v_y \cdot f))^{|F|})^{|H|} \tag{3}$$

where $sgn(x) = 1 \; if \; x \geq 1; and -1 \; otherwise$. $p(sgn(v_x \cdot f) = sgn(v_y \cdot f))$ (i.e., the likelihood to have a similar hash value for products x and y with respect to a random vector) can be computed from the following lemma.

Lemma 1. *For a given pair of normalized vectors v_x and v_y, the likelihood to produce similar sign values from their signed projections with respect to a random vector f is:*

$$p(sgn(v_x \cdot f) = sgn(v_y \cdot f)) = 1 - \frac{1}{\pi} arccos(v_x \cdot v_y)$$

Proof.

$$\begin{aligned} p(sgn(v_x \cdot f) = sgn(v_y \cdot f)) &= 1 - p(sgn(v_x \cdot f) \neq sgn(v_y \cdot f)) \\ &= 1 - (p(v_x \cdot f \geq 0, v_y \cdot f < 0) + p(v_y \cdot f \geq 0, v_x \cdot f < 0)) \\ &= 1 - 2 * p(v_x \cdot f \geq 0, v_y \cdot f < 0) \quad \textit{(By symmetry)} \end{aligned}$$

The set $\{f : v_x \cdot f \geq 0, v_y \cdot f < 0\}$ represents the intersection of two half spaces whose dihedral angle is $arccos(v_x \cdot v_y)$.Since a full sphere (i.e., covers all possible random vectors) has a dihedral angle of 2π, $p(v_x \cdot f \geq 0, v_y \cdot f < 0) = \frac{arccos(v_x \cdot v_y)}{2\pi}$.

$$p(sgn(v_x \cdot f) = sgn(v_y \cdot f)) = 1 - \frac{arccos(v_x \cdot v_y)}{\pi}$$

The lemma follows.

By substituting from Lemma 1 to Equation 3,

$$p(x \Rightarrow y)_{omba} = 1 - (1 - (1 - \frac{arccos(v_x \cdot v_y)}{\pi})^{|F|})^{|H|} \tag{4}$$

(2) Based on Lift. In our negative sampling-based softmax approximation (see Section 4.2 in [2]), we sample $|N_z|$ negative samples for every genuine product x that co-occurred with y. Define the *Lift* score between x and y in the empirical distribution of the training dataset and the noise distribution as $Lift(y,x)_{train}$ and $Lift(y,x)_{noise}$ respectively. Then, the combined *Lift* score from both these distributions can be computed as a mixture of the both scores, which are weighted based on the number of samples coming from each:

$$Lift(y,x) = \frac{1}{|N_z|+1} Lift(y,x)_{train} + \frac{|N_z|}{|N_z|+1} Lift(y,x)_{noise} \tag{5}$$

However, the samples coming from $Lift(y,x)_{train}$ alone represent the true co-occurances of x and y. Thus, the likelihood to have a strong association between product x and y (based on *Lift*) can be computed as:

$$\begin{aligned} p(x \Rightarrow y)_{lift} &= \frac{\frac{1}{|N_z|+1} Lift(y,x)_{train}}{\frac{1}{|N_z|+1} Lift(y,x)_{train} + \frac{|N_z|}{|N_z|+1} * Lift(y,x)_{noise}} \\ &= \frac{Lift(y,x)_{train}}{Lift(y,x)_{train} + |N_z| * Lift(y,x)_{noise}} \end{aligned} \tag{6}$$

By substituting $Lift(x,y)$ as $P(y|x)/P(y)$,

$$\begin{aligned} p(x \Rightarrow y)_{lift} &= \frac{P_{train}(y|x)/P_{train}(y)}{P_{train}(y|x)/P_{train}(y) + |N_z| * P_{noise}(y|x)/P_{train}(y)} \\ &= \frac{P_{train}(y|x)}{P_{train}(y|x) + |N_z| P_{noise}(y|x)} \end{aligned} \tag{7}$$

where P_{train} and P_{noise} represents the probabilities computed using the empirical distribution of the training dataset and the noise distribution respectively. In Equation 7, $Lift_{noise}(x,y)$ is computed as $P_{noise}(y|x)/P_{train}(y)$, because the context products (product y in this case) are always selected from the distribution of the training dataset (see Section 4.2 in [2]).

According to negative sampling based softmax approximation in [1] (which is used to approximate softmax in this work), OMBA learns product representations that satisfy the following two constraints (see [1] for more details):

$$|N_z| P_{noise}(y|x) = |N_z| P_{noise}(y) = 1 \tag{8}$$

$$\sum_{\forall x} exp(A * v_y \cdot v_x) = 1 \ (self \ normalizing \ constraint) \tag{9}$$

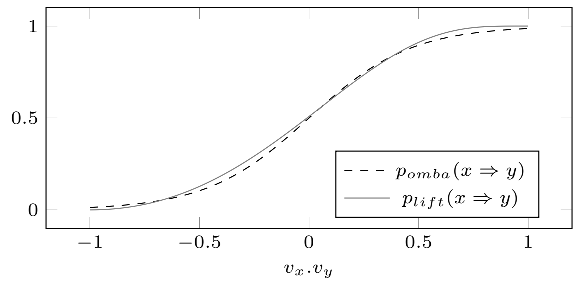


Fig. 2: Probability to have a strong association between x and y (as modelled by OMBA and LIFT) with respect to the dot product of their embeddings for $|F| = 4$, $|H| = 11$, and $A = 4.3$. The corresponding curves for other parameter values are simulated at https://www.desmos.com/calculator/tfssy4w7lh

where v_x and v_y denote the normalized embeddings in Equation 4. However, v_x and v_y are not necessarily normalized in Equation 9. Thus, the constant A is used to compensate for that. By substituting Equations 8 and 9 into Equation 7,

$$\begin{aligned}
p(x \Rightarrow y)_{lift} &= \frac{P_{train}(y|x)}{P_{train}(y|x)+1} && (from\ Eq.\ 8)\\
&= \frac{\frac{exp(A*v_x \cdot v_y)}{\sum_{\forall x} exp(A*v_y \cdot v_x)}}{\frac{exp(A*v_x \cdot v_y)}{\sum_{\forall x} exp(A*v_y \cdot v_x)}+1} && (from\ Eq.\ 1\ in\ [2]\ )\\
&= \frac{exp(A*v_x \cdot v_y)}{exp(A*v_x \cdot v_y)+1} && (from\ Eq.\ 9)\\
&= \sigma(A*v_x \cdot v_y) && (10)
\end{aligned}$$

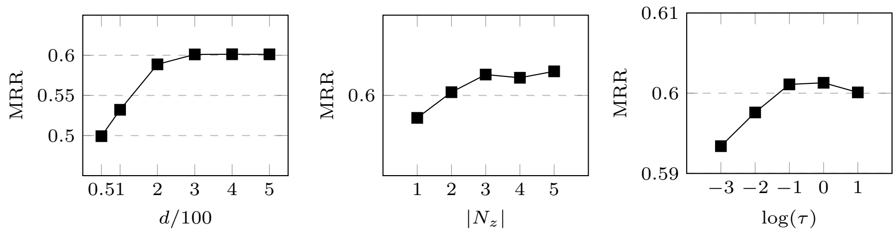
Optimal solution to $|F|$ and $|H|$. Then, the integer solutions for parameters $|F| = 4$ and $|H| = 11$, and real solutions for $A = 4.3$ are found such that $p(x \Rightarrow y)_{omba} = p(x \Rightarrow y)_{lift}$ (see Figure 2 for the curves of $p(x \Rightarrow y)_{omba}$ and $p(x \Rightarrow y)_{lift}$ with the selected parameters). Such a selection of hyper-parameters theoretically guarantees that the rules produced from OMBA-ARM are higher in *Lift*, which is a statistically well defined measure for strong associations.

4 Hyper-parameter Sensitivity of OMBA

Figure 3 shows the sensitivity of the hyper-parameters of the OMBA-OME module, which returns the ideal parameters for the *intra-basket item retrieval task* as: (1) $d = 300$; (2) $|N_z| = 3$; and (3) $\tau = 0.1$.

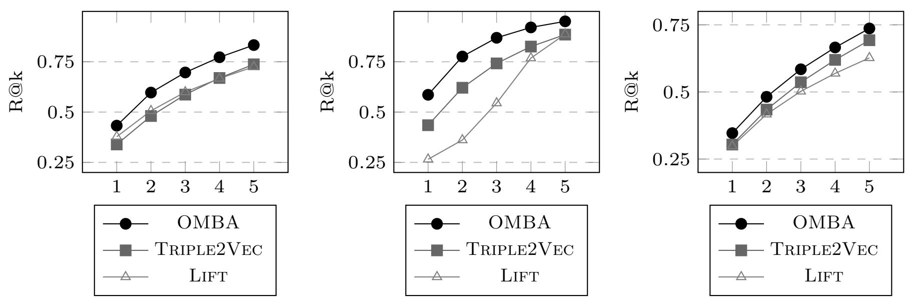
5 More results on *intra-basket item retrieval*

Figure 4 shows the results, evaluated using recall at different k values, for the *intra-basket item retrieval* task using the CJ, IC, and TF datasets. The results



(a) Effect of d (b) Effect of $|N_z|$ (c) Effect of τ

Fig. 3: MRRs for the *intra-basket item retrieval* task with different hyper-parameter settings using CJ dataset



(a) CJ Dataset (b) IC Dataset (c) TF Dataset

Fig. 4: Recall at different k values for *intra-basket item retrieval*

show that OMBA consistently outperforms other two baselines for different k values.

References

1. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Proc. of NIPS (2013)
2. Silva, A., Luo, L., Karunasekera, S., Leckie, C.: OMBA: User-Guided Product Representations for Online Market Basket Analysis. In: Proc. of ECML-PKDD (2020)