距離と属性を制約としたPrefixSpanによる感情表現抽出

佐藤　一誠† 　平手　勇宇†† 　山名　早人††,†††

† 早稲田大学理工学部
†† 早稲田大学大学院理工学研究科
††† 国立情報学研究所
E-mail: †{issei,hirate}@yama.info.waseda.ac.jp, ††yamana@waseda.jp

あらまし　シーケンシャルパターンマイニングをテキストデータに適用させる場合，アイテムの出現頻度の制約だけでは，ユーザーにとって興味の無いパターンが大量に抽出されてしまう．本稿では，PrefixSpan を改良し，次の 2 つの制約を加えることで，ユーザーの嗜好を反映したパターンを抽出する手法を提案する．2 つの制約とは，(1) アイテム間の距離 (アイテム数) の制約と (2) アイテムの属性の制約である．(1) は，距離情報を含む PrefixSpan を提案することで実現した．(2) は，シーケンスを，アイテムと属性のタプルと定義することで実現した．また，本提案手法を，Amazon レビューページに適用し，(名詞，形容詞) のような感情表現となりうるパターンの抽出を試みた．その結果，距離の制約 (最大値) が 30 までは，従来手法よりも早くパターンを抽出できることを確認した．さらに，従来手法では抽出できなかったパターンを 1.56 倍多く抽出し，従来手法で抽出されてしまうユーザーにとって興味のないパターンを 1/250 ほど減少させて抽出することに成功した。

キーワード　データマイニング，シーケンシャルパターンマイニング，PrefixSpan，感情表現抽出

Sentiment Mining using PrefixSpan constrained by Item Interval and Item Attribute

Issei SATO†, Yuu HIRATE††, and Hayato YAMANA††,†††

† Faculty of Science and Enginnering, Waseda University
†† Graduate School of Science and Enginnering, Waseda University
††† National Institute of Infomatices
E-mail: †{issei,hirate}@yama.info.waseda.ac.jp, ††yamana@waseda.jp

Abstract　When applying Sequential Pattern Mining (SPM) to text data, a huge number of uninteresting patterns for user is extracted by SPM constrained only by the number of occurences in sequence database. In this paper, we propose the new SPM scheme to extract interesting patterns for user by the following two constraints on PrefixSpan. The two constraints are (1) the constraint of item interval and the constraint of item attribute. (1) is accomplished by proposing the method extracting patterns including item interval. (2) is accomplished by defining a sequence as an ordered list composed of tuples of item and its attribute. Using Amazon customer reviews, we confirmed that our method is able to extract patterns faster than the existing method and to exclude uninteresting patterns keeping interesting patterns.

Key words　Data Maining, Sequential Pattern Mining, PrefixSpan , Extraction of sentiment expression

1. はじめに

高速ネットワーク技術と安価な大容量記憶装置の発達により，膨大なテキストデータが Web や企業に蓄積されるようになってきた．膨大なテキストデータから有用な情報を獲得するための手段としてデータマイニングが注目を集めている．

データマイニングの分野の１つにデータベース中に頻出するアイテムの組合せを，順序を考慮して効率よく抽出するシーケンシャルパターンマイニングという手法がある [1]．PrefixSpan [2] はシーケンシャルパターンマイニングの代表的なアルゴリズムの１つである．シーケンシャルパターンマイニングが提案された当初は，データベースにおけるパターンの出現頻度が，最小

サポート値と呼ばれる任意の整数よりも大きいパターンを抽出するという単純なものであった．しかし，出現頻度の制約だけでは，ユーザーにとって興味の無いパターンも大量に抽出されてしまうため，近年，様々な制約を付加する手法が提案されている [3] [4] [5] [6] [7] [8] [9] [10].

テキストデータに対しシーケンシャルパターンマイニングを適用させる場合も同様に，出現頻度の制約だけでは，以下の 2 つの問題点が生じる可能性がある．

（1） アイテム (単語) 間の距離 (アイテム数) が離れすぎていて，アイテム間に関連性のないパターンが抽出されてしまう．

（2） 記号や助詞など，単独では意味をなさない要素のみで構成されるパターンが大量に抽出されてしまう．

よって，本提案手法は，(1)(2) の問題を解決することを目的とする．その実現方法として，PrefixSpan [2] を改良し，出現頻度の制約に加えて,「アイテム (単語) 間の距離 (アイテム数)」と「アイテム (単語) の属性 (品詞)」を制約として付加する手法を提案する．

本提案手法は，アイテム (単語) 間の距離 (アイテム数) 情報を含めてパターンを抽出することで,「アイテム (単語) 間の距離 (アイテム数)」の制約を加えることができる．これにより (1) の問題を解決する．また，アイテム (単語) 間の属性 (品詞) 情報を含めてパターンを抽出することで,「アイテム (単語) の属性 (品詞)」の制約を加えることができる．これにより (2) の問題を解決する．

なお，(1) の問題に対し，係り受け構造を利用する手法もあるが [11] [12]，係り受け解析器に精度が依存し，日本語の文法に正確な文以外では精度が落ちてしまう．日本語の文法に正確でない文にも対応できるように，本提案手法では，アイテム (単語) 間の距離 (アイテム数) を制約として用いる．

本提案手法は，従来手法と比べ，以下の 3 つの点が異なる．

(I) アイテム間の距離 (アイテム数) 情報をパターンとして含みつつ，抽出できるパターン数の減少を抑える．

(II) シーケンスをアイテムと属性のタプルの列と定義することでアイテムの属性を制約とする．

(III) アイテム間の距離 (アイテム数) の制約 (最大値，最小値) は固定せず，パターンを構成するアイテムの属性によって複数の制約距離を設定できるようにする．

(I) について説明する．従来のアイテム間の距離 (アイテム数) 情報を含むパターンの抽出手法では，距離 (アイテム数) の違うパターンを別のパターンとして区別している [3] [4]．距離 (アイテム数) の違うパターンを別のパターンとして区別すると，個々のパターンの出現回数 (サポート値) が減少する．そのため，従来のシーケンシャルパターンマイニング手法と比べると，同じ出現回数 (サポート値) では，抽出できるパターン数が減少してしまう．

本提案手法では、従来のシーケンシャルパターンマイニング手法 で抽出されるパターンに，アイテム間の距離情報を含めることができる．よって，同じ出現回数 (サポート値) では，従来の距離情報を含むパターンの抽出手法よりも多くのパターンを抽出できる。

(II) について説明する．アイテムの属性を制約とする提案は、すでにされている [7] [8]．本提案手法では，シーケンスをアイテムと属性のタプルの列と定義することで，属性を含むパターンを抽出している点で従来手法とは異なる．属性を含むパターンの抽出によって，特定の属性のみで構成されるパターンの抽出が可能となる．

(III) について説明する．従来手法では、距離の制約 (最大値，最小値) は、どのアイテム間でも同じ値である．例えば，距離の制約として最大値 5 と決定すると，シーケンス中のどのアイテム間でも最大値 5 という制約が均一に課せられる．しかし，ユーザーが，複合名詞+形容詞のパターンを抽出したい場合は，名詞 [0] 名詞 [0-5] 形容詞のように制約をアイテム間で可変にする必要がある ([x] はアイテム間の距離 (アイテム数) を表し，[x-y] はアイテム間の距離 (アイテム数) が x 以上 y 以下であることを示す)．従来手法では，このようなパターンに特化した抽出を行うことができない．

本提案手法は，隣り合うアイテムの属性によって，距離の制約 (最大値，最小値) を変えることができる点で従来手法とは異なる．

本稿では，さらに、予備実験として，本提案手法を Amazon [23] のレビューページへ適用し感情表現となりうるパターンの抽出を試みた．

以下, 第 2 節では，関連研究について述べる．第 3 節では，提案手法についての説明を述べる．第 4 節では，Amazon レビューへの適用例を述べる．第 5 節では，本提案手法の評価実験について述べる．最後に, 第 6 節で，本稿のまとめと今後の課題を述べる．

2. 関 連 研 究

2.1 アイテム間の距離 (アイテム数) を制約とする PrefixSpan

PrefixSpan [2] は，アイテム間の距離 (アイテム数) を区別せず同じパターンとみなす．アイテム間の距離が離れているパターンも同一視するため，関連性の無いパターンが抽出される可能性がある．そこで，アイテム間の距離を制約とし，距離の違うパターンを別のパターンとして区別することで，距離情報を含み，関連性の無いパターンの抽出を抑える手法が提案されている [3] ．

しかし，従来手法では，距離の違うパターンを異なるパターンとして区別しているため，個々のパターンの出現回数 (サポート値) が減少し，抽出できるパターン数が減少してしまう．例えば，$< f[1]a >$ が 7 回, $< f[2]a >$ が 4 回出現し, 出現回数が 10 回以上のものを抽出するとした場合，PrefixSpan では同一視するため，$< f, a >$ の出現回数が 合計の 11 回となり抽出されるが，距離の違うパターンを異なるパターンとして区別すると，$< f[1]a >, < f[2]a >$ ともに抽出されなくなってしまう．

この問題に対し，最大誤差数というパラメータを導入してパターンの抽出量を増加させる手法も提案されている [4]．しかし，PrefixSpan と比べると抽出量は少ない．また，距離が範囲として扱われるので，アイテム間の正確な距離情報が失われて

しまう．

本稿では，PrefixSpan と同等の抽出量で，アイテム間の正確な距離情報を含むパターンの抽出手法を提案している点で従来手法とは異なる．また，従来手法では，距離の制約 (最大値，最小値) は固定値で，どのアイテム間も同一の制約値を課せられるが，本提案手法では，隣り合うアイテムの属性によって，制約 (最大値，最小値) を変化させている点で異なる．

2.2 アイテムの属性を制約とする Prefixspan

ユーザーにとって必要なパターンのみを抽出するために，アイテムの属性を制約とする提案がされている [7] [8]．PrefixSpan の射影条件として，属性を制約としている．

2.3 感情表現抽出

一般消費者や顧客の評判・意見などをテキストデータから抽出するために，テキストデータからの感情表現抽出の研究が行われている [17] [15] [18] [14]．一般に，感情表現抽出の研究の多くは，単語に含まれる感情極性 (プラス (ポジティブ) イメージ／マイナス (ネガティブ) イメージ) を抽出することを 1 つの目的としている．感情極性は，単語の組合せによって変化し，単体で規定できるものではない．例えば，「泣く」という単語は，「映画 A で泣いた」のように，単語の組合せ（映画，泣く）によって，映画の評価や意見としてはプラスイメージになる．このような単語の組合せ (複合語) によって感情極性の変化を抽出する研究が行われている [19] [20] [21] ．従来の研究では，単語の組合せパターンを抽出する手法として，人手による抽出や単語 n-gram 統計を用いた抽出を行っている．しかし，人手による抽出はコストがかかる．単語 n-gram 統計では，n の値が多くなればなるほど，計算コストが高くなる．また，連続な単語列のパターンしか抽出できない．

シーケンシャルパターンマイニングを用いたテキストからのパターン抽出の研究も行われている [11] [13]．係り受け解析器を用いてテキストを木構造の構造化データとし，PrefixSpan を構造化データへ適用できるように拡張することで，係り受け関係を考慮したパターンの抽出を行っている．係り受け関係を含めたパターンの抽出ができるので言語的に意味の無いパターンの抽出を抑えるのに効果的である．しかし，日本語として正しい構文であるような文でなければ精度が落ちてしまう．

3. 提 案 手 法

本節では，提案手法の概要について述べる．

3.1 提案手法の概要

テキストデータに対しシーケンシャルパターンマイニングを適用させる場合，出現頻度の制約だけでは，以下の 2 つの問題点が生じる可能性がある．

（1） アイテム (単語) 間の距離 (アイテム数) が離れすぎていて，アイテム間に関連性のないパターンが抽出されてしまう．

（2） 記号や助詞など，単独では意味をなさない要素のみで構成されるパターンが大量に抽出されてしまう．

アイテム (単語) 間の距離 (アイテム数) が離れすぎているパターンは，頻出であってもアイテム (単語) 間に関係が無い可能性がある．

また，記号や助詞など，単独では意味をなさない要素のみで構成されるパターンのような，ユーザーにとって興味の無い大量のパターンの発生は，ユーザーの求める特定のパターンを埋もれさせてしまい，有用な情報を得る妨げとなる．

よって，本提案手法は，(1)(2) の問題を解決し，ユーザーの嗜好を反映したパターンを抽出することを目的とする．その実現方法として，PrefixSpan [2] を改良し，出現頻度の制約に加えて，「アイテム (単語) 間の距離 (アイテム数)」と「アイテム (単語) の属性 (品詞)」を制約として付加する手法を提案する．

3.2 従来手法との比較

本提案手法は，従来手法と比べ，以下の 3 つの点が異なる．

(I) アイテム間の距離 (アイテム数) 情報をパターンとして含みつつ，抽出できるパターン数の減少を抑える．

(II) シーケンスをアイテムと属性のタプルの列と定義することでアイテムの属性を制約とする．

(III) アイテム間の距離 (アイテム数) の制約 (最大値，最小値) は固定せず，パターンを構成するアイテムの属性によって複数の制約距離を設定できるようにする．

(I) について説明する．従来のアイテム間の距離 (アイテム数) 情報を含むパターンの抽出手法では，距離 (アイテム数) の違うパターンを別のパターンとして区別している [3] [4]．距離 (アイテム数) の違うパターンを別のパターンとして区別すると，個々のパターンの出現回数 (サポート値) が減少する．そのため，従来のシーケンシャルパターンマイニング手法と比べると，同じ出現回数 (サポート値) では，抽出できるパターン数が減少してしまう．

本提案手法では、従来のシーケンシャルパターンマイニング手法 で抽出されるパターンに，アイテム間の距離情報を含めることができる．よって，同じ出現回数 (サポート値) では，従来の距離情報を含むパターンの抽出手法よりも多くのパターンを抽出できる。

(II) について説明する．アイテムの属性を制約とする提案は、すでにされている [7] [8]．本提案手法では，シーケンスをアイテムと属性のタプルの列と定義することで，属性を含むパターンを抽出する．属性を含むパターンの抽出によって，属性を制約とするシーケンシャルパターンマイニングが可能となる．

(III) について説明する．従来手法では、距離の制約 (最大値，最小値) は、どのアイテム間でも同じ値である．

本提案手法は，隣り合うアイテムの属性によって，距離の制約 (最大値，最小値) を変えることができる点で従来手法とは異なる．

3.3 シーケンシャルパターンマイニング [1] [2]

提案手法の説明の準備として，シーケンシャルパターンマイニングについて説明する．

3.3.1 シーケンス

$I = \{i_1, i_2, \cdots, i_n\}$ をアイテム集合とする．シーケンスとは，順序を持つアイテムの列である．シーケンス s を $s =< i_1, i_2, \cdots, i_l >$ と表記する．$i_k (k = 1, 2, \cdots, l)$ は任意のアイテムである．

シーケンス中のアイテムの個数をシーケンスの長さとする．

シーケンス $s =< i_1, i_2, \cdots, i_l >$ の長さは l であり，長さ l のシーケンスを l-シーケンスとする．

あるシーケンスα中のすべてのアイテムが，別のシーケンスβ中に存在し，その順序も保持している場合，αをβのサブシーケンスと呼び，βをαのスーパーシーケンスと呼ぶ．αとβの関係を$\alpha \sqsubseteq \beta$と表記する．

3.3.2 シーケンスデータベース

シーケンスデータベース S とは，シーケンス ID(sid) とシーケンス s のタプル $< sid, s >$ の集合である．

$$S = \{< sid_1, s_1 >, < sid_2, s_2 >, \cdots, < sid_n, s_n >\}$$

3.3.3 サポート値

シーケンスαのシーケンスデータベース S におけるサポート値とは，S 中のすべてのシーケンスのうち，シーケンスαを含むタプルの数である．

$$support_S(\alpha) = \| \{< sid, s > | < sid, s > \in S \wedge \alpha \sqsubseteq s\} \|$$

3.3.4 シーケンシャルパターンマイニングの問題定義

シーケンシャルパターンマイニングとは，シーケンスデータベース S から，最小サポート値と呼ばれる任意の正の整数 ζ に対し，$support_S(\alpha) \geqq \zeta$ となるような シーケンスαを全て抽出する問題である．$support_S(\alpha) \geqq \zeta$を満たすシーケンス$\alpha$を (頻出) シーケンシャルパターンと呼び，長さ lの (頻出) シーケンシャルパターンを $l-$ (頻出) シーケンシャルパターンと呼ぶ．

3.4 PrefixSpan [2]

PrefixSpan は，2001 年に Pei らによって提案されたシーケンシャルパターンマイニングのアルゴリズムである [2]．PrefixSpan は，Prefix projection という射影方法とその射影によって生成される射影データベースを用いて深さ優先的にマイニングを行うアルゴリズムである．

まず，PrefixSpan の説明にあたって必要な Prefix，Postfix，射影 (Prefix projection) などの用語について説明する．次に，アルゴリズムについて説明する．

3.4.1 Prefix と Postfix

あるシーケンス $s =< i_1, i_2, \cdots, i_n >$, アイテム a に対し，$i_1 \neq a, i_2 \neq a, \cdots, i_{m-1} \neq a, i_m = a$ となるような正の整数 $m(\leqq n)$ が存在すると仮定する．このとき，シーケンス $< i_1, i_2, \cdots, i_m >$ を s の a に対する Prefix とする．また，シーケンス $< i_{m+1}, i_{m+2}, i_n >$ を s の a に対する Postfix とする．もし，m が存在しないときは，Prefix,Postfix は未定義とする．

3.4.2 射影 (Prefix projection)

あるシーケンスデータベース S に対し，アイテム a によって射影するとは，シーケンスデータベース S 中の各 sid 毎に，シーケンス s に対する a の Postfix を作成し，その Postfix を改めてシーケンスデータベースとする操作である．また，このようにして作成されたシーケンスデータベースを射影データベースと呼ぶ．アイテム a でシーケンスデータベース S を射影して作成された射影データベースを $< a >$ 射影データベースと呼び，$S_{<a>}$ と表記する．

射影と射影データベースの例を 図 1 に示す．図 1 中のシーケンスデータベース S を a で射影すると，各 sid の Postfix は，$sid = 10$ では $< c, d >$, $sid = 20$ では $< b, c >$, $sid = 30$ では無し, $sid = 40$ では $< a, b >$ となる．よって，これらが射影データベースとなる．

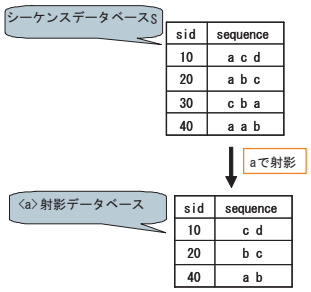

図 1　射影の動作例

3.4.3 PrefixSpan のアルゴリズム

PrefixSpan はシーケンスデータベースに対し，深さ優先探索で射影を繰り返し，頻出シーケンシャルパターンを抽出する手法である．基本的な流れを以下示す．

（1） 長さ 1 の頻出シーケンシャルパターンをシーケンシャルデータベースから抽出する．すべての頻出シーケンシャルパターンは，この長さ 1 の頻出シーケンシャルパターンを Prefix とするシーケンシャルパターン (部分シーケンシャルパターン) から成る．

（2） 各々の部分シーケンシャルパターンを抽出する．部分シーケンシャルパターンを，それぞれ対応する長さ 1 の頻出シーケンシャルパターンで射影した射影データベースから再帰的に抽出する．

3.5 提案手法

シーケンシャルパターンマイニングの代表的な手法の 1 つである PrefixSpan [2] を拡張し，アイテム間の距離 (アイテム数) とアイテムの属性を制約とするシーケンシャルパターンマイニング手法を提案する．以下，提案手法の各用語について述べる．次に，アルゴリズムを説明する．

3.5.1 用語定義

■射影 Level

射影 Level k とは，k-頻出シーケンシャルパターンによるシーケンスデータベースの射影である．まだ射影されていない初期のデータベースを射影 Level 0 で射影されたデータベースとみなす．

■ sid-pos 連想配列

sid-pos 連想配列とは，アイテムを含むシーケンスの sid と対象とするシーケンス中での最初の出現位置 (pos) を，キー=sid, 値= pos とした連想配列であり，sid → pos と表記する．なお，出現位置 (pos) の値は，シーケンスの先頭を 1 とする．また，あるアイテムに対する sid-pos 連想配列の集合を sid-pos 集合とし，$\{sid \rightarrow pos\}$ と表記する．図 2 の (II) に具体例を示す．

■ k-頻出パターンデータベース

k-頻出パターンデータベースとは，シーケンスデータベース中

での最小サポート値以上の頻出アイテム item とその item の sid-pos 集合のタプル $< item, \{sid \rightarrow pos\} >$ の集合である．

図 2 の (III)) に具体例を示す．

■射影アイテムデータベース

射影アイテムデータベースとは，射影したアイテム proj-item とその proj-item の sid-pos 集合のタプル $< proj-item, \{sid \rightarrow pos\} >$ の集合である．図 2 の (IV)) に具体例を示す．

■アイテムの属性

本論文では，テストデータを対象としたアイテムの属性として，「品詞」をアイテムの属性とする．

3.5.2　提案手法のアルゴリズム

以下，提案手法のアルゴリズムを説明する．

（1） シーケンスをアイテムとその属性のタプルの列とする．

（2） 射影 Level (k-1) で射影されたデータベースから，頻出アイテムを抽出する．

（3） 次のすべての制約条件を満たすアイテムのみ k-頻出パターンデータベースへ追加する．

（a） 射影 Level k で指定された属性を持つアイテム

（b） 射影 Level k で指定された距離を満たすアイテム

（c） $k \geqq 2$ ならば，射影アイテムデータベース中でもっとも新しく追加されたアイテムの属性に応じて，制約として指定された距離 (最大値，最小値) を満たすアイテム

制約を満たす頻出アイテムがない場合は，射影 Level(k-1) の射影は終了する．

（4） k-頻出パターンデータベースから辞書順にアイテムを選び射影する (属性 p のアイテム i で射影したとする)．

（5） アイテム i を射影アイテムデータベースへ加える．

（6） 距離情報を含むパターンを抽出したい場合は，射影アイテムデータベースをもとに出力する．

（7） 属性 p のアイテム i で射影されたデータベースを射影 Level k で射影されたデータベースとし，制約をもとに (k+1)-頻出パターンデータベースを作成する．

深さ優先的に k=1,2,3$\cdots$ と上記を繰り返す．

3.5.3　提案手法の動作例

図 2，3 を用いて提案手法の動作例を説明する．

アイテムの集合 $I = \{a, b, c, d, e\}$，属性の集合 $P = \{p_1, p_2, p_3, p_4, p_5\}$，アイテムと属性の対応をそれぞれ $a - p_1$, $b - p_2$, $c - p_3$, $d - p_4$, $e - p_5$ とする．

抽出条件として，属性が，p_1，$(p_3 \mid p_5)$，任意の属性 (以下* と表す) という順番になるシーケンスを抽出するものとする．また，属性 p_1 と $(p_3 \mid p_5)$ の間の距離 (アイテム数) は 0 であり，$(p_3 \mid p_5)$ と任意のアイテムの間の距離 (アイテム数) は，0 以上 1 以下であるパターンを抽出するものとする．最小サポート値は，2 とする．

抽出条件を踏まえると，距離と属性の制約を次のように設定する．

（1） 射影 Level1 は，属性 p_1 のみ射影する．

（2） 射影 Level2 は，属性 $(p_3 \mid p_5)$ のみ射影する．

（3） 射影アイテムデータベースに最も新しく登録されたアイテムの属性が p_1 であり，頻出パターンデータベースに登録するアイテムの属性が $(p_3 \mid p_5)$ であるならば，距離 (アイテム数) の制約として最大値 0 を設定する．

（4） 射影 Level2 の射影後は (射影アイテムデータベースの登録アイテム数が 2 の場合は)，距離 (アイテム数) の制約として最小値 0，最大値 1 を設定する．

なお，今回の場合，(3) の制約は，「射影 Level1 の射影後は (射影アイテムデータベースの登録アイテム数が 1 の場合は)，距離 (アイテム数) の制約として最大値 0 を設定する」としても同じである．また，(4) の制約は，「射影アイテムデータベースに最も新しく登録されたアイテムの属性が $(p_3 \mid p_5)$ であるならば，距離 (アイテム数) の制約として最小値 0，最大値 1 を設定する」としても同じである．

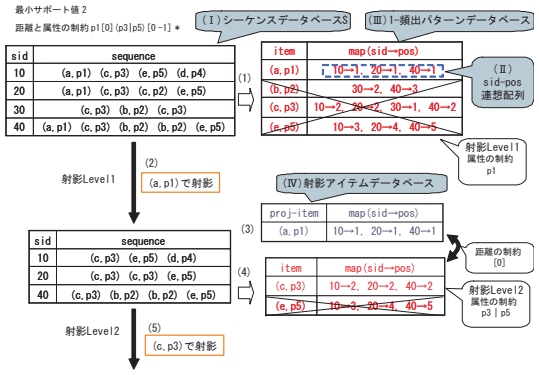

図 2　アイテムの距離と属性を制約としたシーケンシャルパターンマイニング手法の動作例

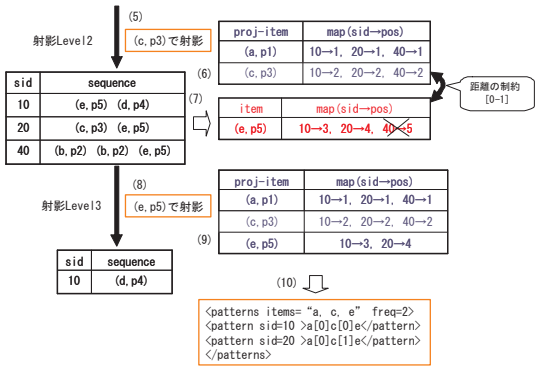

図 3　アイテムの距離と属性を制約としたシーケンシャルパターンマイニング手法の動作例

（1）　シーケンスデータベース S から，最小サポート値以上であり，制約 (1) より属性が p_1 である頻出アイテムを抽出し，1-頻出パターンデータベースを作成する (図 2 の (1))．(a, p_1) で射影する (図 2 の (2))．射影したアイテムを射影アイテムデータベースへ登録する (図 2 の (3))．

（2） (a, p_1) で射影して得られた射影データベースから，最小サポート値以上であり，制約 (2) より属性が $(p_3 \mid p_5)$ であり，さらに制約 (3) より距離の制約を満たす頻出アイテムを抽出し，2-頻出パターンデータベースを作成する (図 2 の (4))．例えば，アイテム e は，アイテム a との距離が各 sid で 1 以上であるの

で，それぞれサポート値としてカウントない．よって，アイテム e のサポート値は最小サポート値未満になり，アイテム e は 2-頻出パターンデータベースへ登録さない．(c, p_3) で射影する (図 3 の (5))．射影したアイテムを射影アイテムデータベースへ登録する (図 3 の (6))．

（3） (c, p_1) で射影して得られた射影データベースから，最小サポート値以上であり，制約 (4) より距離の制約を満たす頻出アイテムを抽出し，3-頻出パターンデータベースを作成する (図 3 の (7))．例えば，sid=40 のアイテム e は，アイテム c との距離が 2 であるのでサポート値としてカウントされない．よって，アイテム e の sid → pos 連想配列 40 → 5 は，2-頻出パターンデータベースへ登録さない．(e, p_5) で射影する (図 3 の (8))．射影したアイテムを射影アイテムデータベースへ登録する (図 3 の (9))．

（4） (e, p_5) で射影した射影データベースには，頻出アイテムは無いので，射影は終了する．ここで，距離情報を含むパターンを抽出する (図 3 の (10))．まず，アイテム e のキー (sid) を取り出し，キーからアイテム e の pos，アイテム a，c の pos を出力し，距離アイテム間の距離 (アイテム数) を計算する．アイテム a の連想配列を，a[Key] などとすると，a[10]=1，c[10]=2，e[10]=3 より，アイテム a とアイテム c の間には 2-1-1=0 個のアイテムが存在するので，距離 0 と計算できる． アイテム c とアイテム e の間には 3-2-1=0 個のアイテムが存在するので，距離 0 と計算できる． 同様にして e の全てのキーに対して行う．

（5） 射影 Level3 では，射影するアイテムはもうないので，射影 Leve2 で射影できるアイテムを頻出パターンデータベースから探す．射影 Level2 で射影するアイテムはもうないので，射影 Leve1 で射影できるアイテムを頻出パターンデータベースから探す．射影 Level1 でも射影できるアイテムは，無いのでここでパターンの抽出は終了する．

4. Amazon カスタマーレビューを用いた抽出実験

本節では，Amazon [23] カスタマーレビューから実際に抽出された例を示す．

Amazon カスタマーレビューを使う利点を次に示す．

（1） 書籍，映画，家電製品など複数のカテゴリーに分かれているのでカテゴリーに特有の表現が抽出できる．

（2） レビューに 5 段階の評価値がついているので，レビューの評価を人手により分類しなくてよい．

（3） Web サービス [24] を提供しているのでレビューの収集が容易に行える．

4.1 データセット

本提案手法を評価するためのデータセットを以下の手順で生成した．

（1） Amazon の本カテゴリーの評価値 5(5 つ星) と評価値 1(1 つ星) から，2005 年 12 月 01 日にレビューを収集した．

（2） 対象とする書籍は，2005 年 12 月 01 日時点で収集可能な最大数 32,000 冊の書籍である．ただし，レビューを含まない書籍も含まれる．

（3） 形態素解析器 Mecab [22] により，形態素解析を行い，1 アイテムを 1 単語 (形態素)，1 レコードを 1 文とした．ただし，1 文とは，「。」,「.」,「？」の文の区切りを表す単語と，逆接を表す単語 (しかし，けれども，が，だが，ところが，それなのに，にもかかわらず) により区切られた文である．なお，文の区切りを表す単語や逆接を表す単語も 1 文中 (文の区切りを表す単語は前の文，逆接を表す単語は後の文) に含まれる．

（4） 各星ごとにレビュー数の偏りが大きいので，1 つ星は 32,000 冊 (レビューを含まない書籍も含まれる) 分すべてのレビューを使い，5 つ星はセールス順に高いほうから 10,000 冊分のレビューを用いた．

（5） 収集できた文は，星 1 つのレビューでは 37,672 レコード (文)，星 5 つのレビューでは 301,956 レコード (文) である．

4.2 抽 出 例

感情表現のパターンとして，最も一般的な < 名詞，形容詞/形容動詞 > というパターンに着目する．まず，パターン < 名詞，形容詞/形容動詞 > のアイテム間の距離の分布を示す．次に，実際の抽出例を示す．なお，ここで形容動詞の語とは，Mecab における形容動詞語幹の語を意味する．

4.2.1 アイテム間の距離情報

本カテゴリーの 5 つ星のレビューから，名詞と形容詞/形容動詞の間の距離 (アイテム数) の統計を取った．アイテム間の距離 (アイテム数) の制約は付加せずに，属性の制約として名詞+形容詞/形容動詞のパターンを抽出するようにした．図 4 は，名詞と形容詞/形容動詞の間の距離 (0 から 20) を横軸として，縦軸にそのパターンの出現頻度をとったグラフである．距離が 1 のとき最大値をとる．距離が 5，6 以降から全体としてなだらかに減衰しつつ，部分としては上下している．

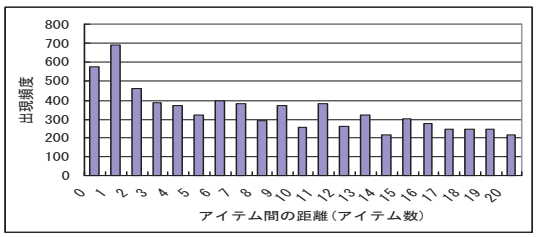

図 4 名詞と形容詞/形容動詞の距離 (0〜20) と出現頻度

4.2.2 抽出されたパターン例

以下，本提案手法の抽出例を紹介する．抽出結果を，いくつかの表にまとめて説明していく．距離の制約は，最大値を 5 とする．属性の制約は，名詞+形容詞/形容動詞のパターンで後に否定語が接続するものは除外してある．+は 5 つ星から得られたパターンを意味し，−は 1 つ星から得られたパターンを意味する．各パターンの左にある数値は，出現頻度である．

表 1，2 は，書籍カテゴリーの 5 つ星，1 つ星のそれぞれの頻度上位のパターンである．

表 3 は，本カテゴリーから「悲しい」を形容詞/形容動詞としてもつパターンである．1 つ星のレビューから「悲しい」を形容詞/形容動詞としてもつパターンは，抽出されなかった．よって，本カテゴリーでは,「悲しい」を形容詞/形容動詞としても

表 1　本カテゴリー 5 つ星の頻度上位のパターン

頻度	名詞	形容 (動) 詞	評価	頻度	名詞	形容 (動) 詞	評価
241	内容	濃い	＋	117	絵	綺麗	＋
217	本	良い	＋	99	レベル	高い	＋
206	奥	深い	＋	92	情報	多い	＋
186	本	好き	＋	91	カッコ	いい	＋
170	本	面白い	＋	90	値段	安い	＋
125	テンポ	良い	＋	85	ストーリー	面白い	＋
119	内容	深い	＋	83	絵	かわいい	＋

表 2　本カテゴリー 1 つ星の頻度上位のパターン

頻度	名詞	形容 (動) 詞	評価	頻度	名詞	形容 (動) 詞	評価
79	頭	悪い	−	34	部分	多い	−
78	人	多い	−	32	気分	悪い	−
76	内容	薄い	−	27	お金	無駄	−
46	レベル	低い	−	26	文章	稚拙	−
39	頭	いい	−	25	絵	下手	−
38	都合	いい	−	22	絵	上手い	−
34	後味	悪い	−	22	気持ち	悪い	−

つパターンは，評価としてプラスになる可能性が高いと考えられる．

表 3 「悲しい」を含むパターン例 (本カテゴリー)

頻度	名詞	形容 (動) 詞	評価	頻度	名詞	形容 (動) 詞	評価
15	人	悲しい	＋	6	人間	悲しい	＋
11	最後	悲しい	＋	5	姿	悲しい	＋
8	ラスト	悲しい	＋	5	自分	悲しい	＋
8	死	悲しい	＋	5	本	悲しい	＋
8	大佐	悲しい	＋				

表 4 は，本カテゴリーから「熱い」を形容詞/形容動詞としてもつパターンである．1 つ星のレビューから「熱い」を形容詞/形容動詞としてもつパターンは，抽出されなかった．よって，本カテゴリーでは,「熱い」を形容詞/形容動詞としてもつパターンは，評価としてプラスになる可能性が高いと考えられる．

表 4 「熱い」を含むパターン例 (本カテゴリー)

頻度	名詞	形容 (動詞) 詞	評価	頻度	名詞	形容 (動詞) 詞	評価
58	胸	熱い	＋	6	気持ち	熱い	＋
40	目頭	熱い	＋	6	体	熱い	＋
20	著者	熱い	＋	5	作品	熱い	＋
17	心	熱い	＋	5	思い	熱い	＋

表 5　は本カテゴリーの「高い」を形容詞/形容動詞として含むものである.「高い」という単語は，単語単体では感情極性を規定できない．レベルと接続するときは評価値が 5 で，リスクと接続するときは評価値が 1 である可能性が高いことが考えられる．

表 5 「高い」を含むパターン例 (本カテゴリー)

頻度	名詞	形容 (動) 詞	評価	頻度	名詞	形容 (動) 詞	評価
99	レベル	高い	＋	0	リスク	高い	＋
11	レベル	高い	−	7	リスク	高い	−

次に複合名詞を含むパターンの例を示す．距離の制約は，名詞間で最大値 0(アイテムなし)，名詞と形容詞/形容動詞の間では，最大値 5 とした．主に，専門用語や固有名詞とのパターンが抽出された．表 6 に，抽出されたパターン例を示す．

表 6 「複合名詞」を含むパターン例 (本カテゴリー)

頻度	名詞		形容詞	評価	頻度	名詞		形容詞	評価
14	具体	例	豊富	＋	4	電車	男	好き	＋
11	好奇	心	旺盛	＋	3	電車	男	リアル	＋
7	ケース	スタディ	豊富	＋	3	電車	男	純粋	＋
4	ケース	スタディ	多い	＋	2	電車	男	いい	＋
2	ケース	スタディ	意外	＋	2	電車	男	かっこいい	＋
2	ケース	スタディ	身近	＋	2	電車	男	すごい	＋
2	ケース	スタディ	徹底的	＋	2	電車	男	素敵	＋

4.2.3　抽出されたパターン数

表 7 に，抽出されたパターン数を示す．なお，アイテム (単語) 間の距離の違うパターンを別のパターンとみなす従来手法と比較すると，提案手法は，従来手法よりも 1.56 倍程度多く抽出されている．つまり，本提案手法は，従来手法では抽出できないパターンを抽出できると言える．

表 7　提案手法と従来手法のパターン数の比較

	提案手法	従来手法
パターン数	3893	2494

5.　提案手法の評価

5.1　評 価 環 境

表 8 に評価環境を示す．

表 8　評 価 環 境

CPU	Intel(R) Pentium(R) 4 2.40GHz
L2 キャッシュサイズ	512 KB
物理メモリサイズ	1GB
OS	RetHat9 カーネルバージョン 2.4.20
コンパイラ	gcc バージョン 3.2.2
プログラム言語	C++

5.1.1　データセット

4.1 の 5 つ星のレビューの 30,000 文をデータセットとした．

5.2　距離を制約とする提案手法の速度評価

最小サポート値をデータベースの全レコード数の 1%とし，シーケンスの長さが 2 以上のパターンを抽出する時間 (ファイルの入出力時間は含まない) を計測した．

図 5 は，距離を制約とする提案手法の距離の制約 (最大値) による速度変化を評価したグラフである．本提案手法は，連想配列を保持するために多くのメモリを消費し，さらに連想配列でのキーから値への探索時間がかかる．よって，本提案手法は距離情報を含まない従来の PrefixSpan よりも速度の面で劣る．

しかし，距離の制約が最大値 25 までは，本提案手法の方が PrefixSpan よりも早い．つまり，距離の制約として最大値 25 以下のパターン抽出では，従来の PrefixSpan よりも高速で，しかもアイテム間の距離情報も含めた抽出が可能である．

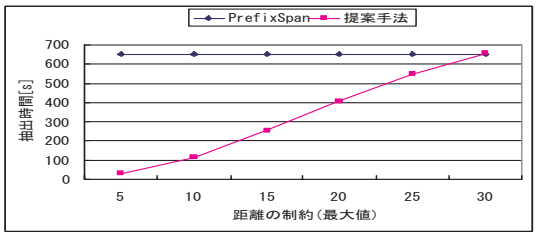

図 5　距離制約の変化による速度評価

5.3　属性を制約とする提案手法の評価

属性を制約とした PrefixSpan の抽出時間と抽出されるパターン数の評価を行った．属性の制約は，ユーザーによって異なるため，厳密な評価はできないが，本稿では，4. 節の Amazon カスタマーレビューを用いた抽出実験での，名詞と形容詞/形容動詞の 2-シーケンスを抽出するための制約を用いて評価を行った．最小サポート値は，4 節と同じ，5 とした．

表 9 に，属性を制約とする提案手法と PrefixSpan のパターン数を示す．従来手法の PrefixSpan に比べ，パターン数は 1/250 ほどに減少させることができた．よって，ユーザーが名詞と形容詞/形容動詞の 2-シーケンスを抽出したい場合，無駄なパターンの抽出を抑え，効率的にパターンを抽出できると言える．

表 9　属性を制約とした提案手法と PrefixSpan のパターン数の比較

	属性を制約とした提案手法	PrefixSpan
パターン数	668	183798

6. おわりに

本稿では，アイテム間の距離とアイテムの属性を制約としたシーケンシャルパターンマイニング手法を提案した．また，本提案手法を，Amazon レビューページに適用し，(名詞，形容詞) のような感情表現となりうるパターンの抽出を試みた．その結果，制約として，アイテム間の距離 (アイテム数) の最大値が 30 までは，従来手法よりも早くパターンを抽出できることを確認した．さらに，従来手法では抽出できなかったパターンを 1.56 倍多く抽出し，従来手法で抽出されてしまうユーザーにとって興味のないパターンを 1/250 ほど減少させて抽出することに成功した。今後の課題を以下に列挙する．

（1） 距離情報を含む PrefixSpan の抽出速度の向上

（2） 抽出された感情表現のパターンに対し，感情極性を決定するための適切なスコア付けの提案と適用

（3） 距離の制約を緩和して得られた距離情報から，距離の制約の最適値を求める手法の提案と適用

（4） 各 Amazon カテゴリーからの感情表現のパターンとその感情極性の抽出

謝　　辞

本研究の一部は，文科省 21 世紀 COE「プロダクティブ ICT アカデミア」及び科学技術振興費「e-Society」プロジェクトによるものである．

文　　献

[1] R.Agrawal and R.Srikant,“ Mining Sequential Patterns, ”In Proc. of ICDE1995, IEEE Press , pp.3-14, 1995.

[2] J.Pei, J.Han, B.Mortazavi-Asl, H.Pnto,Q.Chen, U.Dayal, and M.Hsu,“ PrefixSpan:Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth, ”In Proc. of ICDE2001, IEEE Press , pp.215-224, 2001.

[3] Hajime Kitakami, Tomoki Kanbara, Yasuma Mori, Susumu Kuroki, and Yukiko Yamazaki, “ Modified PrefixSpan Method for Motif Discovery in Sequence Databases, ”In Proc. of PRICAI2002, pp.482-491, Springer Verlag, 2002.

[4] 塔野薫隆, 北上始, 田村慶一, 森康真, 黒木進, “ Modified PrefixSpan 法を用いた頻出正規パターンの抽出を目指して, ”DBSJ Letters, Vol3. no.1, pp.61-64, 2004.

[5] R.Srikant and R.Agrawal, “ Mining Sequential Patterns : Generalizations and Performance improvments, ”In Proc. of EDBT1996, pp.3-17,1996.

[6] M.Garofalakis, R.Rastogi, and K.shim,“ SPIRIT:Sequential pattern mining with regular expression constraints, ”In Proc. of VLDB1999, pp.223-234,1999.

[7] J.pei, J.Han, and W.Wang, “ Mining sequential patterns with constraints in large databases, ”In Proc. of CIKM2002, pp.18-25, 2002.

[8] 山本薫, 工藤拓, 坪井祐太, 松本裕治,“ 系列パターンマイニングによる対訳表現抽出, ” 情処研報（NL）, Vol.2002, No.044, pp.15-22, 2002.

[9] 櫻井 茂明, 植野 研, 酢山 明弘, 折原 良平, “ 時系列イベントパターンマイニングにおける時間制約の導入, ”In Proc. of DEWS2005, 6C-o1, 2005.

[10] 平手 勇宇, 小松 俊介, 山名 早人, “ イベント発生間隔を考慮したシーケンシャルパターンマイニング, ”情報研報 (DBS),Vol.2005, No.137, pp.321-328, 2005.

[11] 工藤拓, 山本薫, 坪井祐太, 松本裕治, “ 言語情報を利用したテキストマイニング, ” 情報研報 (NL), Vol.2002, No.020, pp.65-72, 2002.

[12] 工藤 拓, 山本 薫, 坪井 祐太, 松本 裕治,“ テキストデータベースからの構文構造のマイニング, ”情報研報 (ICS), Vol.2002,No.045, pp.139-144, 2002.

[13] 松本 翔太郎, 高村 大也, 奥村 学, ”単語の系列及び依存木を用いた評価文書の自動分類, ” In Proc. of FIT2004,2004.

[14] 藤村滋, 豊田正史, 喜連川優, “ Web からの評判および評価表現. 抽出に関する一考察, ”信学技報 ,Vol.104, No.177, pp.141-146, 2004.

[15] 那須川哲哉, 金山博, “ 文脈一貫性を利用した極性付評価表現の語彙獲得, ” 情報研報 (NL), Vol.2004, No,73, pp.109-116, 2004.

[16] D. Gluhl, L. Chavet, D. Gibson, J. Meyer, P. Pattanayak, A. Tomkins and J. Zien,“ How to build a WebFountain: an architecture for very large-scale text analysis, ”IBM sytems Journal 43(1), pp.64-77, 2004

[17] Jeonghee Yi and Wayne Niblack, “ Sentiment Mining in WebFountain, ”In Proc. of ICDE2005, pp.1073-1083, 2005.

[18] 熊本忠彦, 田中克己, “ Web ニュース記事からの喜怒哀楽抽出, ” 情報研報 (NL), Vol.2005, No.1, pp..15-20, 2005.

[19] 小林のぞみ, 乾健太郎, 松本裕治, 立石健二, 福島俊一,“ テキストマイニングによる評価表現の収集, ” 情報研報 (NL), Vol. 2003, No.23(NL-154), pp.77-84, 2003.

[20] 藤村滋, 豊田正史, 喜連川優, “ 文の構造を考慮した評判抽出手法, ” 電子情報通信学会 第 16 回データ工学ワークショップ (DEWS2005).

[21] 高村大也, 乾孝司, 奥村学,“ 極性反転に対応した評価表現モデル, ” 情報研報 (NL), Vol.2005, No.73, pp.141-148, 2005.

[22] MeCab: Yet Another Part-of-Speech and Morphological Analyzer, http://chasen.org/ taku/software/mecab/

[23] Amazon, http://www.amazon.co.jp/

[24] Amazon Web Services, http://www.amazon.co.jp/exec/obidos/subst/associates/join/webservices.html/