ショートノート

Word2Vecに基づく購買履歴からの アイテムベクトル学習†

名渡山 夏子*¹・岡本 一志*²

自然言語処理の分散表現技術のひとつである Word2Vec を購買履歴に応用することで、低計算コストでのアイテムベクトル学習の実現を目指す。実 POS データを用いた実験により、Word2Vec のパラメータであるウィンドウサイズと次元数およびトランザクションデータの与え方がアイテムベクトルの学習に与える影響を検証する。実験結果より、ウィンドウサイズは可能な限り大きくし、次元数は 40 以上に設定し、1 トランザクション内の商品の種類に基づいたデータを与えることで、特徴空間上で同じ分類の商品が近傍に現れやすくなるようなアイテムベクトル学習となることを確認している。

キーワード：推薦システム、協調フィルタリング、購買履歴、アイテムベクトル、分散表現

1. はじめに

協調フィルタリングでは、ユーザの購入履歴やレーティングからユーザ-アイテム行列を作成し、類似するユーザや購入が期待できるアイテムなどを予測する。そのアプローチのひとつとして、ユーザ-アイテム行列からユーザベクトルとアイテムベクトルに行列を分解する手法 [1–3] があり、分解により得られたベクトルから類似度計算を行うことで推薦処理を実現している。しかしながら、ユーザ-アイテム行列は巨大な疎行列であり、行列分解には現実的な時間や機器で計算できない課題が生じる。

本研究では、自然言語処理における分散表現の技術のひとつである Word2Vec [4] を用いて、商品購買履歴から低い計算コストでのアイテムベクトルの生成を目指す。Word2Vec の学習処理は 1 文単位で行われ、商品購買履歴への応用では、ユーザ毎の購入商品リストを入力とするため、巨大なユーザ-アイテム行列を扱わずに学習することができる。既に Word2Vec を商品購買履歴に適用した研究 [5] が行われているものの、この研究では Word2Vec のパラメータ設定に関する指針は示されていない。さらに、近傍の単語を使って学習する Word2Vec において単語の並びは重要な要素であるものの、購入商品リスト内の商品の並び（単語の並びに対応）をどのようにすれば良いかも明らかではない。

パラメータの設定やデータの与え方がアイテムベクトルの生成に与える影響を明らかにするため、Word2Vec を実 POS データに適用しアイテムベクトルの特性を検証する実験を行う。こ

の実験では、Word2Vec のパラメータであるウィンドウサイズを 1~10、次元数を 2~100 に 1 ずつ変化させる。さらに、購入商品リスト内に購入数分だけ同じ商品を重複して並べることや、購入商品リスト内の商品の順序をランダムに並び替える実験も行う。学習したアイテムベクトルの良さを、同じ商品分類の商品が特徴空間上で近傍に出現しやすいことと仮定し、評価尺度を定め、その尺度に基づき最適なパラメータや購入商品リストの与え方について検討する。

本論文の構成を次に示す。2 節では、アイテムベクトルを獲得するための既存手法やその問題点について提示し、その後、Word2Vec を適用するメリットについて述べる。3 節では、本研究で用いる実 POS データの詳細や Word2Vec に入力するための変換法について説明し、そして、Word2Vec のパラメータや購入商品リストの並べ替えの方法を示す。4 節では、得られたアイテムベクトルの評価尺度を定め、パラメータ等の違いがアイテムベクトル学習に与える影響について考察する。

2. アイテムベクトルの獲得と Word2Vec

協調フィルタリングでは、 m 人のユーザが n 個のアイテムについて評価した（購入数やレーティングに対応、評価していないという情報も含まれる） $m \times n$ のユーザ-アイテム行列を用いて類似するユーザや購入が期待できるアイテムなどを予測する。協調フィルタリングには、 $m \times n$ のユーザ-アイテム行列を $m \times k$ のユーザ行列と $k \times n$ のアイテム行列に分解するアプローチがある [1–3]。このアプローチでは、行列分解により得られたユーザベクトルやアイテムベクトルを用いた類似度計算により情報推薦を実現している。現在、行列分解の手法として、潜在意味解析（Latent Semantic Analysis）[2] や非負値行列因子分解（Non-negative Matrix Factorization）[3] などが応用されている。しかしながら、これらの行列分解の手法は、巨大な疎行列を対象とすることから、計算コストが高い課題がある。また、新規のユーザやアイテムも推薦対象として含めるために、再度行列分解を行うことも想定される。そのため、ユーザベクトルやアイテムベクトルをできるだけ低い計算コストで

† Word2Vec Based Item Vector Learning from Purchase Data
Natsuko NADOYAMA and Kazushi OKAMOTO

*1 電気通信大学情報理工学部（現在、クックパッド株式会社）
Faculty of Informatics and Engineering, The University of Electro-Communications (Currently, Cookpad Inc.)

*2 電気通信大学大学院情報理工学研究科
Graduate School of Informatics and Engineering, The University of Electro-Communications

獲得する手法が欠かせない。

自然言語処理において、分散表現は単語を低次元の実数ベクトルで表現することを指し、その技術のひとつとして Word2Vec が提案されている [4]。Word2Vec は文の集合を入力とし、注目単語の近傍に現れる単語から注目単語のベクトル表現を学習する。近年では、商品（アイテム）のベクトル表現の獲得に Word2Vec を商品購買履歴に適用した研究も報告されている [5]。Word2Vec は 3 層のフィードフォワードニューラルネットワークに基づく手法であり、学習処理は 1 文単位で行われる。Word2Vec の商品購買履歴への応用ではユーザ毎の購入商品リストを入力とするため、巨大なユーザ-アイテム行列を扱う必要がなく、ユーザ毎の購入商品リストを逐次処理することでアイテムベクトルの学習を行うことになる。これにより、Word2Vec の商品購買履歴への応用には、行列分解のアプローチに比べアイテムベクトル獲得に必要な計算コストが低いメリットがある。

その一方で、文献 [5] では Word2Vec のパラメータ設定に関する指針は示されていない。加えて、近傍の単語を使って学習する Word2Vec において単語の並びは重要な要素であるものの、購入商品リスト内の商品の並び（単語の並びに対応）をどのようにすれば良いかも明らかになっていない。適用先が自然言語における文ではなく商品購買履歴であり、Word2Vec をアイテムベクトルの獲得に活用するには、パラメータや購入商品リスト内の商品の並びの定め方を明らかにする必要がある。

3. リスト内の商品の並びと Word2Vec のパラメータがアイテムベクトル学習に与える影響の評価実験

Word2Vec を実 POS データに適用し、パラメータの設定やデータの与え方がアイテムベクトルの生成に与える影響を検証する実験を行う。1 回の買い物で同時に購入した商品の集合を 1 トランザクションとする。Word2Vec には学習後のベクトルの次元数や、何個隣の単語（商品）まで学習するのかを意味するウィンドウサイズなどのパラメータがあり、単に Word2Vec を適用するだけでなく、商品購買履歴に適したパラメータの与え方を明らかにする。また、トランザクション内の商品の並びは商品をレジに通した順序に相当するが、近傍の単語を使って学習する Word2Vec において単語の並びは重要な要素であるため、トランザクション内の商品の並びがアイテムベクトルの学習に与える影響も検証する。

3.1 トランザクションデータへの変換

本実験では、商品購買履歴として、日経メディアマーケティング株式会社が販売する NEEDS SCAN（コーヒー飲料をレシートに含む POS データ）を用いる。この POS データのレシート（トランザクション）数は 593,793、総商品数は 10,914 であり、全ての商品は 192 の大分類に人手で分類されている [6]。データの収集期間は 2014 年 4 月 1 日から 2015 年 3 月 31 日までの 1 年間である。

POS データには商品名や価格、商品コードなど全 20 項目の商品情報が記載されており、本実験ではその中からレシート番

レシート番号	商品コード	数量	分類コード	購入商品名
1	10000	2	130	缶コーヒー1
1	20000	1	80	パン
2	30000	3	130	缶コーヒー2

(a) POS データ

レシート番号	重複	非重複
1	{10000, 10000, 20000}	{10000, 20000}
2	{30000, 30000, 30000}	{30000}

(b) トランザクションデータ

図 1 本実験で用いる POS データおよびトランザクションデータへの変換例

号、商品コード（JAN コード）、購入商品名、分類コード、数量を使用する。本実験で扱う POS データの例を図 1(a) に示す。商品コードとは商品に対して一意に与えられる数字である。分類コードとは類似商品の集約・処理を簡単に行うために定められた商品分類（カテゴリ）を意味する。また、レシート番号はトランザクション番号に相当し、POS データにおいては 1 回の買い物で同時に購入した商品には同じレシート番号が割り当てられる。

本実験では、Word2Vec を POS データに適用するために、POS データを Word2Vec の入力形式に従ったトランザクションデータに変換する。図 1(a) の POS データの変換例を図 1(b) に示す。図 1(b) の表全体をトランザクションデータ、各行をトランザクションと呼び、本実験では、レシート番号を元に商品毎に商品コードを並べた形式（非重複）か、商品の数量分まで加味して商品コードを並べた形式（重複）の 2 種類を想定する。また、Word2Vec における単語を商品の商品コード、1 文をトランザクション（1 回の買い物で同時に購入した商品）として扱う。なお、トランザクションデータを作成する際に、次のいずれかの条件を満たす POS データはトランザクションデータに変換する際に取り除く。

1. 購入商品名または分類コードが未記入の商品
2. 分類コードが商品分類コードブックに掲載されていない商品
3. トランザクションの長さが 1 で、かつ、その商品が他のトランザクションに含まれていない商品

NEEDS SCAN には欠損があるデータも存在し、実験では分類コードのみを利用することから 1, 2 の条件を設ける。また、Word2Vec では近傍の単語を使って学習を行うが、トランザクションの長さが 1 の場合、近傍の商品が存在しないため学習を行えない。当該商品が他のトランザクションに含まれているならば学習を行えるが、トランザクションの長さが 1 かつ他のトランザクションと共起しない商品は学習に使えないため 3 の条件を設けることとする。

3.2 トランザクションデータの与え方

同じ商品を複数個同時に購入する際を想定し、重複を考慮する場合と考慮しない場合の 2 種類のデータセットを用意する。重複のデータセットは同じ商品を 1 トランザクションの中に購入数分だけ並べることに相当する。例えば、パンを 2 個、

コーヒーを 1 缶購入した POS データについては、重複のデータセットでは{パン, パン, コーヒー}, 非重複のデータセットでは{パン, コーヒー}のトランザクションに変換することになる(実際にはパンとコーヒーの商品コードを並べることに相当)。重複のデータセットでは 1 トランザクションの長さが増加し、同じ商品を使って学習することになる。一般的な文章では同じ単語が近傍に現れることはあまり無く、Word2Vec では同じ単語を使って学習を行うことは例外的な扱いといえる。本実験で用いる POS データでは、1 トランザクションあたりの平均商品数は、重複のデータセットに変換した場合 2.74、非重複のデータセットに変換した場合 2.49 となる。

また、トランザクション内の商品の並びがアイテムベクトルの学習に与える影響を検証するため、トランザクション内の商品をランダムに並び替えることを行う。具体的には、トランザクション内の商品の並びをランダムにシャッフルしたものを s 種類作成する。 s は 1 から 20 までとする。例えば、 $s = 2$ では、{パン, コーヒー, お茶}のトランザクションから{お茶, パン, コーヒー}と{お茶, コーヒー, パン}の 2 種類のトランザクションを生成する。

3.3 Word2Vec のパラメータ設定

本実験では、Mikolov らによる実装の Word2Vec (<https://code.google.com/archive/p/word2vec/>) を用いる。この実装では、最小出現数のデフォルトパラメータは 5 に設定されているが、商品の出現回数に関係なくすべての商品を学習する必要があるため本実験では最小出現数を 0 に設定する。また、文献 [5] と同様にネガティブサンプリング数は 15、サブサンプリングの閾値は 10^{-3} とする。自然言語処理の 1 文の長さに比べ 1 トランザクション内の商品数は少ないため、本実験では何個隣の単語まで学習するのかを表すウィンドウサイズと、学習後のベクトルの次元数の 2 つのパラメータを変化させ、このことがアイテムベクトルの学習にどのような影響を与えるかを評価する。実験で用いる POS データではトランザクションの長さが 5 以下のトランザクション数は全体の 90% を超えているものの、適切なウィンドウサイズを求めるためにウィンドウサイズを 1 から 10 まで 1 ずつ変化させる。アイテムベクトルの次元数が低い場合、内積や距離といった(非)類似度計算における計算コストが低くなるメリットがある一方で、アイテムベクトルの次元数が低くなりすぎると商品をベクトルとして表現する際の表現力が低下する可能性が生じる。そこで、本実験では最低限必要な次元数を検討するため次元数を 2 から 100 まで 1 ずつ変化させることを行う。

4. アイテムベクトルの評価尺度と最適なパラメータ設定に関する考察

学習したアイテムベクトルの良さを、同じ商品分類の商品が特徴空間上で近傍に出現しやすいことと仮定する。その定量的評価は、各分類に属するアイテムベクトル間のユークリッド距離の総和により行う。

n 個の商品について、ランダムシャッフル数(倍率) s ・次元数 d ・ウィンドウサイズ w のパラメータで学習

したアイテムベクトルと分類ラベルをペアにした集合を $X_{s,d,w} = \{(\mathbf{v}_1, c_1), (\mathbf{v}_2, c_2), \dots, (\mathbf{v}_i, c_i), \dots, (\mathbf{v}_n, c_n)\}$ とする。 $\mathbf{v}_i \in V \subset \mathbb{R}^d$ は i 番目の商品のアイテムベクトルを、 $c_i \in L = \{l_1, l_2, \dots, l_m\}$ は i 番目の商品が m 種類ある商品分類の内どの分類に属しているかを表している。 c_i は事前に定められているものとし、本実験では POS データに付属する商品分類コードブック [6] の大分類を割り当てるものとする(すなわち $n = 593793$, $m = 192$)。

商品分類 $l \in L$ に属する商品の集合が、 d 次元の特徴空間上で近傍に集まっているかどうかは

$$\epsilon(X_{s,d,w}, l) = \sum_{j=1}^{n-1} \sum_{k=j+1}^n \mu(c_j, l) \mu(c_k, l) \left\| \frac{\mathbf{v}_j}{\|\mathbf{v}_j\|} - \frac{\mathbf{v}_k}{\|\mathbf{v}_k\|} \right\| \quad (1)$$

$$\mu(c, l) = \begin{cases} 1 & (c = l) \\ 0 & (\text{otherwise}) \end{cases} \quad (2)$$

$$\|\mathbf{v}\| = \max \{\|\mathbf{v}_1\|, \|\mathbf{v}_2\|, \dots, \|\mathbf{v}_i\|, \dots, \|\mathbf{v}_n\|\} \quad (3)$$

のように定義する。式 (3) はアイテムベクトル集合 V の中の最大ノルムを意味しており、全てのアイテムベクトルを $\|\mathbf{v}\|$ で割ることで、 V に含まれるベクトルのノルムの最大値が 1 になるよう正規化している。関数 ϵ の値が小さくなるほど同じ分類のアイテムベクトル間の距離が近いことを表しており、特徴空間上で商品分類 l の商品が近傍に現れやすくなることに相当する。そして、全商品分類について ϵ を求め、その総和

$$f(X_{s,d,w}) = \sum_{l \in L} \epsilon(X_{s,d,w}, l) \quad (4)$$

により同じ分類の商品が特徴空間上で近傍に出現しやすいかどうかの評価指標 f を定める。本実験では、3.3 節で述べた通り、 $s \in \{0, 1, \dots, 20\}$, $d \in \{2, 3, \dots, 100\}$, $w \in \{1, 2, \dots, 10\}$ のパラメータを用いることとする。

4.1 ウィンドウサイズとランダムシャッフル数の違いによる影響評価

次元数 $d = 40$ のとき、学習済アイテムベクトルと分類ラベルをペアにした集合 $X_{s,40,w}$ の評価値 $f(X_{s,40,w})$ をヒートマップで可視化した図を図 2 に示す。図 2 において、横軸はウィンドウサイズ w 、縦軸は学習に用いるトランザクションデータの倍率 s ($s = 0$ は元のトランザクションデータ) であり、(a) は重複のデータセット、(b) は非重複のデータセットに対応している。ヒートマップではセルの色が黄色に近づくにつれ $f(X_{s,40,w})$ の値が低くなり、濃青に近づくにつれ $f(X_{s,40,w})$ の値が高くなることを示している。図 2 より、重複・非重複のデータセットを問わず s が 0 または 1 のときに $f(X_{0,40,w})$ および $f(X_{1,40,w})$ の値が特に低くなることを確認できる。本実験では、1 トランザクションから s トランザクションを生成しているが、 s の値が増加するにつれ $f(X_{s,40,w})$ の値が高くなっており、同じ分類の商品が特徴空間上で近傍に現れづらくなると考える。また、 s の値に関わらず w が 1 のときは $f(X_{s,40,1})$ の値が高く、 w が増加するにつれて $f(X_{s,40,w})$ の値が低くなる傾向が確認できる。 w は 1 トランザクションにおいて何個隣の商品まで学習に使うのかを表すパラメータであり、 w が大きくなるにつれ学習に用いる商品数が増えるため、多くの商品で

(a) 重複のデータセット

(a) 重複のデータセット

(b) 非重複のデータセット

(b) 非重複のデータセット

図2 アイテムベクトルと分類ラベルをペアにした集合 $X_{s,40,w}$ の評価値 $f(X_{s,40,w})$ のヒートマップ

図3 $s = 1$ で 500 回実験した際の評価値 $f(X_{1,40,w})$ のヒートマップ

学習を行うことにより同じ分類の商品が特徴空間上で近傍に現れやすくなると考える。

4.2 トランザクション内の商品の並びの違いによる影響評価

トランザクションデータの倍率 $s = 1$ のときに $f(X_{1,40,w})$ の値が低くなることを確認しており、より詳細な分析を行うため $s = 1$ でのアイテムベクトル学習を 500 回行う。次元数 d のとき、500 回の実験により得られた学習済アイテムベクトルと分類ラベルをペアにした集合を

$$\{X_{1,d,1}^1, X_{1,d,1}^2, \dots, X_{1,d,1}^{500}, X_{1,d,2}^1, \dots, X_{1,d,10}^{500}\} \quad (5)$$

とする。ここでは、 $d = 40$ のときの $f(X_{1,40,w})$ に関するヒートマップを図3に示す。図3では、横軸をウィンドウサイズ、縦軸を試行回数としている。図3より、(b)の非重複のデータセットのほうが $f(X_{1,40,w})$ の値が低いことが確認できる。Word2Vecは近傍の単語を使って学習を行うため、重複のデータセットでは自分自身の商品コードを使って学習することになる。図3より、(a)の重複のデータセットは、(b)の非重複の

データセットに比べて全体的に濃青になっており、自分自身の商品コードを使って学習を行うことが $f(X_{1,40,w})$ の値に悪影響を与える傾向にあると考える。また、 $w = 1$ のとき $f(X_{1,40,1})$ の値が最も高く、 w が増加するにつれて $f(X_{1,40,w})$ の値が低くなるのがわかる。これは、4.1節と同様に、 w が大きくなると学習に用いる近傍の商品が増えるため、同じ分類の商品が特徴空間上で近傍に現れやすくなるためと考える。

4.3 安定する次元数の検証

4.2節の実験結果より、非重複のデータセットの方が重複のデータセットに比べて $f(X_{1,40,w})$ の値が低くなり、さらに w の増加により $f(X_{1,40,w})$ の値が低くなることを確認している。次に、適切な次元数について検証するため、非重複のデータセットを用い、次元数を変化させた場合の $f(X_{1,d,w})$ の値を比較する。4.2節と同様に、学習済アイテムベクトルと分類ラベルのペアの集合 $X_{1,d,w}$ の評価値 $f(X_{1,d,w})$ の結果に関するヒートマップを作成する。図4は次元数 d を 20, 40, 60 で固定し作成したヒートマップである。なお、図4(b)は図3(b)のスケールを変更した図に等しい。図4(b)、図4(c)では $w = 1$ のと

図4 非重複のデータセットでの500回の試行における各次元のアイテムベクトルの評価値 $f(X_{1,d,w})$ のヒートマップ

き最も $f(X_{1,d,w})$ の値が高く、 w が増加するにつれ $f(X_{1,d,w})$ の値が減少していることがわかる。すなわち、 d が40または60においては、 w が増加すると同じ分類の商品が特徴空間上で近傍に現れやすくなると考える。対して、図4(a)では $w = 3$

のときの $f(X_{1,20,3})$ の値と $w = 10$ のときの $f(X_{1,20,10})$ の値があまり変わらないようにも見え、図4(b)や図4(c)のように w の増加に伴って $f(X_{1,d,w})$ の値が低くなる傾向が明確に表れない。 $d \geq 40$ では w が増加すると $f(X_{1,d,w})$ の値は低くなる傾向になることがわかったが、安定する次元数 d とウィンドウサイズ w について定量的に考える。

ランダムシャッフル数 s ・次元数 d ・ウィンドウサイズ w のパラメータで学習済アイテムベクトルと分類ラベルのペアの集合において、 $f(X_{s,d,w})$ の値は同じ商品分類内のアイテムベクトル間のユークリッド距離の総和を意味している。このとき、 $d, d' \in \mathbb{N}$, $d \neq d'$ について、 $f(X_{s,d,w})$ と $f(X_{s,d',w})$ の値は異なる次元での距離の総和になるため、直接比較することができない。そこで、 t 回の試行で求めた $t \times w$ 種類の $f(X_{s,d,w})$ の最大値と最小値を用いて、 $f(X_{s,d,w})$ の値を0～1に正規化する。

ランダムシャッフル数 s 、次元数 d 、ウィンドウサイズ w 、 t 回目の試行で求めたアイテムベクトルの評価値を $f(X_{s,d,w}^t)$ と表す。 $w \in W = \{1, 2, \dots, 10\}$, $t \in T = \{1, 2, \dots, 500\}$ としたとき、 $f(X_{s,d,w})$ の最大値 $f_d^{\max}$ と最小値 $f_d^{\min}$ は

$$f_d^{\max} = \max \{f(X_{s,d,w}^t) | w \in W, t \in T\} \quad (6)$$

$$f_d^{\min} = \min \{f(X_{s,d,w}^t) | w \in W, t \in T\} \quad (7)$$

で表される。式(6)と式(7)を用いて、 $f(X_{s,d,w}^t)$ の正規化を

$$f'(X_{s,d,w}^t) = \frac{f(X_{s,d,w}^t) - f_d^{\min}}{f_d^{\max} - f_d^{\min}} \quad (8)$$

で定義する。このとき、任意の s, d, w の組み合わせにおいて $|T|$ 回計算した $f'(X_{s,d,w})$ の平均値 $\mu_{s,d,w}$ と標準偏差 $s_{s,d,w}$ は

$$\mu_{s,d,w} = \frac{1}{|T|} \sum_{t \in T} f'(X_{s,d,w}^t) \quad (9)$$

$$s_{s,d,w} = \sqrt{\frac{1}{|T|} \sum_{t \in T} (f'(X_{s,d,w}^t) - \mu_{s,d,w})^2} \quad (10)$$

のように求められる。また、 t 分布表から自由度 $|T| - 1$ 、信頼係数 95% に対応する τ 値を計算することで、信頼区間 $c_{s,d,w}^+, c_{s,d,w}^-$ も

$$c_{s,d,w}^+ = \mu_{s,d,w} + \frac{\tau * s_{s,d,w}}{\sqrt{|T|}} \quad (11)$$

$$c_{s,d,w}^- = \mu_{s,d,w} - \frac{\tau * s_{s,d,w}}{\sqrt{|T|}} \quad (12)$$

のように求められる。

横軸にウィンドウサイズ w 、縦軸に $f'(X_{1,d,w})$ の値をとり、平均値 $\mu_{1,d,w}$ と信頼区間 $[c_{1,d,w}^+, c_{1,d,w}^-]$ をプロットした図を図5に示す。図5において、(a)は非重複のデータセットで d が10, 20, 30, 40の結果をプロットした図であり、(b)は非重複のデータセットで d が50, 60, 70, 80の結果をプロットした図であり、図中の丸印は $f'(X_{1,d,w})$ の平均値 $\mu_{1,d,w}$ 、その上下にある横線が信頼区間 $[c_{1,d,w}^+, c_{1,d,w}^-]$ を意味している。図5より、 $d \geq 40$ の場合、 w が増加すると $\mu_{1,d,w}$ は減少するが、 $d < 40$ だとその傾向が確認できない。Word2Vecを英文に用いる場合において、 w が増加すると学習量が増えるため、結果的にベクトルの質は向上することにつながる事が報告されて

(a) 次元数 $d \in \{10, 20, 30, 40\}$ (b) 次元数 $d \in \{50, 60, 70, 80\}$ 図5 500回の試行における各次元の $f(X_{l,d,w})$ の平均値と信頼区間

いる [4]。しかし、 $d < 40$ でこの傾向が確認できなかったのは、 $d < 40$ では次元数が低すぎるため商品表現する力が弱く、商品の特性が欠損している可能性があると考えられる。 $d \geq 40$ ならば、どの次元数でも問題ないが、類似度計算において次元数は低いほうが計算コストは低くなるため、本実験の環境においては $d = 40$ が最良の次元数といえる。また、 w が増加すると信頼区間の幅が小さくなることが確認できる。これは、 w の

増加にともない $f'(X_{l,d,w})$ のばらつきが小さくなることを意味し、商品の並び方に関わらず $f'(X_{l,d,w})$ の値が安定することを示唆している。したがって、トランザクション数 60 万・商品数 1 万の条件であれば、次元数 $d = 40$ 、ウィンドウサイズ w はできるかぎり大きくするようにパラメータを設定すると、商品情報を失わずにアイテムベクトル学習を行えると考える。

5. おわりに

本研究では、協調フィルタリングで扱うアイテムベクトルを低い計算コストで獲得するために、Word2Vec [4] の商品購買履歴への応用について検討した。先行研究 [5] では、Word2Vec のパラメータやトランザクションデータの与え方について明らかになっていない点があり、本研究ではパラメータやデータの与え方がアイテムベクトルの学習に与える影響を検証した。

検証実験では 1 年分の実 POS データ (トランザクション数 593,793・商品数 10,914) を用いて、Word2Vec のパラメータのウィンドウサイズを 1~10、次元数を 2~100 で 1 ずつ変化させ、さらに、1 トランザクション内での同じ商品の重複の有無や、トランザクション内の商品の順序をランダムに並び替えることを行った。学習したアイテムベクトルの良さを、同じ商品分類の商品が特徴空間上で近傍に出現しやすいことと仮定し、ヒートマップで可視化することで評価した。実験結果より、可能な限りウィンドウサイズは大きく・次元数は 40 以上で設定し、1 トランザクション内の商品の種類に基づいたトランザクションデータ (非重複のデータセット) を用いることで、特徴空間上で同じ分類の商品が近傍に現れやすいようなベクトル学習を行えることを明らかにした。また、本実験ではトランザクションデータを 1~20 倍に増加させて実験を行っているが、ランダムシャッフルのみでトランザクション数を増やさないほうが同じ分類の商品が近傍に現れやすいアイテムベクトルを生成することも確認した。

今後は最適な条件で求めたアイテムベクトルを用いた情報推薦システムを構築していくことを考えている。また、Word2Vec は商品の購買履歴に適用することは可能であるものの、レーティングなどの実数値を含むユーザー-アイテム行列に適用することはできない制約もある。そのため、Word2Vec とは異なったアプローチにより、実数値のユーザー-アイテム行列に対応したアイテムベクトル学習法についても検討する必要があると考えている。

謝辞

本研究は JST 科学技術人材育成補助金テニュアトラック普及・定着事業の支援を受けて実施したものである。

参考文献

- [1] Yehuda Koren, Robert Bell, Chris Volinsky: Matrix Factorization Techniques for Recommender Systems, IEEE Computer, 42(8), 30-37, 2009.
- [2] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, Richard Harshman: Indexing by latent semantic analysis, Journal of the American Society of Information Science, 41(6), 391-407, 1990.

- [3] Daniel D. Lee, H. Sebastian Seung: Learning the parts of objects by non-negative matrix factorization, *Nature*, 401, 788–791, 1999.
- [4] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, Jeffrey Dean: Distributed Representations of Words and Phrases and their Compositionality, *Neural Information Processing Systems* 2013, 3111–3119, 2013.
- [5] Oren Barkan, Noam Koenigstein: Item2Vec: Neural Item Embedding for Collaborative Filtering, *arXiv*, arXiv:1603.04259, 2016.
- [6] 日本経済新聞社：日経商品ディクショナリー 商品分類コードブック, 2015.

(2016 年 11 月 12 日 受付)
(2016 年 12 月 9 日 採録)

[問い合わせ先]

〒182-8585 東京都調布市調布ヶ丘 1 丁目 5 番地 1

電気通信大学大学院情報理工学研究科

岡本 一志

TEL: 042-443-5280

E-mail: kazushi@uec.ac.jp

—— 著 者 紹 介 ——

なつこ
名渡山 夏子 [非会員]

2015 年 沖縄工業高等専門学校メディア情報工学科卒業。2017 年、電気通信大学情報理工学部総合情報学科卒業。同年、クックパッド株式会社入社。卒業研究では情報推薦の研究に従事。

おかもと かずし
岡本 一志 [正会員]

2006 年 高知工科大学工学部情報システム工学科卒業。2008 年 同大学院工学研究科基盤工学専攻修士課程修了。2011 年 東京工業大学大学院総合理工学研究科知能システム科学専攻博士後期課程修了。同年、千葉大学アカデミック・リンク・センター特任助教。2015 年 電気通信大学大学院情報理工学研究科総合情報学専攻助教。博士（工学）。データサイエンスの研究に従事。電子情報通信学会、情報処理学会、日本知能情報ファジィ学会各会員。

Word2Vec Based Item Vector Learning from Purchase Data

by

Natsuko NADOYAMA and Kazushi OKAMOTO

Abstract:

Word2Vec, a distributed representation method in natural language processing, is applied to purchase data in order to achieve item vector learning with low-computational cost. We perform an experiment with real POS data, and it validates how window size and dimension parameters and input purchase data format affect item vector learning. The experimental results suggest that learned item vectors within same category are located neighborhoods on the feature space under the following conditions: window size is as large as possible; dimension is more than 40; input data format is based on item variation.

Keywords: recommender system, collaborative filtering, purchase data, item vector, distributed representation

Contact Address: **Kazushi OKAMOTO**

Graduate School of Informatics and Engineering, The University of Electro-Communications

1-5-1, Chofu-ga-oka, Chofu-city, Tokyo 182-8585, Japan

TEL: +81-42-443-5280

E-mail: kazushi@uec.ac.jp