情報処理学会論文誌 Vol. 52 No. 1 269–283 (Jan. 2011)

新たな弱教師付き型分類手法 Bautext

グェン ファム タン タオ[†1,†2] 岡部 誠[†3,†4]
尾内 理紀夫[†5,†6] 林 貴宏[†7] 西岡 悠平[†2]
竹中 孝真[†2] 森 正弥[†2]

本稿では，web 上の大量のレビュー情報を要約する際の基盤技術として，単語を意味的カテゴリに分類するための手法，**Bautext** を提案し評価する．**Bautext** は弱教師付き手法であり，係り受け関係と相互情報量に基づいた名詞・名詞句のカテゴリ分類を行う．**Bautext** の特徴は以下の 4 つである．1）既存のブートストラッピング法等は，性能が多数のパラメータに依存するため，ユーザは良い分類精度を得るためのパラメータ設定を試行錯誤して見つける必要があった（小町ら，2010）．一方，**Bautext** においてはユーザは多数のパラメータ設定をする必要がなく，少数の種語を与え，各カテゴリと単語の関連度（配属スコア）を計算することにより，漸次種語を増加させ，分類を自動化させている．2）既存のブートストラッピング法では，反復ごとに多数のカテゴリが 1 つの単語を獲得しようとするときに再度評価のステップがあった．一方，**Bautext** においては，各カテゴリが独立な特徴語集合を持ち，それをもとに各カテゴリへの単語の配属スコアを計算し，最大スコアのカテゴリが単語を獲得することでこの再度評価のステップをなくした．そのため，ブートストラッピング法と比べて高速な分類アルゴリズムとなっている．3）既存のブートストラッピング法では意味ドリフトという課題がある．意味ドリフトの原因は，反復処理の過程において，新しい単語を獲得するために使われる抽出パターン数が定数個であるため，以前の各反復で抽出できた適切な抽出パターンの影響が消されることにあると考えられる．これに対して，**Bautext** では，各カテゴリが，独立な特徴語集合に今まで抽出できた適切な特徴語（抽出パータンと同じ役割）を保存することと反復ごとに分類対象の単語をランダムに選択させることにより，意味ドリフトを制御する効果が期待できる．4）目的の分類カテゴリに加えて「その他」カテゴリを導入することで，本来評価対象となりえない単語が「その他」カテゴリに移動し，目的の分類カテゴリの適合率が向上するという特徴がある．評価実験では，まず「その他」カテゴリの導入効果を確認した．また，代表的なブートストラッピング法である **Basilisk** および **Espresso** の 2 手法と **Bautext** とを比較し，両者に比べ，**Bautext** が分類精度，速度，使いやすさの 3 点において有効な手法であることを確認した．

Bautext as a New Minimally Supervised Classification Technique

Nguyen Pham Thanh Thao,[†1,†2] Makoto Okabe,[†3,†4]
Rikio Onai,[†5,†6] Takahiro Hayashi,[†7]
Yuhei Nishioka,[†2] Takamasa Takenaka[†2]
and Masaya Mori[†2]

We present and evaluate Bautext, a method for classifying terms into semantic categories, as a fundamental technique used for review summarization of drastically increasing volume of user reviews on the internet. Bautext is a minimally supervised technique for classifying nouns and noun phrases based on dependency relations and mutual information. Bautext has four important features. 1) There is no parameter that the user must manipulate except for seed words. Using an existing bootstrapping method, the user has to find a reasonable setting of multiple parameters by trial and error, on which the classification accuracy heavily depends (Komachi, et al., 2010). On the other hand, Bautext has no such a parameter, and after specifying seed words, no user intervention is required. 2) Bautext is a fast method compared with state-of-the-art bootstrapping methods. 3) Bautext is supposed to constrain sematic drift with independent feature sets for each category and the randomly choosing a term for classification in each classifation step. 4) We introduce “other” category to improve the precision. Adding an extra “other” category to the target categories, it is possible to improve the precision significantly on the trade-off between precision and recall. In our experiment, we compare Bautext with two major bootstrapping methods, Basilisk and Espresso, which show that Bautext is superior in classification accuracy, computational expense, and usability.

†1 電気通信大学大学院電気通信学研究科
Graduate School of Electro-Communications, University of Electro-Communications
†2 楽天技術研究所
Rakuten Institute of Technology
†3 科学技術振興機構さきがけ
PRESTO, Japan Science and Technology Agency (JST)
†4 電気通信大学情報理工学部
University of Electro-Communications
†5 電気通信大学電気通信学部
Faculty of Electro-Communcations, University of Electro-Communications
†6 電気通信大学大学院情報理工学研究科
Graduate School of Infomatics and Engineering, University of Electro-Communications
†7 新潟大学工学部情報工学科
Department of Information Engineering, Faculty of Engineering, Niigata University

1. はじめに

インターネットの普及にともない，一般人がレビューサイトやブログ等に発信するコンテンツは増加している．これらの情報は，ユーザにとっては買い物をしたりサービスを利用したりするときの参考情報として，そして，企業にとってはユーザのフィードバック情報として，注目を集めてきた．そして，ウェブに存在する大量のレビューを読むことはレビュー閲覧者にとって大きな負担となるため，評判要約の研究に対してその実用化への要求が高まっている．大量のレビューを適切に要約すること，そして，商品やサービスの評価ポイントを詳細に探索したり検索したりできるようにすることが望ましい．

属性に着目した多くの既存研究は電気製品に主眼を置き，製品の属性ごとに，属性の評判（好評，不評）とその属性に対する具体的なコメントを羅列する形に要約を行ってきた[12)–15)]．しかし，ホテル，レストラン等サービス系のレビューの場合は，**評価対象**（例：“接客態度”，“朝食” 等）が電気製品よりバラエティに富んでいるため，上述の羅列型の要約より，各カテゴリにまとめた方が評価ポイントが把握しやくなると考えられる．たとえば，ホテルの場合，レビュー・データから評価対象を抽出し，「サービス」，「部屋」，「立地」，「食事」等のカテゴリに振り分けることである．

本研究が最終的に目指すのは**図 1** のように，たとえば，「サービス」カテゴリにおける詳細な評価対象とそれらの評判（好評か不評か）をコンパクトに提示することである．ただし，本稿では，レビューデータから評価対象を表す単語を抽出し，事前に定義されたカテゴリ群に自動的に分類するタスクを研究目的とする．

図 1　本研究の目指す評判要約（「サービス」カテゴリの例）
Fig. 1　Aiming summarization style.

単語のカテゴリ分類では一般的に機械学習手法が使用される．教師ありの学習手法では，学習に使うデータは人手で作成することが多いので，作成コストが大きい．一方，教師なし学習は，類似したデータがまとまるだけで学習結果の意味づけが難しい．そこで，少数の教師データから学習を開始し，学習の過程で学習に使うデータを拡大していく半教師あり学習（semi-supervised learning）は有力である．半教師ありの手法は大きく 2 種類に分けることができる．1 番目は確率モデルを駆使して，ラベル未知のデータに対して，各ラベルごとにデータ（特徴ベクトルとして表現される）の確率分布を反復的に調整し，真の分布に収束させる手法である．あるラベルに対する真の分布とは，そのラベルに属するデータが本来従う確率分布のことである．代表的な手法には **EM** アルゴリズムがある[23),24)]．

2 番目はヒューリスティックな仮説に基づく **Self-training** と呼ばれる手法である[16)]．ラベル付きデータで学習した学習器（マッピング関数）を使って，ラベルなしデータのラベルを予測する．その中から信頼度の高いものを選び，次の反復の学習データに追加して再び学習する．これらのステップを繰り返す手法である．

本研究では比較的少量の教師データのみを必要とする新たな弱教師付き分類手法（minimally supervised classification techniques）を提案する．提案手法は **Self-training** に属する手法として位置づけられる．提案手法の入力は，分析用のコーパスと各カテゴリに対する少数の種語（教師データ）のみである．少数の種語からスタートして，レビューデータから抽出された評価対象を 1 つずつ各カテゴリに分類していく．

ユーザの利用シーンを想定すると，図 1 で示しているように，各カテゴリにおいて，代表的な評価対象のみ表示するので，誤分類の評価対象が表示されれば，ユーザに大きな違和感を与えることになる．そのため実用化に際し，再現率よりも適合率を優先すべきである．

また，一般的に自動分類の結果は 100%の精度（再現率も適合率も）を得ることができないため，自動分類システムが出力した結果を人手でフィタリングする後処理の過程が必要と考えられる．これは，編集者が各カテゴリの分類結果に目を通して，上位から順に正しい分類結果のみを選択する過程である．このため，上位の適合率が高いほど，この作業の負担が軽減できる．実用化に向けては，高い上位適合率を持つ自動分類が必要であり，本研究はそれを目指す．

本稿では，検証実験としてレビューデータに出現する評価対象を分類する．具体的には，Web で公開されている楽天トラベル「お客様の声」を解析し，評価対象を抽出して 7 つの目的カテゴリ（サービス，部屋，風呂，立地，設備・アメニティ，食事，料金）に分類する．また，分類結果の適合率を向上させるため，目的カテゴリ以外に，「その他」カテゴリを設

定する．これにより，目的カテゴリに分類すべきでない単語が「その他」に分類されるようになり，結果的に適合率が向上することを目指す．

以降，2 章で関連研究をあげ，本研究の位置づけを明確にする．3 章で提案手法である **Bautext** アルゴリズムについて説明する．4 章で評価実験について述べる．比較実験では，「その他」カテゴリの導入効果を確認する．また，**Bautext** と既存手法である **Basilisk** および **Espresso** の分類精度を比較する．5 章で全体をまとめる．

2. 関 連 研 究

属性に着目した評判要約の研究として，Hu ら[12),13)]，Popescu ら[14)] や Su ら[15)] 等の教師なしの手法を採用した研究があるが，これらの研究は属性の抽出にとどまり，抽出した属性をカテゴリやトピックに整理しないため，評価対象が雑多なレビュー・データには適さない．

本研究と同じように，評価対象をトピックにグルーピングする研究には Tivo ら[17)] の研究がある．この手法は教師なしモデル PLSA を採用している．この手法は分類精度に限界があり，トピックが判定できない単語が多く存在するという問題がある．

Riloff ら[1)]，Schapire ら[2)] や Roark ら[3)] 等はブートストラッピング法を提案した．彼らは同じカテゴリに属する単語が近くに出現する（同じリスト，結合語，同格語または複合語名詞に属する）という仮定に基づき，カテゴリのメンバの抽出を行った．この仮定で正しい関係を抽出できる場合もあるが，満足できる抽出の精度は得られていない．

Jones ら[4)] や Riloff ら[5),6)] はこのアイデアを発展させて，単なる単語間の距離関係ではなく，抽出パターンによる単語の候補の抽出を行った．抽出パターンは種語が主語や目的語に含まれるフレーズである．たとえば，“headquartered in X”，“to occupy X” 等は「場所」を指す単語の抽出のために使われる．そして，マルチ・レベルのブートストラップ（multi-level bootstrapping）により，反復ごとに新しい単語を獲得する．マルチ・レベルのブートストラップの利点は反復ごとに抽出パターンが再度スコアリングされるため，一番適したパターンが抽出に利用されることにある．

しかし，ブートストラッピング法の問題点として知られているのは意味ドリフト（semantic drift）である．これは関連性の低い単語または抽出パターンが加わると精度が急速に落ちる傾向があるということである．意味ドリフトを減らすために，James ら[10)] と McIntosh ら[11)] は相互的排除ブートストラップのアルゴリズム（Mutual Exclusion Bootstrapping）を提案した．これはすべての一般性の高い単語とパターンを抽出過程から外すことにする．しかし，この方針は，中小規模の分析では抽出できる単語やパターンが限定されるため，再現率が大きく低下すると考えられる．

Phillips ら[7)] や Thelen ら[8)] はマルチ・レベルのブートストラップのアルゴリズムをさらに発展させ，**Basilisk** を提案した．**Basilisk** では，各カテゴリでのブートストラップは相互的そして並列的に行われる．これにより，メンバ語が多いカテゴリが他のカテゴリの領域を侵害する可能性を減らしている．たとえば，並列ではなく順次にカテゴリを処理していく場合を考える．カテゴリ A が B より先に処理されると，B に属する可能性が高い単語が B の処理が始まる前に A に獲得されてしまうということもある．そのため，**Basilisk** は A と B の処理を同時にスタートさせて，A と B が同じ単語を獲得しようとしたならば，より適切なスコアリングで対応する．このアイデアにより，**Basilisk** は最も良い精度を出している手法として知られるようになった．

Basilisk は Riloff ら[1),2)] の研究をベースにして，改善され続けてきた手法であるが，同じくブートストラッピング法として **Expresso** という手法が存在する．**Espresso** は基本的に **Basilisk** と同じ仕組みを持っているが，パターンとインスタンスの評価式，停止条件が異なっている．また，**Basilisk** は反復ごとに 5 個の単語がアウトプットとして獲得されるが，**Espresso** は上位の単語を次の反復の種語としては利用するが，アウトプットとしては扱わない．

一方，提案手法は，ユーザは少数の種語のみを与えるだけという使いやすさを持ち，新たな弱教師付きアルゴリズムにより，特徴語の重みとそれをもとにした単語と各カテゴリとの関連度（配属スコア）を計算することにより適合率を向上させ，「はじめに」に述べたように，分類システムの実用化への貢献を目指している．

3. Bautext アルゴリズム

3.1 Bautext の特徴

Bautext の基本的な考え方は，同じ特徴的な単語群と共起する単語が同じカテゴリに属する可能性が高いと考えることにある．たとえば，「○○がおいしい」，「○○を食べた」等のように “おいしい”，“食べる” 等とよく共起する単語はカテゴリ「食事」のメンバである可能性が高いと考えられる．“おいしい”，“食べる” 等の単語を特徴語と呼ぶ．

本手法はすべての名詞，名詞句を評価対象候補とするため，1 つのカテゴリはそのカテゴリの中身となるメンバ語集合と特徴語集合からなる（**図 2** 参照）．特徴語としては内容語（名詞，名詞句，動詞，動詞句，形容詞）のみを使う．

Bautext の特徴は以下の 3 点である．

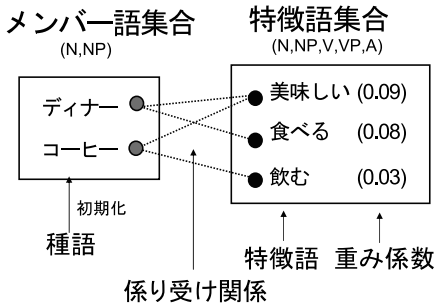

図 2　Bautext におけるカテゴリの構成
Fig. 2　Category composition of Bautext.

(1)　既存のブートストラッピング法に比べパラメータがない（種語のみ使用する）．
(2)　既存のブートストラッピング法に比べ高速である．
(3)　「その他」カテゴリの導入により適合率向上を図っている．
(1) については 3.1.1 項，(2) については 3.1.2 項で述べる．(3) については 3.9 節，4.2.2 項で詳しく述べる．

3.1.1　パラメータ数

ブートストラッピング法の弱点はパラメータの最適な設定法がないことである[18]．基本的なパラメータとして，1）インスタンス抽出に使うパターンの個数 P，2）各反復で抽出するインスタンスの個数 I がある．

P が小さすぎれば，各カテゴリにおいては，少数の関連性の強いパターンが反復ごとに繰り返し選択されることが多く，新しい単語が抽出できなくなり，反復が早く終わってしまう．逆に，P が大きすぎれば，関連性の低い単語も多く抽出し，意味ドリフトが発生する可能性がある．また，抽出パターンが定数個選択されるため，新しい単語の抽出に使われるパターンが毎回変わる可能性があり，前の反復において正しく選択されたパターンが選択されなくなる可能性がある．これに対して，**Bautext** では，前の反復において抽出された特徴語をすべて特徴語集合に保存することで，P をなくした．全特徴語が保存されるため，重みが小さな特徴語でも新しい単語を抽出することができる．

ブートストラッピング法では，各カテゴリにおいて，反復ごとに選択された P 個の抽出パターンによって新しい I 個の単語が抽出される．これは各カテゴリが独立に行う．I を小く設定すれば分類時間がかかり，I を大きく設定すれば，関連性の低い単語が多く抽出されるため，適切な I の調整が一般的に困難である．一方，**Bautext** では，各分類ステップでは，1 つの単語に対して，各カテゴリが競争して，その単語と一番関連性が高い（計算式の詳細は 3.5 節を参照）特徴語集合を持つカテゴリが獲得する．

3.1.2　高　速　性

Basilisk の特徴は反復ごとに最適な P 個のパターンを選択するためにすべてのパターンが繰り返し評価されることにある．しかし，多数のパターンの中には関連性のないパターンが多く存在するため，反復ごとに大きな無駄な計算が繰り返されている．また，各反復においては，多数のカテゴリが 1 つの単語を獲得しようとするときに，再度評価するステップがあるので，さらに時間がかかる．これに対して，**Bautext** では，各カテゴリが独立な特徴語集合を持ち，それをもとに各カテゴリへの単語の配属スコアを計算し，最大スコアのカテゴリが単語を獲得することで，この再度評価のステップをなくし，高速化を図っている．

3.2　前　処　理

3.2.1　名詞，名詞句，特徴語の抽出パターン

本研究は評価対象のほとんどが名詞句であることを想定し，各名詞・名詞句と係り受け関係を持つ内容語（名詞，名詞句，動詞，動詞句，形容詞）を特徴語として扱う．本研究は形態素解析器 Mecab [⋆1] を使って，各品詞単語を以下のように抽出した（本稿に使用する品詞情報は IPA 品詞体系[⋆2]を参考にしている）．

(1)　形容詞としては「自立・形容詞」と「非自立・形容詞」の両方が特徴語として抽出される．たとえば，
例 1：リンゴが 赤い
例 2：リンゴが 赤く ておいしい
において Mecab は例 1 と例 2 の「赤い」をそれぞれ「自立・形容詞」，「非自立・形容詞」として解析している．
(2)　動詞句としては「名詞サ変連続」に続き，「自立動詞」が出現するパターンを抽出する．たとえば，「更新できる」，「利用する」等である．
(3)　名詞句の抽出はやや複雑であり，**図 3** の抽出パターンにマッチした名詞句をすべて抽出した．**表 1** に抽出されたパターンの中から例としていくつかのパターンとそれ

⋆1 形態素解析器 Mecab
http://mecab.sourceforge.net/
⋆2 IPA 品詞体系
http://www.unixuser.org/ euske/doc/postag/index.html

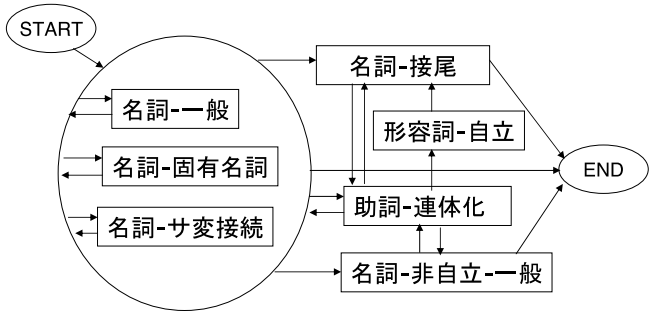

図 3　名詞句の抽出パターン
Fig. 3　Extraction patterns for noun phrase.

表 1　名詞句の抽出パターンと抽出例
Table 1　Sample of extraction patterns and extracted noun phrases.

パターン	例の名詞句
名詞-一般・名詞-一般・名詞-一般	足・もみ・マッサージ，コンタクト・ケア・用品，レディース・プラン・特典，マイナス・イオン・効果，...
名詞-一般・名詞-サ変接続・名詞-接尾-一般	Drink・サービス・券，空気・清浄・機，コーヒー・割引・券，靴・脱・場，男性・従業・員，海鮮・料理・店，立体・駐車・場，...

ぞれの抽出例を示す．

3.2.2　係り受けペアの抽出

本研究の 1 つの基本要素は名詞，名詞句と特徴語間の係り受け共起関係である．共起関係を用いる既存研究には，ある距離内にともに出現する 2 つの単語が共起関係を持つという「ウィンドサイズ」法がある．この方法の利点は網羅性にあるが，一方，共起関係がゆるいため，適切でない関係が多く抽出され，分類精度に影響することが考えられる．

本研究では各名詞・名詞句と係り受け関係を持つ内容語は名詞・名詞句を特徴づける役割があると考え，名詞・名詞句がカテゴリに分類されるとき，これらの内容語を特徴語とした．特徴語として扱われる単語類は名詞，名詞句，動詞，動詞句，形容詞である．特徴語を生成するために，データセットから係り受けペアの抽出が必要である．

係り受けペアの抽出手順を以下に記す．

- **Step 1**　文を 1 つ取り出し，左から右に文節ずつ走査していく．
- **Step 2**　処理中の文節を p とすれば，p 内の名詞を抽出する．文節の最後が連体化助詞「の」で終わった場合，次の文節を処理して，名詞句を抽出する．
- **Step 3**　抽出された名詞・名詞句を n とすると，n の先行の係り句を抽出して，1 つの係り受けペアとして保存する．同様に後行の係り句を係り受けペアとして保存する．
- **Step 4**　n を名詞・名詞句集合に，n の係り句を特徴語集合に保存する．
- すべてのデータの処理が終わったら終了，そうでなければ **Step 1** に戻る．

たとえば，「美味しいステーキを食べる」の例では 2 つの係り受けペア（ステーキ，美味しい），（ステーキ，食べる）が抽出される．

3.3　種語の設定

初期設定では，人手で各カテゴリに種語を与える必要がある．分類結果は選択された種語に依存すると考えられるため，本研究では，少数の種語で多くの特徴語を生成すべく，最も出現頻度の高い単語から種語を選ぶ方針を採用した．実際は以下の手順で種語を選択する．

(1)　名詞句を出現頻度の降順にソートする．
(2)　上位から順に 1 個 1 個の単語を手動で各目的カテゴリに種語として設定する．そのとき，目的カテゴリに設定できない単語は「その他」カテゴリに種語として設定する．

3.4　特徴語の重み係数

1 つの特徴語が同時に多数のカテゴリの特徴語となりうるので，その特徴語の重み係数をカテゴリごとに設ける．カテゴリ k においての特徴語 f_i の重み係数は k における f_i の意味的情報量の大きさを表す．その結果，カテゴリ k においては，より k の意味を表す単語にはより大きな重み係数が設定される．たとえば，カテゴリ “食事” においては “美味しい” の重み係数は “良い” の重み係数より大きく設定される．

相互情報量は意味的特性に注目した単語の抽出に効果的であるため，特徴語の選択・抽出の研究に多く使われている[25)–27)]．相互情報量には 2 種類がある．1 番目は 2 つの離散確率変数 X，Y の相互依存の尺度を表す相互情報量[21)]（Mutual Information，以降は I_{MI} と記述する）であり，

$$I_{MI}(X,Y) = \sum_{x_i \in X} \sum_{y_j \in Y} P(x_i, y_j) \log \frac{P(x_i, y_j)}{P(x_i)P(y_j)}$$

により定義される．2 番目は 2 つのイベント（確率変数の値）の相互依存の尺度を表す自己相互情報量（Pointwise Mutual Information，以降は I_{PMI} と記述する）[20)] である．

$$I_{PMI}(x_i, y_j) = \log \frac{P(x_i, y_j)}{P(x_i)P(y_j)}$$

ただし，$x_i \in X$，$y_i \in Y$

Espresso では後者を使っているが，I_{PMI} は低頻度の単語に対して大きな重みを与える偏り（バイアス）がある．一方，I_{MI} の式のとおり，I_{MI} は $P(x_i, y_j)$ で重み付けされた I_{PMI} の平均である．低頻度の単語ペアは $P(x_i, y_j)$ によって影響が低減されるため，I_{MI} は上述のバイアスを受けない[19]．このため，本研究では重み係数と配属スコアの計算に I_{MI} を採用した．

ここでは，カテゴリ k を考え，k の特徴語集合を $F_k = \{f_1, f_2, ..., f_h\}$ とし，k のメンバ語集合を $N_k = \{n_1, n_2, ..., n_l\}$ とする．このとき，特徴語 $f_i \in F_k$ のカテゴリ k に対するの重み係数は基本的に語 f_i とメンバ語集合 N_k の間の相互情報量で計算されるが，一般性の高い語（高頻度の単語）の影響を減らすため，$TF\text{-}IDF$ の考え方を適用し，上の相互情報量を R_i で除する．R_i は語 f_i が共起する名詞句の数である．下はカテゴリ k における特徴語 f_i の重み係数の計算式である．

$$W_k(f_i) = \frac{1}{R_i} \sum_{n_j \in N_k} P(f_i, n_j) \log \frac{P(f_i, n_j)}{P(f_i)P(n_j)}$$

ただし，$P(f_i, n_j) = \frac{Q(f_i, n_j)}{F_{pair}}$，$P(f_i) = \frac{Q(f_i)}{F_{word}}$，$P(n_j) = \frac{Q(n_j)}{F_{word}}$，$F_{word}$ は単語の出現頻度の総数，F_{pair} は係り受けペアの出現頻度の総数，$Q(f_i)$，$Q(n_j)$ はそれぞれ f_i と n_j の出現頻度，$Q(f_i, n_j)$ は n_j に係った f_i の出現頻度である．

よって，本重み係数は名詞句のカテゴリ分類において，低頻度語から高頻度語まで，幅広い範囲にわたって有効なスコアとして機能するものだと考える．

3.5　配属スコア

Bautext の分類アルゴリズムでは，名詞句 n_i を分類する際，n_i の各カテゴリに対する配属スコアを計算し，最大配属スコアが算出されたカテゴリに n_i がメンバ語として分類される．配属スコアの計算には特徴語の重み係数を考慮した相互情報量を用いる．

Bautext においては分類プロセス中に各カテゴリの特徴語が拡張されていくため，名詞句 n_i を分類する時点のカテゴリ k の特徴語集合を F_k とすると，名詞句 n_i のカテゴリ k に対する配属スコアは次の式で計算される．

$$S_k(n_i) = \sum_{f_j \in F_k} W_k(f_j) P(f_j, n_i) \log \frac{P(f_j, n_i)}{P(f_j)P(n_i)}$$

式のとおり，メンバ語集合と特徴語集合の間の相互情報量はメンバ語の配属スコアの和である．つまり，配属スコアは，そのメンバ語がそのカテゴリに対して，どのくらいそのカテゴリの情報量を増やせるかという貢献度の尺度となっている．

3.6　分類アルゴリズム

本節では分類アルゴリズムの詳細を述べる．**図 4** では例として 2 つのカテゴリ「食事」と「部屋」のインスタンスを獲得するプロセスを示している．それぞれの種語集合は（ディナー，コーヒー）と （部屋，ベッド）とする．

Step1　各カテゴリの種語集合を手動で設定する．

Step2　各カテゴリの初期化．

- 各カテゴリに対して，種語集合を初期メンバ語集合とし，各名詞句と係り受け関係を持つ単語を収集し，カテゴリの初期特徴語集合を生成する．図 4 では，右側の語が特徴語，エッジが係り受け関係を示している．
- 次に，各特徴語に重み係数を付ける．この重み係数は特徴語とメンバ語集合との相互情報量で計算される．つまり，重み係数が特徴語のメンバ語集合との結び度合いを表す．図 4 では，同じ特徴語 “良い” が異なるカテゴリに対し，異なる重み係数を持っている．

Step3　分類．

- 未分類の名詞・名詞句 n_i を取得する．
- n_i の各カテゴリに対する配属スコアを計算する．
- n_i に対して，カテゴリ k が最大配属スコアを算出したならば，n_i は k に分類される．図 4 は名詞 “お茶” の分類を例として示している．「食事」と「部屋」に対して，“お茶” の配属スコアはそれぞれ 0.8 と 0.1 であるため，“お茶” が「食事」に分類される．
- k の特徴語集合を F_k とすると，F_k に属していなかった，そして n_i と係り受け関係を持つ単語が F_k に新しい特徴語として追加される（図 4 では，“香ばしい” が新しく追加される）．

Step4　更新．

- k のメンバ語集合が 1 個増えたので，新しいメンバと係り受け関係を持つすべての特徴語の重み係数を更新する．図 4 は “飲む”，“香ばしい” と “良い” の重み係数が更新される．

Step5　すべての名詞句が分類されたならば終了．そうでなければ **Step3** に戻り，処理を繰り返す．

3.7　係り受け解析の誤りが分類精度に与える影響

Bautext では，各カテゴリの重要な特徴語の重みが反復ごとに漸次増加するアルゴリズ

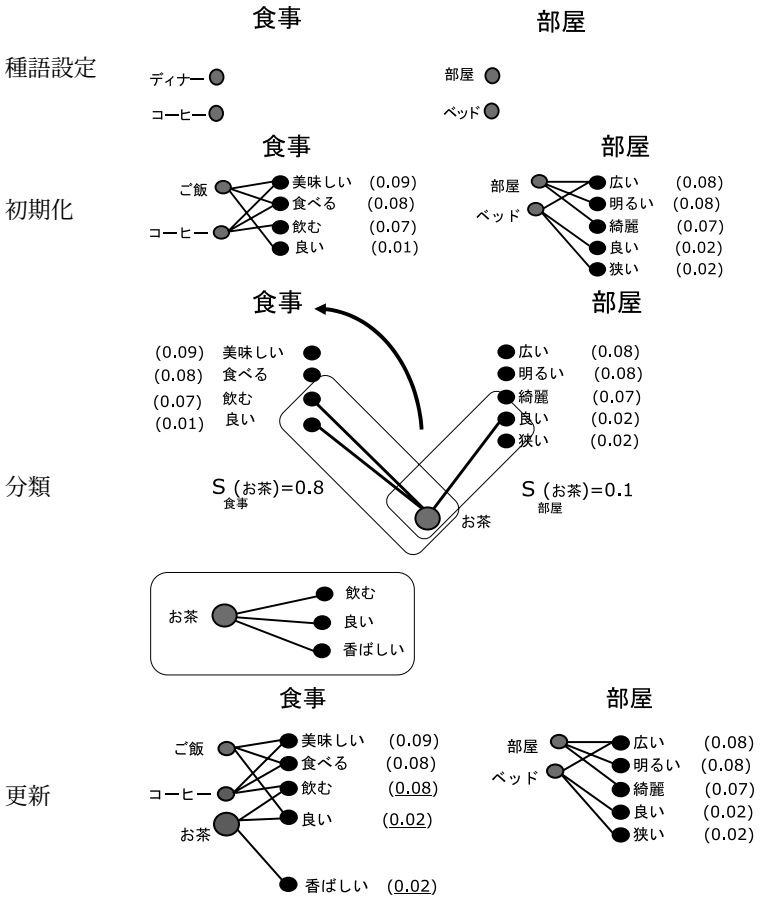

図 4　分類アルゴリズムによる処理プロセス
Fig. 4　Processing steps of classification algorithm.

ムである．逆に，重要でない特徴語の重みはが小さく制御される．このため，分類精度に対して，重要でない特徴語の影響が漸次低下していく．

適切でない関係が抽出された場合の例をあげる．たとえば，「明日食べに行くうどんはおいしいよ」の文から（「うどん」,「行く」）が係り受け関係ペアとして抽出されたとする．このとき，「うどん」がカテゴリ「食事」に分類されたならば，「行く」が「食事」の特徴語となるが，これは適切とはいえない．しかし，食事関連の単語と単語「行く」の関連性が低いため，**Bautext** アルゴリズムにより結果的に「行く」の重みが小さく抑えられ，食事カテゴリの分類精度への影響を小さくすることができる．

一方，適切な係り受け関係が抽出されない場合，つまり，あるカテゴリにとって適切な特徴語が抽出されない場合はこれらの特徴語に強い関連性を持つ単語がどのカテゴリにも分類されない恐れがある．

3.8　意味ドリフトに対する効果

2 章で述べたように **Basilisk** と **Espresso** 等のブートストラッピング法では意味ドリフトという課題がある．ブートストラッピング手法では，カテゴリに属する単語（種語）と共起する抽出パターンが定数個選択され，このパターン集合を用いて新しい単語を獲得するという処理ステップを繰り返す．

意味ドリフトというのは，誤分類の単語がある程度累積していくに従って，これらの誤分類の単語の影響が漸次大きくなり，関連性の低い単語を漸次獲得することである．この悪影響をどう抑えるのかは意味ドリフト対策のポイントである．

既存のブートストラッピング手法では，抽出パターンが定数個選択されるため，繰返し処理の過程において，新しい単語を獲得するために使われるパターンが変化する．誤分類の単語がある程度累積したときに適切でないパターンが選択される確率が高くなり，これらのパターンの影響が増大すること（これは適切なパターンの影響が消されることを意味する）が意味ドリフトの原因だと考えられる．

これに対して **Bautext** で採用した対策は次のようなものである．

- 各カテゴリが自分の特徴語集合（抽出パターンの集合と同じ役割を持つ）を持ち，前の処理ステップで抽出された特徴語のすべてがこの集合に保存され，新しい単語の獲得に使用される．これにより，適切な特徴語の影響が消されることなく新しい単語を抽出することができる．
- 分類対象の単語を事前に抽出しておいた名詞・名詞句リストからランダムに選択し，各カテゴリの特徴語集合との連結度（配属スコア）によって分類を決定することとした．このように，分類対象の単語が適切でない特徴語（パターン）と無関係（ランダムに）選択されるから，これらの適切でない特徴語の影響が増大する確率が低くなる．

以上により，**Bautext** アルゴリズムでは，これら 2 つの対策によって意味ドリフトを抑制する効果が期待できる．

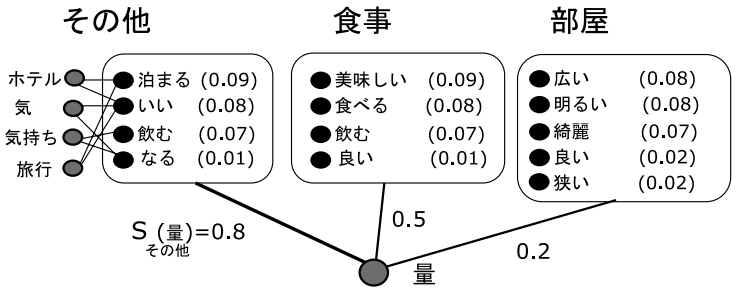

図 5「その他」カテゴリの設定
Fig. 5　Setup for category “Other”.

3.9　「その他」カテゴリの設定

Bautext アルゴリズムは，ここまででは一番高い配属スコアを出すカテゴリに名詞・名詞句を分類せざるをえないため，適切なカテゴリがない場合は誤分類が起こりかねない．この問題に対処するために，本研究では，ユーザが設定した各カテゴリ（目的カテゴリと呼ぶ）以外に，「その他」カテゴリを設定することにした．「その他」カテゴリを設定することにより，目的カテゴリに分類すべきでない名詞・名詞句が「その他」カテゴリに分類されることを狙う．**図 5** は名詞 “量” の誤分類例の防止の様子を示している．「その他」カテゴリも目的カテゴリと同様，種語群が人手により設定され，名詞・名詞句群と特徴語群も分類プロセス中に拡張されていく．

なお，「その他」カテゴリに分類される名詞・名詞句はすべてゴミ（不適切な単語）というわけではない．誤分類の防止を目的とするため，目的カテゴリと違う話題を含む可能性があり，「その他」カテゴリに分類するということである．

「その他」カテゴリが設定されることにより，目的カテゴリに分類すべき単語が「その他」カテゴリに誤って分類されてしまう可能性も考えられる．実際「その他」カテゴリにおいてどの程度このトレードオフを抑えられるかは興味深いところである．考察結果は 4.2.2 項で述べる．

4. 実　　験

4.1 実 験 設 定

本節では 4.2 節で説明する 4 種類の実験に共通する実験設定について述べる．

4.1.1　実験データとカテゴリ設定

楽天トラベル「お客様の声」からダウンロードしたレビューを実験データとし，7 つの目的カテゴリ「部屋」,「食事」,「風呂」,「サービス」,「設備・アメニティ」,「立地」,「料金」と「その他」カテゴリの計 8 カテゴリに評価対象の候補（名詞，名詞句）を抽出して分類する．そのうち，6 つのカテゴリ（「部屋」,「食事」,「風呂」,「サービス」,「設備・アメニティ」,「立地」）は実際「お客様の声」サイトにおいてホテルに対する 6 つの評価項目として設定されている．

上述のカテゴリ設定では，「設備・アメニティ」や「部屋」等の一部の評価対象が「サービス」に属したり，「風呂」の一部も「部屋」に属したりすると考えられる．つまり，各カテゴリ間の独立性がないため，分類問題としては良いカテゴリ設定ではない．しかし，実際に運営しているサイトに設定されているため，本研究は実用的なカテゴリ設定として取り込んだ．

「その他」カテゴリには，“気”，“気持ち”，“感じ” 等の評価対象となりにくい単語が分類される．これ以外，“旅館”，“景色” 等の単語は一般的には評価対象となりうるが，今回の 7 つの目的カテゴリに分類することは適切でないため「その他」カテゴリに分類すべきだと考える．

4.1.2　種語の設定

実験データから，すべての名詞・名詞句を抽出し，出現頻度の降順にソートし，そのリストを上から下に走査しながら，各カテゴリの種語を人手で選択した．カテゴリごとに 10 単語以内に収めるようにしたため，「サービス」は 10 個の種語が揃った時点で終了した．結果的に「料金」には 4 個の種語しか収集できなかった．選んだ結果を**表 2** に示す．

4.1.3　評 価 方 法

実験を行うにあたり正解データの作成に関して被験者（複数人の学生）に協力してもらった．**表 3** はこの正解データを示している．

評価実験では，実験結果とこの正解データを比較し，適合率，再現率，F 値を集計し評価する．そして，他の手法と比較する際には，結果の全体を評価するため，2 種類の平均（マイクロ的，マクロ的）を計算する．マイクロ平均はカテゴリの枠を越えて，データセットの全体で計算する平均であり，マクロ平均は各カテゴリで計算した結果の平均である．それぞれの計算式は以下であり μ はマイクロ，M はマクロを示す．A，B，C，D の意味は**表 4** のとおりである．ただし，$|K|$ はカテゴリ総数，P_i はカテゴリの適合率，R_i はカテゴリの再現率である．

表 2　各カテゴリの種語群
Table 2　Seed words for each category.

カテゴリ	**種語**（出現頻度）
食事（計 9 語）	朝食（1,145），食事（1,248），料理（564），夕食（499），バイキング（406），パン（375），レストラン（197），味（193），コーヒー（168）
部屋（計 9 語）	部屋（6,127），音（439），ベッド（337），トイレ（268），臭い（233），窓（186），テレビ（175），冷蔵庫（177），シャワー（172）
風呂（計 5 語）	風呂（1,932），温泉（521），浴場（483），露天風呂（246），お湯（214）
サービス（計 10 語）	サービス（637），対応（422），フロント（377），笑顔（137），接客（99），フロントの方の対応（94），スタッフ（61），従業員（61），応対（79），配慮（50）
立地（計 5 語）	駅（682），立地（686），場所（459），コンビニ（261），スーパー（32）
設備（計 7 語）	設備（497），駐車場（390），アメニティ（243），施設（143），空気清浄機（63），加湿器（52）
料金（計 4 語）	値段（794），料金（569），価格（394），コストパフォーマンス（134）
その他（計 32 語）	ホテル（1,846），気（1,217），感じ（767），機会（743），人（568），お世話（532），宿（519），宿泊（519），気持ち（380），他（352），子供（307），雰囲気（291），種類（276），割（244），印象（240），内容（240），プラン（238），利用（236），気分（227），旅館（226），仕事（221），申し分（206），最高（203），建物（195），思い出（191），景色（181），外（174），好感（172），是非（167），不満（167），旅行（164），とも（157）

表 3　人手により分類されたデータ
Table 3　Distribution of manually classified data among categories.

カテゴリ	**総数**	**割合**
食事	185	9.2%
部屋	267	13.3%
風呂	93	4.6%
サービス	208	10.3%
立地	117	5.8%
設備・アメニティ	107	5.3%
料金	53	2.6%
その他	**986**	**48.9%**

$$P^{\mu} = \frac{A}{A+B} = \frac{\sum_{i=1}^{|K|} A_i}{\sum_{i=1}^{|K|}(A_i + B_i)}$$

$$R^{\mu} = \frac{A}{A+D} = \frac{\sum_{i=1}^{|K|} A_i}{\sum_{i=1}^{|K|}(A_i + D_i)}$$

表 4　表記の意味
Table 4　Symbol's meaning.

カテゴリ i		人手	
		TRUE	FALSE
Bautext	TRUE	A_i	B_i
	FALSE	D_i	C_i

$$P^{M} = \frac{\sum_{i=1}^{|K|} P_i}{|K|}$$

$$R^{M} = \frac{\sum_{i=1}^{|K|} R_i}{|K|}$$

4.2　実 験 結 果

4.2.1　（実験 1）分類順と精度への影響の測定

本実験では，単語の分類順が精度に与える影響について確認した．その理由は，**Bautext** では，反復ごとに 1 つの単語をランダムに選び分類対象とするため，前の反復で分類された単語の分類正誤が，後に続く分類に選択される単語の分類結果に与える影響の有無を確認する必要があるからである．本実験ではランダム順で 20 回単語を分類した．適合率，再現率，F 値の平均値（計 8 カテゴリのマイクロ平均），および標準偏差を求めた．

- 適合率の平均値 **0.54**，標準偏差 **0.005**，
- 再現率の平均値 **0.42**，標準偏差 **0.004**，
- F 値の平均値 **0.47**，標準偏差 **0.003**，

が得られた．実際に 20 回分の測定データは**図 6** のようになっており，いずれの尺度も平均値に対し，ばらつき（標準偏差）が十分に小さいことが確認できる．従って「分類順が結果に与える影響が小さい」と考えられる．

4.2.2　（実験 2）「その他」カテゴリの効果の確認

本実験では，「その他」カテゴリは分類性能を向上させる効果があるか，そして，「その他」が設定されることによりどの程度適合率が向上したかを確かめる．

表 5 は「その他」が設定されなかった場合と設定された場合の結果を示している．「目的のみ」とは 7 つの目的カテゴリのみの平均を意味する．マイクロ平均では，再現率 −17%，適合率 +19%で，マクロ平均の方は，再現率 −12%，適合率 +17%で再現率が低下した分，適合率が改善されたことが分かる．結果的に F 値は変わらない．しかも，「その他」の再現

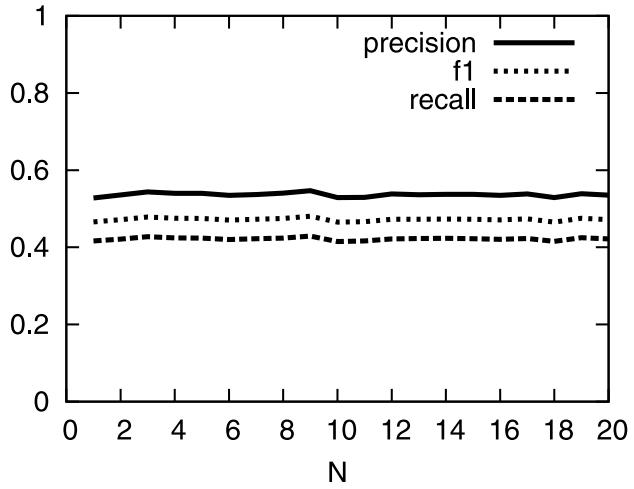

図 6　Bautext 分類順の影響
Fig. 6　Effect of classified order.

表 5　Bautext の分類結果における「その他」の効果
Table 5　Effect of category “Other” at Bautext’s classification result.

カテゴリ	「その他」なし 実行時間：20s			「その他」あり 実行時間：28s		
	Re	**Pr**	**F**	**Re**	**Pr**	**F**
食事	0.59	0.42	0.49	0.49	0.60	0.54
部屋	0.58	0.27	0.37	0.30	0.41	0.35
風呂	0.32	0.12	0.17	0.22	0.33	0.27
サービス	0.41	0.34	0.37	0.13	0.51	0.21
立地	0.43	0.32	0.37	0.34	0.54	0.42
設備・アメニティ	0.07	0.30	0.11	0.07	0.37	0.12
料金	0.43	0.34	0.38	0.37	0.50	0.43
その他	—	—	—	0.58	0.58	0.58
マイクロ平均（目的のみ）	0.45	0.29	0.35	0.28	0.48	0.36
マクロ平均（目的のみ）	0.40	0.30	0.35	0.28	0.47	0.35

率と適合率が 58%であり，「その他」カテゴリは精度向上効果があるといえる．

図 7 は 7 つの目的カテゴリの上位 N 単語の適合率のグラフを示している．それぞれのグラフでは「その他」カテゴリが設定されなかった場合（点線）と設定された場合を示してい

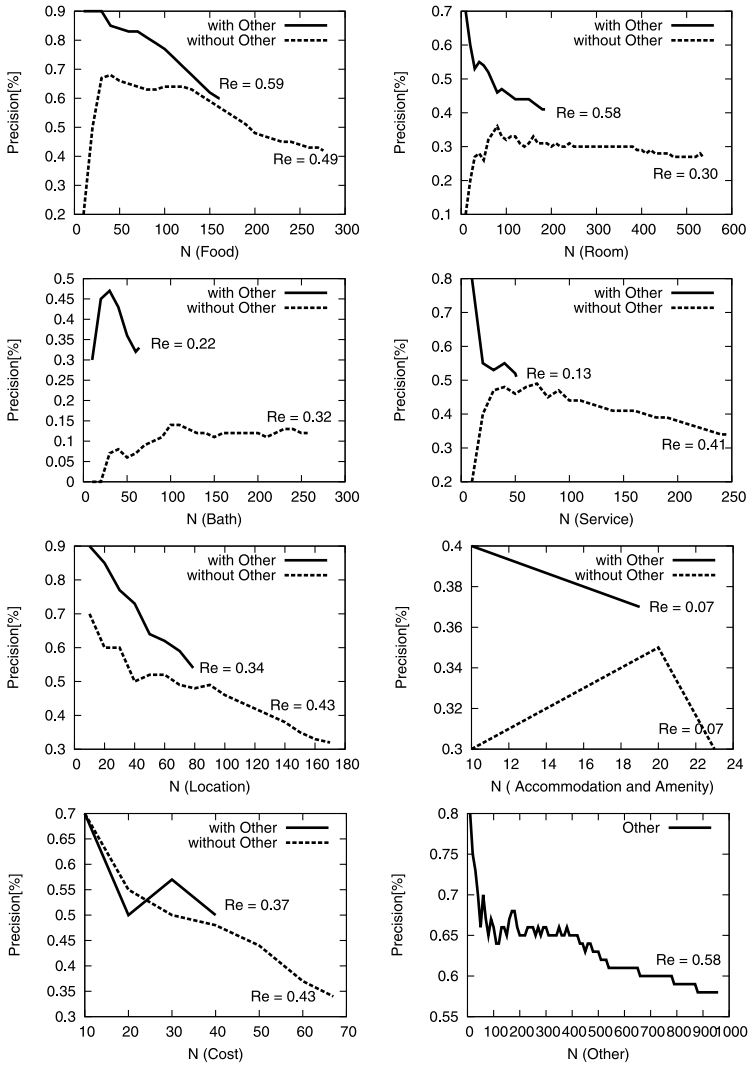

図 7　「その他」カテゴリの効果
Fig. 7　Effect of category “Other”.

る．単語が配属スコアの降順にソートされている．右の列の最下にあるグラフは「その他」カテゴリの結果である．それぞれのカテゴリ分類精度の範囲が違うため，縦軸の数値の範囲が違っていることに注意されたい．

結果を見ると「その他」が設定されると，カテゴリ全体の適合率が向上するだけでなく，上位の適合率も大きく改善されることが分かる．たとえば，食事（Food）の上位 100 語の適合率を見ると「その他」が設定されなかった場合と比べ，適合率が 65%から約 80%に向上している．同じように，部屋（Room）の上位 100 語の適合率も 30%から 45%に改善されていることが分かる．

1 章で述べたように，上位適合率が高いという結果は実用化において大きな利点である．つまり，上位適合率が高いということは，図 1 に示すように各カテゴリにおいて代表的な評価対象のみを表示する際に，評価対象の誤分類が少なく，ユーザに大きな違和感を与える可能性が小さいことを意味する．また，自動分類システムが出力した結果を人手でフィタリングする後処理が必要になったとしても，この作業の負担を軽減できる．

4.2.3 （実験 3）Basilisk との比較

既存手法である **Basilisk** は，単語のカテゴリ分類を目的として利用可能という点で **Bautext** と共通している．一方で，**Basilisk** はインスタンスを抽出するパターンを利用する方法をとっており，**Bautext** の係り受けを利用した方法とは異なる．したがって，解析的に **Bautext** と **Basilisk** を比較することは困難である．そこで，実験により **Bautext** と **Basilisk** の精度比較を行った．**Basilisk** 等のブートストラッピング法の大きな弱点として多数のパラメータの最適な設定法がないことがある[18]．分類精度にパラメータ依存性があるため，公平性を保つために，使用したデータセットに最適なパラメータ設定を探して比較することにした．

Basilisk の設定すべきパラメータとして，1）インスタンス抽出に使うパターンの個数 P，2）各反復で抽出するインスタンスの個数 I，3）実行の反復回数 N がある．**Basilisk** では P=20 と設定されるのが一般的なので[8]，本実験では，P=20 と固定し N の上限を 100 に設定した．I が分類性能に影響を与える傾向があるため[10]，I を変化させ，適合率と再現率のバランスである F 値のマイクロ平均で最適な I を探すことにした．**図 8** は目的カテゴリのマイクロ平均の結果を，**表 6** は I に応じた分類時間を示している．

図 8 を見ると，I を増やすと適合率が大きく向上されるが，再現率も大きく低下する．F 値で考えれば一番良い分類精度は I=100 のときだといえる．Thelen らはその論文[7),8)] では I=5 と設定したが，I=5 の分類精度と比べ，I=100 の方は同じ再現率を出していながら，適合率のマイクロ平均が 10%以上高くなっている．

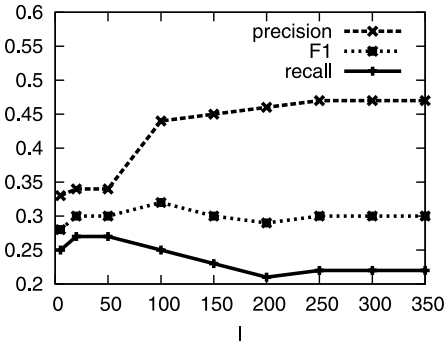

図 8　Basilisk におけるパラメータと分類性能の変化
Fig. 8　Performance change of Basilisk in variation of parameters.

表 6　Basilisk における I と分類時間の変化
Table 6　Change of processing time in variation of parameter I.

I	分類時間
5	32 m11 s
20	12 m26 s
50	10 m45 s
100	**6 m45 s**
150	6 m16 s
200	5 m58 s
250	5 m55 s
300	5 m49 s
350	5 m30 s

また，Thelen らの論文では「その他」カテゴリが設定されなかったが，「その他」が **Basilisk** の分類精度を改善させる効果があると実験で分かったため，本実験では **Basilisk** に対しても「その他」が入ったカテゴリ設定で行った．**表 7** は I=100 との比較結果を示している．これにより，「サービス」と「立地」カテゴリ以外では，残りの 6 つのカテゴリにおいては，**Bautext** が **Basilisk** を F 値で上回った．「食事」と「料金」では適合率と再現率の両方において **Basilisk** より上回った．「部屋」，「風呂」，「設備・アメニティ」では，適合率または再現率のどちらか一方で上回っており，F 値は **Bautext** が上回っている．「サービス」と「立地」においては劣っているが，全体のマイクロ平均とマクロ平均ではすべてのカテゴリ

表 7　Basilisk I=100 との比較結果
Table 7　Performance of Bautext comparing to Basilisk when I=100.

Category	Bautext			Basilisk I=100		
	Re	Pr	F	Re	Pr	F
食事	**0.49**	**0.60**	**0.54**	0.43	0.56	0.48
部屋	**0.30**	0.41	**0.35**	0.11	0.43	0.17
風呂	0.22	**0.33**	**0.27**	0.21	0.24	0.22
サービス	0.13	**0.51**	0.21	**0.30**	0.41	**0.35**
立地	0.34	0.54	0.42	0.34	**0.65**	**0.44**
設備・アメニティ	0.07	**0.37**	**0.11**	0.04	0.27	0.07
料金	**0.37**	**0.50**	**0.43**	0.28	0.33	0.30
その他	**0.58**	0.58	**0.58**	0.21	0.68	0.32
マイクロ平均（目的のみ）	**0.28**	**0.48**	**0.36**	0.25	0.44	0.32
マクロ平均（目的のみ）	**0.28**	**0.47**	**0.35**	0.24	0.41	0.31

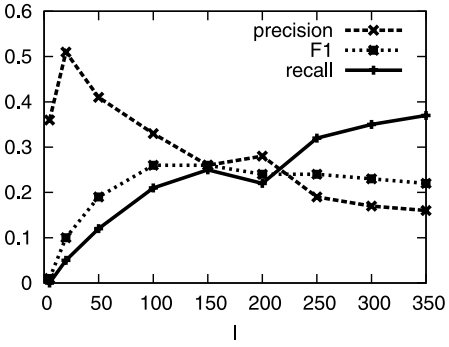

図 9　Espresso におけるパラメータと分類性能の変化
Fig. 9　Performance change of Espresso in variation of parameters.

で **Bautext** は **Basilisk** を上回っている．上述したように，「食事」，「風呂」，「設備・アメニティ」，「料金」において，その適合率は **Bautext** が上回っており，実用化に向けて適合率を重視するという研究目的に合致している．

そして，分類性能以外に関心のある点は分類速度である．表 6 を見ると，I=100 において，同じデータ・セットを処理するために，**Basilisk** は 6 分 45 秒もかかったが，**Bautext** の分類時間は 28 秒であり，**Basilisk** より 10 倍以上も速い．したがって，**Bautext** は **Basilisk** と比べて，より良い分類性能を出しながら，分類速度も速いことが分かった．

実用的な応用として，**Bautext** の速い分類速度を活かした以下の仕組みが考えられる．**Bautext** を採用した分類器と人間による共同分類の仕組みである．**Bautext** を実行させ，得られた結果の上位から人が良い種語を選択し，各カテゴリに手動で追加する．この作業を繰り返すことによって分類精度が上がると考えられる．**Bautext** の高速性ゆえ，この仕組みは十分な実用性があると考えられる．

Basilisk の特徴は反復ごとにパターンプール（pattern pool）に入っている抽出パターンが繰り返し評価されることにある．しかし，パターンプールはデータセットに出現するすべての抽出パターンを含有するため，反復ごとに再評価されるパターンの数が多い．この中に関係のないパターンが多く存在するため，無駄な時間を費やしてしまうと考えられる．また，反復ごとに各カテゴリが同じ単語を獲得しようとするとき，これらの単語を再度評価するステップがあるので，このステップでも時間がかかると考えられる．これに対して，**Bautext** では，共有のパターンプールではなく，それぞれのカテゴリにそれぞれの特徴語群（特徴語プール）を定義しているため，この無駄な時間を短縮することができたといえる．

4.2.4 （実験 4）Espresso との比較

Espresso は，単語のカテゴリー分類を目的として利用可能という点で **Bautext** と共通しているが，**Basilisk** 同様パターンを利用する方法をとっている．そこで実験により **Bautext** と **Espresso** の精度比較を行った．**Espresso** は基本的に **Basilisk** と同じ分類アルゴリズムを採用している．大きな相違点は分類の停止条件にある．**Basilisk** は反復ごとに抽出した単語を獲得インスタンスとして出力するため，各カテゴリにおいては，新しい単語が獲得できなくなった時点で自動的に終了する．しかし，**Espresso** は反復ごとに抽出した単語を出力の単語ではなく，次の反復の種語として用いるため，明確な停止条件を設定しなければ停止しない．Pantel らは論文 9) で停止条件を示しているが，この停止条件を本研究に使用しているデータセットに適用したときの分類性能が悪かったため，一般的な停止条件ではないと考えられる．そこで，小町ら[18] を参考して，正しく抽出できる単語の数が下がり始めたら終了とする．

また，**Espresso** では，各カテゴリが独立にインスタンスを獲得していくため，同じ単語を多くのカテゴリで獲得されることが許される．すなわち，「その他」を導入してもフィルタリング効果がない．このため，**Bautext** との比較実験では「その他」が設定されなかったときの **Espresso** の分類結果を採用した．

Basilisk と同様，P=20 と $N = 100$ に固定し，I を変化させ，マイクロ平均で最適な I を探すことにした．**図 9** は目的カテゴリのマイクロ平均の結果を示している．**Basilisk** は

表 8　Espresso $I = 100$ との比較結果
Table 8　Performance of Bautext comparing to Espresso when $I = 100$.

Category	Bautext			Espresso $I = 100$		
	Re	Pr	F	Re	Pr	F
食事	**0.49**	0.60	**0.54**	0.33	0.71	0.45
部屋	**0.30**	**0.41**	**0.35**	0.12	0.33	0.18
風呂	**0.22**	**0.33**	**0.27**	0.19	0.19	0.19
サービス	0.13	0.51	0.21	0.17	0.37	0.23
立地	0.34	**0.54**	**0.42**	0.34	0.44	0.38
設備・アメニティ	0.07	**0.37**	**0.11**	0.08	0.10	0.09
料金	**0.37**	**0.50**	**0.43**	0.35	0.20	0.25
その他	0.58	0.58	0.58	—	—	—
マイクロ平均（目的のみ）	**0.28**	**0.48**	**0.36**	0.21	0.33	0.26
マクロ平均（目的のみ）	**0.28**	**0.47**	**0.35**	0.23	0.33	0.27

I が大きくなると適合率が向上し，再現率が低下するが（図 8），**Espresso** は I が大きくなると適合率が低下し，再現率が向上するという逆の傾向を見せている．そして，一番良い分類性能を出しているのは $I = 100$ の場合である．

表 8 は **Bautext** と **Espresso** の $I = 100$ の場合の分類の比較結果を示している．これにより，「サービス」以外は，残りの 7 つのカテゴリにおいては，**Bautext** が **Espresso** を上回った．「部屋」，「風呂」と「料金」では適合率，再現率，F 値，すべての数値で上回った．「食事」，「立地」と「設備・アメニティ」では適合率あるいは再現率のどちらか一方で上回り，F 値は **Espresso** を上回った．そして，全体では，すべてのカテゴリのマイクロ平均とマクロ平均は **Espresso** を上回った．特に，**Espresso** と比べ，**Bautext** の適合率の平均（マイクロ平均とマクロ平均）は 14%以上上回っていることが分かる．上述したように，「部屋」，「風呂」，「立地」，「設備・アメニティ」，「料金」において，その適合率は **Bautext** が上回っており，実用化に向けて適合率を重視するという本研究の目的に合致している．

分類速度に関しては **Bautext** は 28 秒，**Espresso** は 18 秒で両方とも分類速度は速い．つまり，**Bautext** は **Espresso** と比べ，同程度の分類速度を持ちながら，より高い適合率の分類精度を出している．前述したように，実用化においては再現率より適合率が重視されるので，この結果は **Bautext** の優位性が示されるといえる．したがって，**Espresso** と比べ，高適合率の分類精度とパラメータがないことは **Bautext** の優位性を示しているといえる．

5. おわりに

Web 上の大量のレビュー情報を要約する際の基盤技術として，単語を意味的カテゴリに分類するための手法，**Bautext** を提案し評価した．**Bautext** は弱教師付き手法で，係り受け関係と相互情報量に基づいた名詞・名詞句のカテゴリ分類を行う．

類似手法である **Basilisk** と **Espresso** と比べ，パラメータがないのは **Bautext** の大きな特徴である．種語以外のユーザが操作すべきパラメータがないため使いやすいといえる．**Basilisk** と **Espresso** は分類性能が多数のパラメータに依存するため，ユーザは高い分類精度を得るための良いパラメータ設定を試行錯誤して見つける必要があった．これに対して，**Bautext** は種語を与えると分類は全自動で行う．

そして，**Basilisk** と **Espresso** と比べて高速な分類アルゴリズムである．実験結果では，**Basilisk** と比べ，10 倍以上の速度を出しながら，より良い分類精度を出した．一方，**Espresso** と比べ，同程度の速度を持つが，圧倒的に高い適合率の分類精度を持つことを確認した．

また，「その他」カテゴリの導入によって，適合率を向上させることができた．目的の分類カテゴリに加えて「その他」カテゴリを導入することで，トレードオフ関係にある適合率と再現率のうち，適合率を大きく向上させる効果が得られた．しかも，**「その他」**カテゴリの導入は **Bautext** だけでなく，**Basilisk** に対しても有効であり，適合率向上効果が得られた．適合率向上は，実用化に向けて適合率を重視するという研究目的に合致している．

評価実験として，Web で公開されている楽天トラベル「お客様の声」に含まれる名詞句を 8 つのカテゴリ（部屋，食事，風呂，サービス，設備，立地，料金，「その他」）に分類した．**Basilisk** と **Espresso** の 2 手法と比較した実験の結果，両者に比べ，**Bautext** が分類精度，速度，使いやすさの 3 点において有効な手法であることを示した．

今後の課題として，手法の更なる拡張および，他手法の利点を導入すること考えている．半教師あり手法による単語・テキストの意味的カテゴリ分類に焦点を置けば，確率モデルを用いた手法が多く存在する．たとえば，藤野らは，識別モデルと生成モデルを統合した半教師あり学習手法を提案している[22]．**Bag-of-words** 表現を用いたテキストの分類に適用し，高精度な分類結果を得ている．この手法が，我々の目的である単語のカテゴリ分類に有効であるかどうかを検証するために，適切な特徴量の検討を行い，比較実験することを予定している．

謝辞　本研究は，JST 戦略的創造研究推進事業さきがけ，楽天技術研究所の支援を受け

た．記して深謝する．

参 考 文 献

1) Riloff, E. and Shepherd, J.: A Corpus-Based Approach for Building Semantic Lexicons, *Proc. Conference on Empirical Methods in Natural Language Processing* (*EMNLP*) (1997).
2) Schapire, R.E. and Singer, Y.: Improved Boosting Algorithms Using Confidence-rated Predictions, *Proc. Annual Conference on Computational Learning Theory* (1998).
3) Roark, B. and Charniak, E.: Noun-phrase Cooccurrence Statistics for Semi-automatic Semantic Lexicon Construction, *Proc. Annual Meeting of the Association for Computational Linguistics* (*ACL*) (1998).
4) Jones, R., McCallum, A., Nigam, K. and Riloff, E.: Bootstrapping for Text Learning Tasks, *Proc. Workshop on Text Mining Foundations, Techniques and Applications* (1999).
5) Riloff, E. and Jones, R.: Learning Dictionaries for Information Extraction by Multi-Level Bootstrapping, *Proc. National Conference on Artificial Intelligence* (*AAAI*) (1999).
6) Riloff, E. and Shepherd, J.: A Corpus-Based Bootstrapping Algorithm for Semi-Automated Semantic Lexicon Construction, *Journal of Natural Language Engineering*, Vol.5, No.2, pp.147–156 (1999).
7) Phillips, W. and Riloff, E.: Exploiting Strong Syntactic Heuristics and Co-Training to Learn Semantic Lexicons, *Proc. Conference of Empirical Methods in Natural Language Processing* (*EMNLP*) (2002).
8) Thelen, M. and Riloff, E.: A Bootstrapping Method for Learning Semantic Lexicons using Extraction Pattern Contexts, *Proc. Conference on Empirical Methods in Natural Language Processing* (*EMNLP*) (2002).
9) Pantel, P. and Pennacchiotti, M.: Espresso: Leveraging Generic Patterns for Automatically Harvesting Semantic Relations, *Proc. International Conference on Computational Linguistics* (*COLING*) (2006).
10) James, R.C. Murphy, T. and Scholz, B.: Minimising Semantic Drift with Mutual Exclusion Bootstrapping, *Proc. Conference of the Pacific Association for Computational Linguistics* (*PACLING*) (2007).
11) McIntosh, T. and James, R.C.: Weighted Mutual Exclusion Bootstrapping for Domain Independent Lexicon and Template Acquisition, *Proc. Australasian Language Technology Association Workshop* (2008).
12) Hu, M. and Liu, B.: Mining and Summarizing Customer Reviews, *Proc. ACM Conference on Knowledge Discovery and Data Mining* (*SIGKDD*) (2004).
13) Hu, M. and Liu, B.: Mining Opinion Features in Customer Reviews, *Proc. National Conference on Artificial Intelligence* (*AAAI*) (2004).
14) Popescu, A. and Etzioni, O.: Extracting Product Features and Opinion from Reviews, *Proc. Conference on Empirical Methods in Natural Language Processing* (*EMNLP*) (2005).
15) Su, Q., Xu, X., Guo, H., Guo, Z., Wu, X., Zhang, X., Swen, B. and Su, Z.: Hidden Sentiment Association in Chinese Web Opinion Mining, *Proc. International World Wide Web Conference* (*WWW*) (2008).
16) Abney, S.: *Semisupervised Learning for Computational Linguistics*, Chapman and Hall/CRC (2007).
17) Tivo, I. and McDonald, R.: Modeling Online Reviews with Multi-grain Topic Models, *Proc. International World Wide Web Conference* (*WWW*) (2008).
18) 小町　守，工藤　拓，新保　仁，松本裕治：Espresso 型ブートストラッピング法における意味ドリフトのグラフ理論に基づく分析，人工知能学会論文誌，Vol.25, No.2, pp.233–242 (2010).
19) Bouma, G.: Normalized (pointwise) mutual information in collocation extraction, *Proc. Conference of the German Society for Computational Linguistics and Language Technology* (*GSCL*) (2009).
20) Church, K.W. and Hanks, P.: Word association norms, mutual information, and lexicography, *Journal of Computational Linguistics*, Vol.16, No.1, pp.22–29 (1990).
21) Cover, T. and Thomas, J.: *Elements of Information Theory*, Wiley & Sons, New York (1991).
22) 藤野昭典，上田修功，斉藤和巳：生成・識別ハイブリッドモデルに基づく半教師あり学習，第 4 回情報科学技術フォーラム（FIT）(2005).
23) Basu, S., Banerjee, A. and Mooney, R.J.: Semi-supervised clustering by seeding, *Proc. International Conference on Machine Learning* (*ICML*) (2002).
24) Blum, A. and Mitchell, T.: Combining labeled and unlabeled data with co-training, *Proc. Annual Conference on Computational Learning Theory* (*COLT*) (1998).
25) Li, J. and Hirst, G.: Semantic knowledge in word completion, *Proc. ACM SIGACCESS Conference on Computers and Accessibility* (*ASSETS*) (2005).
26) Fang, H. and Zhai, C.: Semantic term matching in axiomatic approaches to information retrieval, *Proc. Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (*SIGIR*) (2006).
27) Markines, B., Cattuto, C., Menczer, F., Benz, D., Hotho, A. and Stumme, G.: Evaluating similarity measures for emergent semantics of social tagging, *Proc. International Conference on World Wide Web* (*WWW*) (2009).

(平成 22 年 3 月 31 日受付)
(平成 22 年 10 月 4 日採録)

グェン ファム タン タオ（正会員）

2008 年電気通信大学電気通信学部情報工学科卒業．2010 年同大学大学院電気通信学研究科博士前期課程情報工学専攻修了．2010 年楽天株式会社入社．現在，楽天技術研究所に所属．自然言語処理と検索技術に興味を持つ．

岡部　誠（正会員）

2003 年東京大学理学部情報科学科卒業．2005 年同大学大学院情報理工学系研究科修士課程，2008 年同博士課程修了．2010 年まで Max Planck Institute for Informatics，computer graphics department でポストドクタを務める．現在，電気通信大学情報理工学部総合情報学科助教/科学技術振興機構さきがけ．コンピュータグラフィックスにおけるユーザインタフェースについて研究し，現在は動画データベースに基づくコンテンツ生成支援について研究を行っている．

尾内理紀夫（正会員）

1973 年東京大学理学部物理学科卒業．1975 年同大学大学院理学系研究科物理学専攻修士課程修了．同年日本電信電話公社（現 NTT）入社．1982 年〜1985 年に ICOT プロジェクトに参画，1997 年〜1998 年に RWC プロジェクトに参画．2000 年より電気通信大学教授．著書に『マルチメディアコンピューティング』（コロナ社），『コンピュータの仕組み』（朝倉書店），編書に『オブジェクト指向コンピューティング』（近代科学社）『インタラクティブシステムとソフトウェア』（近代科学社）等．マルチメディア情報処理，情報検索，セマンティックコンピューティング等に興味を持つ．工学博士（東京大学）．人工知能学会，日本ソフトウェア科学会，ACM 各会員．

林　貴宏（正会員）

1975 年生．1998 年金沢大学工学部電気・情報工学科卒業．2000 年同大学大学院自然科学研究科博士前期課程電子情報システム専攻修了．2003 年同研究科博士後期課程数理情報科学専攻修了．博士（工学）．2001 年石川工業高等専門学校電子情報工学科助手，2003 年電気通信大学電気通信学部情報工学科助手，2007 年同学科助教を経て，2009 年新潟大学工学部情報工学科准教授，現在に至る．マルチメディア情報検索の研究に従事．電子情報通信学会，日本ソフトウェア科学会，人工知能学会各会員．

西岡　悠平（正会員）

1977 年生．2003 年京都大学大学院情報学研究科複雑系科学コース修了．2005 年上期 IPA 未踏ソフトウェア創造事業スーパークリエータ認定．現在，楽天技術研究所チーフテクノロジストとして情報推薦とセマンティック Web の研究に従事．

竹中　孝真（正会員）

1991 年大阪大学工学部金属材料工学科卒業．同年より新日鉄ソリューションズ株式会社に勤務，2007 年楽天株式会社入社．楽天技術研究所に所属し，自然言語処理，レコメンデーションの研究を経て，現在，研究開発運営に従事．情報処理学会，言語処理学会，ACM 各会員．

森　正弥（正会員）

1975 年生．1998 年慶應義塾大学・経済学部卒業．2006 年楽天株式会社入社．現在，同社執行役員兼楽天技術研究所長として，研究開発組織のマネジメントに従事．電子情報通信学会会員．著作に『クラウド大全』（日経 BP 社，共著），『ウェブ大変化パワーシフトの始まり』（近代セールス社）等．