6 自然言語処理におけるマイニング技術の応用

工藤 拓（NTT コミュニケーション科学基礎研究所）
taku@cslab.kecl.ntt.co.jp

新保 仁（奈良先端科学技術大学院大学情報科学研究科）
shimbo@is.naist.jp

本稿では，これまでの統計的自然言語処理において扱われた各種のタスクをデータマイニング手法に基づいて解釈し直し，その問題点を改善する我々のグループの最近の試みについて紹介する．

取り上げる分野は，(1) 機械翻訳のための対訳表現抽出，(2) 自然言語データを対象にした SVM 分類器の高速化，および，(3) 文書全体ではなく，より小さな単位を対象とした分類問題のための素性選択，の 3 つである．

系列マイニングによる対訳表現抽出

対訳表現抽出問題

対訳表現抽出は，たとえば，“New York” と “ニューヨーク”，“make a mistake” と “間違う”，といった異なる言語間の対訳表現を発見するタスクである．このタスクが想定する応用に，一般の外国語辞書には載っていない，特定分野における専門用語・表現辞書の自動構築があり，したがってこのタスクは機械翻訳や翻訳支援システムの基盤技術の 1 つをなす．手がかりとしては，文単位で対応づけされた，対訳コーパス (parallel corpus) が用いられる．このため，結局のところ対訳表現抽出は，これらの対訳文の中から相関する表現対を見つけるという問題となる．

このタスクにおいて障害となるのは，

1．対訳表現の単位として，単純な，単語どうしの 1 対 1 対応のみを考えるのでは不十分なこと (上記の “make a mistake”)．コロケーションは逐語訳すると外国語では意味が通じない表現となることは広く知られている－たとえば，“間違いを作る” は正しい表現ではない．

2．表現が連続した単語列からなるとは限らないこと (“as … as possible” ↔ “できるだけ”)

といった点である．過去の研究では，まず単言語ごとに共起頻度の高い単語列 (コロケーション) を取り出し，その後で (これら単言語コロケーション集合の直積で与えられる) 2 言語に跨る表現対の集合から，一定の統計的な尺度に基づき相関の高いものだけを改めて列挙する手法がとられてきた．これは主として，対訳表現抽出問題に先行して単言語におけるコロケーション同定問題への取り組みがあり，そのためのシステムが存在していた，という歴史的な理由による．この方法では，非連続表現まで網羅的に考慮しようとすると，2 言語に渡る表現対の組合せの数は膨大なものとなる．したがって，対象を連続表現に限定する，あるいは，一方の言語における表現の候補を何らかの (言語学的な) ヒューリスティックを用いて限定した後，他方の言語における表現との共起を調べる，といった方法がとられてきた．また，対象

を，助詞や助動詞といった機能語を排除して，自立語からなる表現のみに限定するといった手段も用いられていた．こういった方法では，最終的な表現対間の相関算出に用いられる，2言語に跨る共起頻度が単言語の頻出表現の絞り込みの際に考慮されず，本来網羅的に列挙した際に発見されるべき対訳表現対が見落とされてしまう可能性がある．

系列パターンマイニング問題としての定式化

前章の問題に対し，山本ら[3), 4)]は，単言語ごとに個別に頻出表現を取り出すのではなく，対訳文を結合して1本の単語列として扱い，系列パターンマイニング問題として再定式化することを提案した．そのような定式化をした上で，頻出系列パターンマイニングアルゴリズムの一種であるPrefixSpanを適用し，ある一定頻度以上出現する単言語，両言語に跨る単語列(非連続単語列も含む)を高速かつ効率的に列挙することに成功した．

ただし，機能語まで含めた頻出対訳表現の取り出しは，(それ自身が高い頻度を持つ)機能語のみからなる無意味な系列が数多く列挙されることにつながり，たとえPrefixSpanを用いたとしても現実的ではない．この問題に対して彼らは，“射影関数”を，取り出すべき系列パターンを構成する隣接要素(ここでは単語)間の関係を定める述語関数として定義し，この関数に基づいて射影操作が可能か否かを判別するようにPrefixSpanを拡張することで対処している．この拡張により，文中で非連続な機能語間の連接を排除する射影関数を定義することで，機能語を含む表現をある程度取り出すことが可能になった．

たとえば非連続の対訳表現としては，“be staying at … hotel”のような，…部分に異なるホテル名が含まれる文を抽象化し，“ホテルに滞在”という対訳表現を抽出することに成功している．機能語を含む表現としては，他に“impressed with”と“に感銘”，などがあげられている．

SVM分類器の高速化

SVMと自然言語処理

Support Vector Machine (SVM)は，統計的言語処理の精度を飛躍的に向上させた機械学習手法といっても過言ではない．SVMは，マージン最大化による高い汎化能力もその1つの魅力であるが，素性選択のコストを大幅に軽減したことが言語処理に対する貢献として評価されてもよいであろう．

従来の機械学習を用いた自然言語処理では，いわゆる「次元の呪い」の問題から，素性選択を適切に行わないと高い精度が得られなかった．基本素性としてどのような素性集合を与えればよいか問題となるが，多くの複雑な自然言語処理タスクでは，基本素性の組合せを考慮する必要がある．たとえば，係り受け解析タスクでは，係り元と係り先の基本素性(単語，品詞，活用など)の組合せを新たな素性として追加せねばならない．結果として，候補となる素性集合が膨大になり，素性選択の問題をより複雑にしていた．一方，SVMを用いた場合，人手で巧妙に素性選択した場合と比較して，少なくとも同等か場合によってはそれ以上の認識率が得られることが実験的に知られている．また，素性の組合せに関しても，多項式カーネルを用いることで，計算量や一般性を落とすことなく考慮することが可能である．

分類速度の問題とその克服

SVMは万能な手法のように見えるが決してそうではない．一番の問題点は，分類速度の遅さである．たとえば，従来のルールベースの固有表現抽出システムに比べ，SVMとカーネル法を用いたシステムは，数百倍遅いという報告がある．SVMとカーネル法を用いた言語解析システムは，大規模テキストデータの解析を必要とする応用には事実上適用困難であった．工藤らは，この問題を克服すべく，SVMの分類高速化をデータマイニングの技術を用いて実現した[6)]．

高速化手法の概要を以下に示す．カーネルを用いた場合の分類器は以下で与えられる．

$$f(\mathbf{x}) = \mathrm{sgn}\left(\sum_{i=1}^{m} y_i \alpha_i K(\mathbf{x}_i, \mathbf{x}) + b\right)$$

ただし，$T=\{(\mathbf{x}_1, y_1),\ldots, (\mathbf{x}_m, y_m)\}$は訓練事例集合(各$\mathbf{x}_j$が事例の素性ベクトル，$y_j$がそのラベル)，$\alpha_i$ (≥ 0)はSVMによって与えられる重み，bはバイアス項である．また，$K(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle$はカーネル関数であり，$\Phi(\cdot)$は高次元特徴ベクトル空間への写像関数である☆1．カーネルを用いず写像関数$\Phi(\cdot)$のみで分類器を表現すると，主問題の分類器

$$f(\mathbf{x}) = \mathrm{sgn}\left(\langle \mathbf{w}, \Phi(\mathbf{x}) \rangle + b\right)$$

☆1 カーネル関数，写像関数$\Phi(\cdot)$については，本特集の鹿島による記事を参照されたい．

が与えられる．ただし，$\mathbf{w} = \Sigma_{i=1}^{l} y_i \alpha_i \Phi(\mathbf{x}_i)$．高速化の基本的なアイディアは，上記のようにカーネルを用いた分類器を単純な線形分類器に変換することにある．しかし，多項式カーネルを用いた場合，素性の d 個までの部分集合が新たな素性として追加されるため，ベクトル $\mathbf{w}$ はきわめて高次元になり，その構築自身が困難になってしまう．そこで，$\mathbf{w}$ の各次元のうち，その値が0付近のものは分類に寄与しないという近似を行う．

$\mathbf{w}$ の近似ベクトル $\mathbf{w}'$ の構築にAprioriアルゴリズムと同種の手法を用いる．Aprioriが扱う問題は，頻出部分集合の列挙であった．$\Phi(\mathbf{x})$ が事例 $\mathbf{x}$ の部分集合を展開する写像となることに注意すると，近似ベクトル $\mathbf{w}'$ の構築問題は，Aprioriが扱う問題と本質的に同一であることが分かる．おおざっぱには，$y_i \alpha_i$ を事例 $\mathbf{x}_i$ の頻度と見なしAprioriを実行すればよい．ただし，頻度が負の数を含めた実数となることと，枝刈り条件が頻度の絶対値によって決まることが通常のAprioriとの違いとなる．この変更は，Aprioriの枝刈り条件を修正することで実現可能となる．

さらに，同様の手法は，集合だけでなく，系列，木といった複雑な構造に対するカーネル (String，Treeカーネル) についても適用可能である．系列にはPerfixSpanといった頻出部分系列マイニング手法，木にはFREQT[1]といった頻出部分木マイニング手法がそれぞれ適用可能である．

性質の異なる3種類の言語処理タスク (名詞句抽出，日本語わかち書き，係り受け解析) に，SVMの高速化手法を適用したところ，通常の手法と比較して，解析精度を落とすことなく30～300倍の高速化に成功した．我々が開発，公開している日本語係り受け解析「南瓜」☆2には，本手法が組み込まれている．

■文の役割分類への応用

文書分類の多様化

自然言語処理の代表的な応用に文書分類がある．文書分類では，一般に文書を単語の集まり (bag-of-words，BOW) によって表現する．その後，各単語を手がかり (素性，特徴量) とし，SVMといった機械学習手法を用いて文書の内容を政治，経済，スポーツといった粒度の粗い分類クラスに分類する．個々の単語がこれらのカテゴリを特徴付けるのに有効な情報を与えるため，BOWのような単純な素性を用いるモデルによっても高い精度を達成することができる．実用化されたシステムも存在し，一定の成功を収めている☆3．

一方で，テキストマイニングの分野では，Web上の製品レビューサイトやアンケート結果などから製品に対する要望や不満などの有用な情報を効率よく入手する要素技術が求められている．このようなタスクでは，意見が主観的に述べられているのか客観的に述べられているのか，あるいは，ある製品をほめているのかけなしているのかなど，書き手の意図に関する分類が求められる．つまり，このようなタスクは，分類するのは文が漠然と表す意味内容ではなく，文そのものが持つクラスであったり，文書中での文の役割についての分類になる．

評判分類　良い点：メールを送受信した日付，時間が表示されるのも結構ありがたいです．
悪い点：何となく，レスポンスが悪いように思います．

主観性判定　主観的：エンジンパーツが豊富で安い．
主観的以外：パワーが足りないっていう人もいます．

図-1　分類対象文の例

図-1に，想定するタスクで分類すべき文例を示す．**評判分類**はWebの掲示板に投稿された携帯電話のレビューを良い点と悪い点に分類するタスク，**主観性判定**は，車のレビューサイトにおいて投稿者が車に対して下している評価のうち，それが投稿者自身の主観的な評価かそうでないかを分類するタスクの具体例である．「パワーが足りないっていう人もいます．」という文は，製品の評価にはなっているが，本人の評価ではないため，主観的な評価文とは判断していない．

文の意味内容だけでなく発話の意図を量るこのようなタスクでは，単純なBOWではなく，単語のつながりや

☆2 http://chasen.naist.jp/~taku/software/cabocha/

☆3 SPAMフィルタリングのソフトウェアなどは同様の技術を用いて実用化されている．

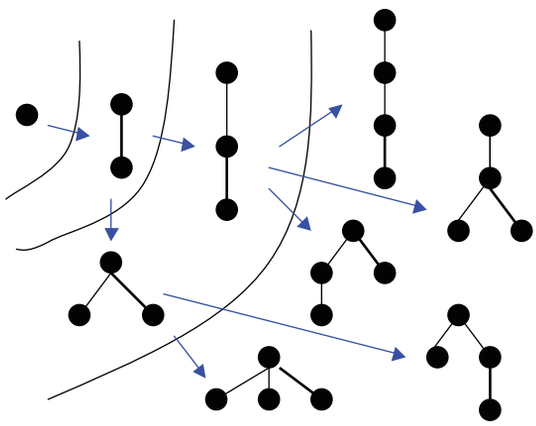

図-2　最右拡張の再帰（探索木）

文体を素性とする必要があるだろう．しかし，どのような単語のつながりや文体が分類に有効な素性となるか事前に分からないことが多い．人手により有効な素性を事前に調査をすることは可能であるが，思いがけない単語のつながりや文体を見落とす可能性がある．また，異なるドメインのテキストやカテゴリに対する頑健性という点からも人手処理には限界がある．データマイニングの立場では，思いがけない単語のつながりや文体こそが人間の意思決定に重要であると考える．そのため，事前に素性を与えるよりは必要となる素性をデータから自動的に発見する方が望ましい．このような素性選択の問題も知識発見という意味でデータマイニングの1つの応用と考えられる．

文の構造を考慮した機械学習手法

工藤らは，データマイニングの手法を用いて自動的に有効な手がかり（素性，特徴量）を発見し，それを用いて分類を行う手法を提案した[7]．従来法との決定的な違いは，文を単語の集合ではなく，木構造として表現する点にある．単語の並び，構文解析木，係り受け木など，自然言語処理において文を木で表現することは一般的な方法である．分類に使う素性は，任意の部分木となる．これは任意の単語のつながりや文体に対応する．

本手法では，学習，分類アルゴリズムの土台としてdecision stumpを用いる．Decision stumpは，1つの素性の有無に基づき分類を行う単純なアルゴリズムである．分類器$h(\mathbf{x})$は具体的には以下のように定義される．

$$h_{\langle t,\, y\rangle}(\mathbf{x}) \stackrel{\text{def}}{=} \begin{cases} y, & t \subseteq \mathbf{x} \text{ の場合} \\ -y, & \text{それ以外の場合.} \end{cases}$$

ただし，$\mathbf{x}$は分類対象の木である．分類器のパラメータは，木tと ラベル $y \in \{\pm 1\}$ の組 (t, y) で与えられる．もし分類対象$\mathbf{x}$が木tを部分木として含む場合は分類結果はyとなり，それ以外は$-y$となる．

Decision stumpの学習は，学習データ $T=\{(\mathbf{x}_i, y_i)\}_{i=1}^{L}$ に対するエラー率を最小にする（正解率を最大にする）最適パラメータ $(\hat{t}, \hat{y})$ を導出することで実現できる．学習自身は一見単純に見えるが，木tの候補には制限を設けず，可能なすべての木を対象としているために，完全列挙に基づく手法は困難となることに注意されたい．この問題は，頻出部分木マイニングアルゴリズムを一部変更することで解決できる．具体的には以下の2ステップの手続きとなる．

1．木の集合から，全部分木を完全に重複なく列挙するための正準的な探索空間を定義する．
2．ステップ1．で構築された空間を探索し，最適パラメータを発見する．このとき，分枝限定法を用いて探索空間を枝刈りする．

ステップ1．に関しては，安部とZakiらが独立に同時期に提案した最右拡張[1), 5)]を用いる．最右拡張では，まず，サイズ1の木が列挙される．そして，サイズ $(k-1)$ の木に1つのノードを追加することでサイズkの木を構築する．この手続きを再帰的に適用することで全部分木を列挙する．しかし，任意の位置にノードを追加すると重複する木を生成してしまうため，ノードは最右枝に末の弟として追加されるという制限を設ける．**図-2**に，最右拡張で定義される探索木の例を示す．手続き2．では，ある部分木tに対し，その全上位木$t' \supset t$の正解率の上限$\mu(t)$を求める．もし，すでに発見された準最適部分木の正解率が$\mu(t)$より大きい場合，tから先の空間は枝刈りできる．

さて，decision stumpだけでは，たった1つの木の有無のみで分類が決定されるので性能が悪い．そこで，decision stumpを弱学習器としてBoostingアルゴリズムを実行し全体の精度を向上させる．Boostingを適用することで，それまでの学習器にとって分離困難な事例に集中して学習した弱学習器が次々に作られる．最終的な文分類は，これらの弱学習器の重み付き多数決によって行われる．

本手法では，精度の高い統合的な分類器が得られるだ

	評判分類	主観性判定
構造なし	76.0	77.4
構造あり	78.7	81.6

表-1　文分類実験の結果（F値）

キーワード	重み λ_t	部分木 t（サポート素性）
A. にくい	0.00402	切れる にくい
	-0.00055	にくい なる た
	-0.00056	にくい.
	-0.00069	読む にくい
	-0.00073	にくい なる
	-0.00076	使う にくい
	-0.00170	にくい
B. 使う	0.00273	使う たい
	0.00015	使う
	0.00013	使う てる
	0.00007	使う やすい
	-0.00010	使う やすい た
	-0.00076	使う にくい
	-0.00085	は 使う づらい
	-0.00188	方 が 使う やすい
	-0.00233	を 使う てる た
C. 充電	0.00280	充電 時間 が 短い
	-0.00410	充電 時間 が 長い

表-2　サポート素性の一部

けでなく，各弱学習器が用いている属性を直接観察することができるので，どのような部分構造が文分類に有効に働いたかを知ることができるという利点がある．構造を持った事例を分類する手法としてTreeカーネル[2]を用いたSVMがあるが，この手法では素性空間が陰に定義されてしまうため，上記のような分析を与えにくい．

前節で言及した，携帯電話の評判分類，および自動車評価文の主観性判別問題（図-1参照）に対する実験結果を，**表-1**に示す．評判分類，主観性判定それぞれ5,700, 7,500文のタグ付きデータを用い，表では5分割交差検定によって得られたF値（精度と再現率の調和平均）を掲げてある．「構造なし」は個々の単語を個別属性として用いる従来法，「構造あり」は文を係り受け木と見なし，その任意の部分木構造（部分木）を素性として用いる提案法である．表より，木構造を用いる提案法が，評判分類，主観性判定のいずれのタスクにおいても文分類精度の向上をもたらしていることがうかがえる．

携帯電話評判分類の実験において，得られた素性（部分係り受け木）の例を**表-2**に示す．本来，得られた素性は部分係り受け木であるが，スペースの都合上，木に含まれる形態素（活用がある場合には，その原型）のみを示してある．数値は，各部分木が分類にどのように寄与しているかを表す重みである．正の数値が「良い点」の分類に肯定的に寄与する属性，負の数値はその逆を示す．たとえば，「切れる」が「にくい」に係る構造は「良い」カテゴリにプラスに働いているのに対し，「にくい」を含むその他の表現（「にくくなった」「読みにくい」など）は，「悪い」カテゴリを分類するのに寄与している．「使う」を含む構造では，「使いたい」「使ってる」「使いやすい」などが正の重みを持つのに対し，「使いやすかった」「を使ってた」のように過去形になったり，「方が使いやすい」のように比較になると負の重みを持つことが分かる．また，同じ「充電時間」を含む場合でも，これが「短い」に係るか「長い」に係るかで，評価が正反対になっている．このような属性は，BOWに基づくモデルでは用いることのできない情報である．

これまでのデータマイニング手法の多くは，単一のデータから頻出する特徴的なパターンを発見することに焦点を当てていた．しかし，機械学習アルゴリズムの素性選択手法として十分有効であることが本実験により確認された．自然言語データ，塩基配列，化合物といった複雑データの分類，変換，生成といった応用にデータマイニングの種々のアルゴリズムがその活躍の場を広げていくであろう．

参考文献

1）Abe, K., Kawasoe, S., Asai, T., Arimura, H. and Arikawa, S.: Optimized Substructure Discovery for Semi-structured Data, in Proc. of PKDD, pp.1-44 (2002).
2）Collins, M. and Duffy, N.: New Ranking Algorithms for Parsing and Tagging: Kernels over Discrete Structures and the Votede Perceptron, in Proc. of ACL, pp.263-270 (2002).
3）山本，工藤，坪井，松本：系列パターンマイニングによる対訳表現抽出，情報処理学会研究報告，2002-NL-149, pp.15-22 (2002).
4）Yamamoto, K., Kudo, T., Tsuboi, Y. and Matsumoto, Y.: Learning Sequence-to-sequence Correspondences from Parallel Corpora via Sequential Pattern Mining, in Proceedings of the HLT-NAACL 2003 Workshop on Building and Using Parallel Texts, pp.50-56 (2003).
5）Zaki, M: Efficiently Mining Frequent Trees in a Forest, in Proc. of KDD, pp.71-80 (2002).
6）工藤，松本：カーネル法を用いた言語解析における高速化手法，情報処理学会論文誌，Vol.45, No.9, pp.2177-2185 (Sep. 2004).
7）工藤，松本：半構造化テキストの分類のためのブースティングアルゴリズム，情報処理学会論文誌，Vol.45, No.9, pp.2146-2156 (Sep. 2004).

（平成16年12月1日受付）