DBSJ Journal, Vol.11, No.1
June 2012

# Wikipediaとベイズ理論を用いた関連エンティティ推測と短文クラスタリングへの応用

## Related Entity Inference Using Wikipedia and Bayes' Theorem and Its Application for Short Text Clustering

白川 真澄♠　中山 浩太郎♦
原 隆浩♣　西尾 章治郎♣

Masumi SHIRAKAWA　Kotaro NAKAYAMA
Takahiro HARA　Shojiro NISHIO

**本論文では，自然文の入力に対してコンテキストに沿った関連エンティティを推測するための手法を提案する．自然文からの関連エンティティ推測というタスクは一般的にキーフレーズ抽出，語句の多義性解消，関連度計算から成る複合的な問題であり，これまで一つの問題としてあまり取り扱われなかった．提案手法ではこれらの問題に対し，Wikipediaから取得可能な情報をベイズ理論の枠組みに落とし込み，ナイーブベイズを用いて統一的に解決する．評価実験では，提案手法によって推測された関連エンティティを素性としてTwitterの投稿メッセージのクラスタリングを行い，既存手法と比較して大きく性能を向上できることを確認した．**

**In this paper, we propose a method to infer contextually related entities from a query text written in natural languages. The task of related entity inference from texts generally consists of keyphrase extraction, word sense disambiguation and relatedness measurement, and has been rarely regarded as a single problem to be solved. Our method integrates the knowledge extracted from Wikipedia into a framework of Bayes' theorem and solves the problem using Naive Bayes. In the experiment, we conducted Twitter message clustering using related entities inferred by our method, and confirmed that it significantly outperformed a comparative method.**

## 1. はじめに

語句間の関連度を計算する手法は，意味を考慮したテキスト解析のための重要な基盤技術である．語句間の関連度計算はWebが普及する前から行われている研究であるが，2006年以降，大規模Web百科事典であるWikipediaを利用した手法が注目を集めている．Wikipediaを用いた関連度計算手法は実際にアプリケーションの基盤技術として用いられることも多いが，その理由として，1) 明示的な知識（ノイズの少ない情報）を用いているため精度が高い，2) 固有名詞や専門用語が豊富，3) データの調達や実装が容易であることが挙げられる．代表的な手法の多くは，二つの語句（あるいはエンティティ）を入力とし，その間の関連度を，Wikipediaの記事やカテゴリ構造などを用いて算出する．しかし，短文クラスタリングを始めとする多くのアプリケーションでは，自然文の入力に対してコンテキストに依存した関連エンティティを取得することが求められるため，関連度計算だけでなく，キーフレーズの抽出，多義語の処理といった問題にも対応する必要がある．これらの問題に対して個別に手法を適用した場合，どの閾値によってキーフレーズを抽出するか，どの閾値によって多義性のある語句をエンティティにリンクさせるかなど，パラメータが増加し，適切なパラメータの設定が難しくなる．また，入力テキストに応じて適切なパラメータが変化するため，パラメータ調整によって安定した出力を得ることは困難である．

そこで本研究では，自然文の入力に対して関連エンティティを取得するための統一的な枠組みを，Wikipediaとベイズ理論を用いて構築する．具体的には，キーフレーズの抽出，多義語の曖昧性解消（エンティティリンキング），関連エンティティの取得といった個別の問題に対して，Wikipediaを用いた手法をベイズ理論の枠組みに落とし込み，ナイーブベイズを拡張した手法により一つの問題として取り扱う．本研究は，自然文からの関連エンティティ推測という複合的な問題に対し，一つの理論的枠組みにおいて解決しており，煩雑なパラメータ調整なしに安定した出力が得られる．また，提案手法を用いて取得した関連エンティティを素性として用いることにより，高精度・高粒度で文書クラスタリングが可能となる．特に，テキストが短い場合においても精度良く関連エンティティを推測できるため，Twitterに代表されるような短文のクラスタリングに有効である．

## 2. 関連研究

Wikipediaを知識抽出の対象とする研究（Wikipediaマイニング）は2006年に注目を集め，以降急速に研究対象としての認知度が高まっていった．Wikipediaは，Wikiをベースにした大規模Web百科事典であり，誰でもWebブラウザを通じて記事内容を変更できることが大きな特徴である．そのため，幅広い分野について，一般的なエンティティから新しいエンティティに至るまで記事が網羅されており，記事（エンティティ）数は，最も多い英語版で380万，日本語版で78万である（2012年1月時点）．また，Wikipediaは，記事の網羅性や即時性だけでなく，密な記事間リンク，質の高いアンカーテキスト，URLによる語義の一意性など，知識抽出のコーパスとして有利な性質を数多く持っている[7]．加えて，Wikipediaの全データがオンラインで無償公開[1]されていることも，Wikipediaマイニングに関する研究が急速に発展した要因の一つと考えられる．

Wikipediaマイニングに関する研究の中でも，関連度計算は多くの研究者が取り組んでいる研究であり，語義曖昧性解消[6]や照応解析[8]など，より高度な意味解析のための基盤技術として用いられている．関連度計算に関する代表的な研

♠ 学生会員　大阪大学 大学院情報科学研究科
shirakawa.masumi@ist.osaka-u.ac.jp
♦ 正会員　東京大学 知の構造化センター
nakayama@cks.u-tokyo.ac.jp
♣ 正会員　大阪大学 大学院情報科学研究科
{hara, nishio}@ist.osaka-u.ac.jp

[1] http://dumps.wikimedia.org/

究として，Strube ら[11]は，これまで WordNet に対して用いられてきた手法を Wikipedia に適用，すなわち Wikipedia のカテゴリ構造上の距離や記事の類似度を測ることで任意のエンティティ間の関連度を算出している．この研究では，複数のベンチマークや照応解析[8]のアプリケーションにおいて評価を行い，Wikipedia が関連度計算のリソースとして有効であることを示した．Gabrilovich ら[1]は，Wikipedia の記事に出現する語句について転置インデックスをとり，記事を基底とするベクトルによって語句を表現（Explicit Semantic Analysis, ESA）することにより，高精度かつ計算コストの低い実用的な関連度計算を達成している．また，ESA の特筆すべき点として，語句間のみならず任意のテキスト間について関連度を計算できることが挙げられる．これにより，文書クラスタリングを始めとする様々なアプリケーションにおいて ESA を適用することが可能であり，現在最も一般的に利用されている関連度計算手法の一つとなっている．実際，Song らの研究[10]では，（比較手法としてではあるが）ESA が Twitter の投稿メッセージ（ツイート）のクラスタリングに対して有効であることを示している．Milne ら[5]は，二つの記事についてフォワードリンク及びバックワードリンクの共有度をもとに関連度を算出している．この手法は，バックワードリンクについてのみ考慮した場合，記事に出現するアンカーリンクについて転置インデックスをとっていることになるため，本質的には ESA と同様のアプローチであるといえる．グラフ理論に基づく手法[7]では，Wikipedia の記事を頂点とするグラフを解析することにより，関連エンティティを取得している．二つの入力に対して関連度を計算するのでなく，一つの入力に対して関連するエンティティを関連度付きで取得できるため，クエリ拡張や広告マッチングなどのアプリケーションの基盤技術として利用可能である．Ito ら[2]は，リンク共起性に基づく手法を提案しており，グラフ理論に基づく手法[7]と比較して高速に連想シソーラス（関連度を定義した辞書）を構築できることを実証している．本研究では，単一の入力テキストに対して関連語句を取得する手法を提案しているが，Wikipedia を用いた関連度計算に関する既存研究で単一の入力テキストを想定したものは，筆者の知る限り存在していない．

## 3. 提案手法

本章では始めに，入力テキストから関連エンティティを推測するにあたって考慮すべき問題について論じる．その後，それらの問題に対し，Wikipedia から取得可能な情報をベイズ理論の枠組みに落とし込み，統一的に扱う手法を提案する．

### 3.1 考慮すべき問題

テキストの入力に対して関連エンティティを推測するというタスクを考える．このようなタスクに対しては，入力テキストからキーフレーズ（特徴語）を抽出し，それぞれの語句に対して関連エンティティを取得した後，複数のキーフレーズに共通して関連しているエンティティを出力する，といったように三つの小問題に分割し，個別に処理するのが一般的である．

まず，テキストが入力として与えられたとき，そのテキストから直接的に関連エンティティを導出することは困難であるため，テキストを語句単位に分割する必要がある．ここでは基本的な処理として，テキストからキーフレーズを抽出することが求められる．抽出したキーフレーズは入力テキストの大まかな内容を表現するものであるため，キーフレーズ抽出の精度が出力結果の精度に直接影響する．キーフレーズを抽出した後，それぞれの語句に対して関連エンティティを取得する．ここで求められていることは，二つの語句に対する関連度計算ではなく，一つの語句に対する関連エンティティの取得であることに注意する．すなわち，全ての語句ペアに対して現実的な時間で関連度計算が可能な手法か，あるいは入力語句からリアルタイムに関連エンティティを取得できる手法でなければならない．また，多義性のある語句が含まれている場合，語義曖昧性解消を行う必要がある．最後に，それぞれのキーフレーズから得られた関連エンティティの中から，複数のキーフレーズが構成するコンテキストに沿った関連エンティティを導き出す．直感的には，より多くのキーフレーズと関連のあるエンティティを優先的に出力することによってコンテキストの制約を考慮できそうであるが，具体的にどのような手法が最も効果的であるかについては検討が必要である．

テキストの入力に対して関連エンティティを取得する際に考慮すべき問題を以下にまとめる．

- 入力テキストからのキーフレーズ抽出
- 語句の曖昧性解消
- 単一の入力に対する関連エンティティ取得
- 複数語句が構成するコンテキスト制約

これらの問題に対しては，個別に焦点を当てた手法が存在しているが，手法を組み合わせる際にパラメータ調整が必要であることが多く，多様な入力に対して安定した精度を達成することが難しい．

### 3.2 手法の概要

前節で説明した，自然文からの関連エンティティ推測という問題に対し，Wikipedia から取得できる情報をベイズ理論の枠組みに落とし込み，ナイーブベイズを用いて統一的に解決する手法を提案する．本研究では，理論的な枠組み自体が持つ推論能力を利用することにより，多様な入力に対してロバストな手法の確立を目指す．

以下ではまず，Wikipedia から取得可能な情報について説明する．その後，それらの情報をもとに，ベイズ理論の枠組みにおいてテキストから関連エンティティを推測する手法について述べる．

### 3.3 Wikipedia から取得可能な情報

#### 3.3.1 キーフレーズ抽出

入力テキストに含まれる語句$t$がこのテキスト中のキーフレーズである確率$\mathrm{P}(t \in T)$を，Wikipedia のアンカーテキストを用いて算出する[4]. Wikipedia の記事の編集方針の一つに「ウィキ化(wikification)」というものがあり，記事中に登場する語句とそれが意味する記事（エンティティ）をリンクさせることで Wikipedia を整理する役目を持っている．このとき，重要な語句あるいは特徴的な語句ほど該当する記事に対してリンクが張られる傾向がある．この経験則に基づき，ある語句がアンカーテキストとして用いられている確率を，キーフレーズの確率として用いる．具体的には，語句$t$が出現する記事の数$CountDocuments(t)$と，その語句がアンカーテキストとして出現する記事の数$CountDocuments(t \in Key)$を用いておよその確率を算出する．

$$P(t \in T) \approx \frac{CountDocuments(t \in Key)}{CountDocuments(t)} \quad (1)$$

#### 3.3.2 エンティティリンキング

語句$t$がエンティティ$e$にリンクされる確率$P(e|t)$を，Wikipediaのアンカーテキストとリンク先の記事を用いて算出する[6]．前述の「ウィキ化」により，記事中に登場する語句とそれが意味する記事がリンクされているため，これを解析することで語句とエンティティの多対多の関係を抽出できる．すなわち，ある語句に注目したとき，その語句のリンク先として選ばれる回数の多い記事ほど，その語句が意味するエンティティとして適していると考えられる．語句$t$がアンカーテキストとしてエンティティ$e$の記事にリンクされている回数を$CountAnchortexts(t,e)$とするとリンク確率は以下の式により表される．

$$P(e|t) \approx \frac{CountAnchortexts(t,e)}{\sum_{e_i \in E} CountAnchortexts(t,e_i)} \tag{2}$$

なお，$E$はWikipediaで定義されているエンティティ（記事）集合である．

#### 3.3.3 関連エンティティ取得

エンティティ$e$が与えられたときにエンティティ$c$が連想される確率$P(c|e)$を算出する．ここでは，単純に記事間リンクを用いた手法と，Wikipediaの関連度計算手法として最もよく使われている手法の一つである ESA[1]を用いた手法について紹介する．前者の記事間リンクを用いた手法では，ある記事に対し，その記事と隣接リンク（フォワードリンクおよびバックワードリンク）によってつながっている記事について推移確率を算出する．エンティティ$e$の記事からエンティティ$c$の記事への隣接リンクの数を$CountLinks(e,c)$とすると，$e$から$c$が連想される確率は次式で表される．

$$P(c|e) \approx \frac{CountLinks(e,c)}{\sum_{c_j \in E} CountLinks(e,c_j)} \tag{3}$$

一方，ESAを用いた手法では，ある記事に対し，その記事にリンクしている記事（バックワードリンク）をベクトルで表し，コサインによってベクトルの類似度を算出する．なお，全ての記事ペアに対して関連度を算出しようとすると膨大な計算量となるため，隣接リンクによってつながっている記事間についてのみ関連度を算出する．エンティティ$e$と$c$のESAによる関連度を$Sim(e,c)$とすると，$e$から$c$が連想される確率は以下の式により定義される．

$$P(c|e) \approx \frac{Sim(e,c)}{\sum_{c_j \in E} Sim(e,c_j)} \tag{4}$$

また，式(2)を用いると，語句$t$からエンティティ$c$が連想される確率は以下のように導出できる．

$$P(c|t) = \sum_{e_i \in E} P(c|e_i)P(e_i|t) \tag{5}$$

#### 3.3.4 エンティティの一般度算出

エンティティ$c$の事前確率$P(c)$は$c$の一般度を意味する．エンティティの一般度は，$P(c|e)$を算出するときに用いた情報と同じものを用いることが望ましい．本研究では，フォワードリンクあるいはバックワードリンクによって隣接している記事間についてのみ関連度を算出しているため，ここではリンク数に基づいた一般度を用いる．つまり，他の記事とより多くリンクしている記事ほど一般度が高くなる．エンティティ$c$の記事の隣接リンクの数を$CountLinks(c)$とすると，事前確率は下記の式により定義できる．

$$P(c) \approx \frac{CountLinks(c)}{\sum_{c_j \in E} CountLinks(c_j)} \tag{6}$$

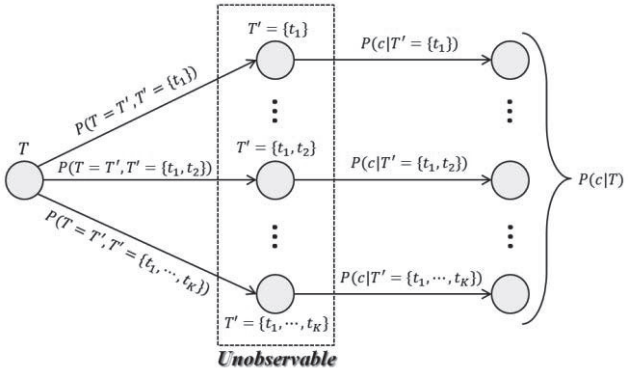


**図1　要素が観測できないキーフレーズ集合に対するナイーブベイズ**

**Fig. 1　Naive Bayes for a set of keyphrases in which members are unobservable.**

### 3.4 ベイズ理論に基づくテキストからの関連エンティティ推測

前節で説明した，Wikipediaから取得可能な情報をもとに，入力テキストからの関連エンティティ推測を試みる．まず，入力が複数のキーフレーズであった場合について考える．すなわち，キーフレーズ集合$T' = \{t_1, \cdots, t_K\}$が与えられたときの$P(c|T')$を求める[2]．ここで，各語句$t_k$については関連エンティティとその確率$P(c|t_k)$が分かっているため，この問題はナイーブベイズを適用できる[10]．具体的には，各語句は互いに生起確率に影響しないという仮定の下で，以下の式により関連エンティティとその確率を算出できる．

$$P(c|T') \propto P(c)\sum_{k=1}^{K} P(t_k|c) \propto \frac{\sum_{k=1}^{K} P(c|t_k)}{P(c)^{K-1}} \tag{7}$$

次に，入力となるキーフレーズ集合$T$の要素が観測できない場合について考える．この前提は，入力テキストの中でどの語句がキーフレーズであるかが分からない場合と同等であり，本研究で達成しようとしている問題に等しい．一般的な解決法として，先にキーフレーズ集合を決定し，その後に上記のナイーブベイズを適用する方法が考えられる．しかし，キーフレーズをどのように決定するかという問題が生じる．例えば，キーフレーズの確率に閾値を設ける方法では，適切な閾値の設定が必要となる．また，最適な閾値は入力テキストに対して変化すると考えられるため，チューニングによっ

[2] 本論文では要素集合が不明なキーフレーズ集合に対して$T$，要素集合が判明しているキーフレーズ集合に対しては$T'$とアポストロフィを付ける．

て適切な閾値を決定することは困難である．

そこで，直接観測できないキーフレーズ集合に対して確率的に集合の状態を決定し，各状態に対してナイーブベイズを適用する手法を用いる[9]．つまり，キーフレーズ集合$T$に関して，全ての起こりうる状態$T'$について$P(c|T')$を計算する．図1は観測できないキーフレーズ集合に対するナイーブベイズを表している．キーフレーズ集合$T$の各状態$T'$について確率$P(T=T')$を計算し，全ての状態についてそれぞれナイーブベイズを適用した後，それらの重み付き和をとっている．キーフレーズ集合$T$がある状態$T'$となる確率は，式(1)を用いて定義できる．

$$P(T=T') = \prod_{t_k \in T'} P(t_k \in T) \prod_{t_k \notin T'} P(t_k \notin T) = \prod_{t_k \in T'} P(t_k \in T) \prod_{t_k \notin T'} (1 - P(t_k \in T)) \quad (8)$$

したがって，図1に示すナイーブベイズによる関連エンティティの推測は，式(7)と式(8)より，以下のように表される．

$$P(c|T) \propto \sum_{T'} \left( P(T=T') \frac{\prod_{t_k \in T'} P(c|t_k)}{P(c)^{|T'|-1}} \right) \quad (9)$$

なお，$|T'|$は$T'$に含まれるキーフレーズの数を意味している．上式を計算しようとすると，入力テキストに含まれる語句数$K$に対して指数関数的に計算量が増加する．これは，すべての$T$の状態$T'$に対してナイーブベイズを適用しているためである．ここで，$t_k \in T'$の場合と$t_k \notin T'$の場合について整理すると，式(9)は次式に変形できる．

$$\frac{\sum_{T'} \left( \prod_{t_k \in T'} P(t_k \in T) P(c|t_k) \prod_{t_k \notin T'} (1 - P(t_k \in T)) P(c) \right)}{P(c)^{K-1}} \quad (10)$$

上式の分子を$t_k$ごとに分解することにより，和集合部分を効率よく計算できる[9]．その結果，以下の式が導かれる．

$$P(c|T) \propto \frac{\prod_{k=1}^{K} (P(t_k \in T) P(c|t_k) + (1 - P(t_k \in T)) P(c))}{P(c)^{K-1}} \quad (11)$$

式(11)は，ナイーブベイズの式(7)における個々の確率$P(c|t_k)$を，$P(c|t_k)$と事前確率$P(c)$の線形結合に置き換えたものであり，$P(t_k \in T)$によってその比重が決まる．結果的に，$P(t_k \in T)$はスムージングの比重を決定するための係数の役割を果たしている．つまり，$t_k$がキーフレーズである場合は$P(c|t_k)$，$t_k$がキーフレーズでない場合は$P(c)$であることを表している．キーフレーズである確率$P(t_k \in T)$が低いほど$P(c)$の値に近づき，分母の$P(c)$と相殺され，ナイーブベイズの結果への影響が小さくなる．

## 4. 評価

### 4.1 評価環境

提案手法の有効性を評価するため，提案手法によって取得した関連語句を用いてTwitterの投稿メッセージ(ツイート)のクラスタリングを行った．具体的には，Twitterのハッシュタグをもとにあらかじめ正解集合を定義しておき，提案手法を用いてツイートから関連エンティティを推測し，関連エンティティを素性としてK-meansクラスタリングを実行した．ハッシュタグとは，ツイートを発信するユーザが意図的に#Obamaや#MacBookのようにキーフレーズの直前に“#”を付けたものであり，そのツイートが言及しているトピックを明示的に表現する役割を持っている[3]．そのため，ハッシュタグを用いて短文クラスタリングのための疑似的な正解データを生成できる[10]．ここでは，ハッシュタグによる正解データが出来る限り正しいクラスタとなるよう，ハッシュタグのキーフレーズとして，曖昧性が低く，互いにトピックが独立しそうなものを選択した．評価に用いた三種類のデータセットを表1に示す．一つ目のデータセット(U)では，正解クラスタとして明確な差のあるユニークなカテゴリを想定し，政治，娯楽，スポーツ，情報技術，健康，宗教の各トピックから一つずつハッシュタグを選択した．二つ目(IT)と三つ目(S)のデータセットでは，同じトピックの中で異なるコンテキストを持つクラスタを想定し，情報技術，スポーツからそれぞれハッシュタグを選択した．データセットの作成手順として，1) 各ハッシュタグによる検索を行い，英語で記述されたツイートを収集，2) 同じデータセット内の別のハッシュタグを含むツイートを削除，3) リツイート（“RT”で始まり，他人のツイートの引用を表す），URLを除去，4) ツイートの末尾にあるハッシュタグは全て除去し，それ以外のハッシュタグは“#”のみを除去，5) 三単語以下のツイートを削除，の各処理を行った．各データセットの統計値を表1にまとめる．

表1　評価に用いた三つのデータセットと統計値
Table 1　Three datasets for evaluation and their statistics.

| データセット | ユニーク(U) | 情報技術(IT) | スポーツ(S) |
|---|---|---|---|
| ハッシュタグ（ツイート数） | #Obama (779)<br>#Bones (949)<br>#PGA (1,243)<br>#Microsoft (1,040)<br>#medicine (1,109)<br>#Christ (871) | #MacBook (1,251)<br>#Silverlight (221)<br>#VMWare (890)<br>#MySQL (1,241)<br>#Ubuntu (988)<br>#Chrome (1,018) | #NFL (1,043)<br>#NHL (1,045)<br>#NBA (1,085)<br>#MLB (752)<br>#MLS (969)<br>#UFC (984)<br>#NASCAR (857) |
| 総ツイート数 | 5,991 | 5,609 | 6735 |
| 総単語数 | 83,748 | 82,608 | 91,613 |
| ツイートあたり平均単語数 | 14.0 | 14.7 | 13.6 |
| 総語彙数 | 19,636 | 16,539 | 18,603 |

評価のベースラインとして，bag-of-wordsモデルによりツイートに出現する単語（ストップワードを除く）をそのままクラスタリングの素性とする手法(BOW)，また比較手法として，Gabrilovichらの ESA[1]によって得られたベクトルを素性とする手法(ESA)を採用した．提案手法における関連度計算には，単純に記事間リンクを用いた手法と，ESAを用いた手法についてそれぞれ評価を行った．提案手法及びESAでは，それぞれ関連語句の上位10，20，50，100，200，500，1000，2000，5000を素性ベクトルとしてクラスタリングを

表 2　ツイートクラスタリングの結果
Table 2　The result of tweet clustering.

| 評価指標 | purity | | | NMI | | | ARI | | |
|---|---|---|---|---|---|---|---|---|---|
| データセット | U | IT | S | U | IT | S | U | IT | S |
| BOW（ベースライン） | 0.591 | 0.580 | 0.533 | 0.292 | 0.320 | 0.296 | 0.218 | 0.277 | 0.239 |
| ESA (上位 10) | 0.492 | 0.644 | 0.467 | 0.213 | 0.428 | 0.247 | 0.180 | 0.408 | 0.182 |
| ESA (上位 20) | 0.503 | 0.649 | 0.475 | 0.219 | 0.426 | 0.229 | 0.170 | 0.394 | 0.174 |
| ESA (上位 50) | 0.554 | 0.659 | 0.518 | 0.262 | 0.445 | 0.301 | 0.212 | 0.425 | 0.224 |
| ESA (上位 100) | 0.567 | **0.695** | 0.554 | 0.297 | **0.483** | 0.365 | 0.241 | **0.451** | 0.261 |
| ESA (上位 200) | 0.565 | 0.651 | 0.559 | 0.311 | 0.475 | 0.341 | 0.248 | 0.419 | 0.232 |
| ESA (上位 500) | 0.537 | 0.613 | **0.604** | 0.299 | 0.421 | 0.382 | 0.226 | 0.354 | 0.279 |
| ESA (上位 1,000) | 0.585 | 0.599 | 0.583 | 0.326 | 0.423 | 0.370 | 0.286 | 0.351 | 0.280 |
| ESA (上位 2,000) | **0.624** | 0.555 | 0.588 | **0.380** | 0.346 | **0.397** | **0.301** | 0.277 | **0.299** |
| ESA (上位 5,000) | 0.623 | 0.424 | 0.533 | **0.380** | 0.211 | 0.359 | 0.292 | 0.140 | 0.218 |
| 提案手法（単純リンク，上位 10) | 0.415 | 0.688 | 0.433 | 0.125 | 0.436 | 0.177 | 0.104 | 0.428 | 0.148 |
| 提案手法（単純リンク，上位 20) | 0.540 | 0.751 | 0.561 | 0.225 | 0.482 | 0.289 | 0.202 | 0.500 | 0.245 |
| 提案手法（単純リンク，上位 50) | 0.638 | 0.795 | 0.690 | 0.342 | 0.552 | 0.441 | 0.292 | 0.572 | 0.427 |
| 提案手法（単純リンク，上位 100) | 0.682 | 0.806 | **0.702** | 0.408 | 0.627 | 0.518 | 0.347 | 0.591 | **0.428** |
| 提案手法（単純リンク，上位 200) | **0.731** | 0.802 | 0.694 | 0.489 | 0.602 | 0.507 | **0.392** | **0.609** | 0.391 |
| 提案手法（単純リンク，上位 500) | 0.716 | 0.786 | 0.690 | 0.495 | 0.667 | 0.527 | 0.370 | 0.602 | 0.370 |
| 提案手法（単純リンク，上位 1,000) | 0.693 | **0.810** | 0.667 | 0.503 | **0.669** | 0.526 | 0.346 | 0.577 | 0.304 |
| 提案手法（単純リンク，上位 2,000) | 0.674 | 0.808 | 0.697 | **0.533** | 0.586 | **0.546** | 0.294 | 0.471 | 0.320 |
| 提案手法（単純リンク，上位 5,000) | 0.673 | 0.630 | 0.666 | 0.531 | 0.445 | 0.494 | 0.293 | 0.335 | 0.274 |
| 提案手法（ESA ベース，上位 10) | 0.562 | 0.725 | 0.631 | 0.244 | 0.451 | 0.346 | 0.245 | 0.477 | 0.325 |
| 提案手法（ESA ベース，上位 20) | 0.620 | 0.756 | 0.684 | 0.314 | 0.506 | 0.434 | 0.299 | 0.527 | 0.416 |
| 提案手法（ESA ベース，上位 50) | 0.688 | 0.795 | 0.722 | 0.428 | 0.575 | 0.535 | 0.410 | **0.613** | 0.455 |
| 提案手法（ESA ベース，上位 100) | 0.727 | 0.810 | **0.761** | 0.492 | 0.637 | **0.554** | 0.467 | 0.596 | **0.504** |
| 提案手法（ESA ベース，上位 200) | 0.729 | **0.811** | 0.707 | 0.492 | 0.603 | 0.550 | 0.452 | 0.608 | 0.430 |
| 提案手法（ESA ベース，上位 500) | **0.774** | 0.794 | 0.713 | 0.552 | **0.644** | 0.547 | **0.537** | 0.600 | 0.402 |
| 提案手法（ESA ベース，上位 1,000) | 0.706 | 0.780 | 0.680 | 0.554 | 0.594 | 0.546 | 0.356 | 0.572 | 0.327 |
| 提案手法（ESA ベース，上位 2,000) | 0.720 | 0.775 | 0.674 | 0.552 | 0.574 | 0.512 | 0.389 | 0.508 | 0.340 |
| 提案手法（ESA ベース，上位 5,000) | 0.717 | 0.641 | 0.638 | **0.564** | 0.444 | 0.439 | 0.368 | 0.346 | 0.301 |

行った．クラスタリングの評価指標には純度(purity)，正規化相互情報量(NMI)，adjusted Rand index (ARI)を用いた．purity は最も適合率の高いクラスタのみを考慮した指標である．一方, NMI と ARI は全てのクラスタを考慮しており，NMI は情報理論的な解釈による指標，ARI は false-positive と false-negative となるクラスタにペナルティを与えた指標である．いずれのスコアにおいても 0 から 1 までの値をとり，値が大きいほどクラスタリングの性能が高いことを意味する．評価実験では，K-means クラスタリングにおいて局所解に陥る可能性を考慮し，それぞれの手法において初期値を変えて 20 回ずつクラスタリングを実行し，最もスコアの高かったものを採用した．

### 4.2 評価結果

ツイートのクラスタリング結果を表 2 に示す（各手法によるスコアの最大値は太字で表している）．ツイートに出現する語句をそのまま用いた場合(BOW)と比較して，Wikipedia を用いて意味情報を拡張する手法（ESA，提案手法）が，いずれの評価指標においても大幅に高いスコアを達成している．これは，ツイートあたりの平均単語数が十数程度（表 1 参照）であり，同じトピックに属するツイートでもあまり語句の共起がみられないためである．実際, Song らの研究[10]においても同様の傾向がみられ，ツイートのような短文のクラスタリングにおいては，BOW や統計的なアプローチではうまく機能しないと報告されている．

ESA でもベースラインと比べてクラスタリング性能が向上しているが，提案手法ではさらに，全ての評価指標において ESA に勝っており，提案手法の有効性が確認できる．これは，提案手法では，入力テキストからの関連エンティティ推測をベイズ理論に基づいて統一的に処理しており，安定した出力を得られるためであると考えられる．

また，提案手法において，単純にリンクの数から関連度を算出した場合と, ESA を用いて関連度を再計算した場合について比較すると，ESA を用いた手法のほうが purity および ARI のスコアが高くなっている．一方，NMI ではほぼ同等のスコアとなっていることから，語句間の関連度として単純な手法を用いても，ナイーブベイズをもとにした統一的な関連エンティティ推測の枠組みにより，比較的安定した出力が得られることが分かる．

クラスタリングの素性として用いる関連エンティティの数に関して，同じカテゴリから異なるトピックを選択したデータセット(IT, S)では，ユニークカテゴリのデータセット(U)よりも少ない数でスコアの最大値を達成している傾向がある．IT や S のデータセットでは各クラスタが意味的に近くに存在しており，関連エンティティを多く用いると，別のクラスタと繋がってしまうためである．また，ESA をベースとした提案手法では顕著にその傾向が現れている．このことから，特に ESA ベースの提案手法では，上位の関連エンティティほどコンテキストに強く依存しており，下位になるにつれて徐々にコンテキストから離れた関連エンティティになるという，順位付きリストとしてより適切な状態で出力を得られることが分かる．

## 5. まとめと今後の課題

本研究では，Wikipedia とベイズ理論を用いた枠組みにより，自然文の入力に対して関連エンティティを推測する手法を提案した．具体的には，キーフレーズ（特徴語）の抽出，多義語の曖昧性解消（エンティティリンキング），入力語句に対する関連エンティティの取得といった従来は個別に扱われていた問題に対して，Wikipedia から抽出可能な情報をベイズ理論の枠組みに落とし込み，ナイーブベイズを拡張した手法により一つの問題として取り扱った．これにより，煩雑なパラメータ調整が不要となり，安定した精度で関連エンティティの推測が可能となる．Twitter の投稿メッセージのクラスタリングによる評価結果から，提案手法によって得られた関連エンティティを素性として用いることにより，既存手法よりも高い精度で短文クラスタリングが可能であることを確認した．

今後の課題として，入力テキストの曖昧性解消が挙げられる．提案手法では自然文で与えられた入力に対して関連エンティティを推測しているが，この推測された関連エンティティをフィードバックさせることで，入力テキストに含まれるエンティティを特定できると考えられる．

### [謝辞]

本研究の一部は，科学研究費補助金基盤研究 B(21300032)，および日本学術振興会特別研究員奨励費(24-807)の助成によるものである．ここに記して謝意を表す．

### [文献]

[1] Gabrilovich, E. and Markovitch, S.: “Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis,” in Proceedings of International Joint Conference on Artificial Intelligence (IJCAI), pp.1606-1611 (2007).

[2] Ito, M., Nakayama, K., Hara, T. and Nishio, S.: “Association Thesaurus Construction Methods based on Link Co-occurrence Analysis for Wikipedia,” in Proceedings of ACM Conference on Information and Knowledge Management (CIKM), pp.817-826 (2008).

[3] Laniado, D. and Mika, P.: “Making Sense of Twitter,” in Proceedings of International Semantic Web Conference (ISWC), pp.470-485 (2010).

[4] Mihalcea, R. and Csomai, A.: “Wikify! Linking Document to Encyclopedic Knowledge,” in Proceedings of ACM Conference on Information and Knowledge Management (CIKM), pp.233-241 (2007).

[5] Milne, D. and Witten, I.H.: “An Effective, Low-Cost Measure of Semantic Relatedness Obtained from Wikipedia Links,” in Proceedings of AAAI Workshop on Wikipedia and Artificial Intelligence (WIKIAI), pp.25-30 (2008).

[6] Milne, D. and Witten, I.H.: “Learning to Link with Wikipedia,” in Proceedings of ACM Conference on Information and Knowledge Management (CIKM), pp.509-518 (2008).

[7] Nakayama, K., Hara, T. and Nishio, S.: “Wikipedia Mining for An Association Web Thesaurus Construction,” in Proceedings of International Conference on Web Information Systems Engineering (WISE), pp.322-334 (2007).

[8] Ponzetto, S.P. and Strube, M.: “Exploiting Semantic Role Labeling, WordNet and Wikipedia for Coreference Resolution,” in Proceedings of Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL), pp.192-199 (2006).

[9] Shirakawa, M., Wang, H., Song, Y., Wang, Z., Nakayama, K., Hara, T. and Nishio, S.: “Entity Disambiguation based on a Probabilistic Taxonomy,” Tech. Rep. MSR-TR-2011-125, Microsoft Research (2011).

[10] Song, Y., Wang, H., Wang, Z., Li, H. and Chen, W.: “Short Text Conceptualization Using a Probabilistic Knowledgebase,” in Proceedings of International Joint Conference on Artificial Intelligence (IJCAI), pp.2330-2336 (2011).

[11] Strube, M. and Ponzetto, S.P.: “WikiRelate! Computing Semantic Relatedness using Wikipedia,” in Proceedings of National Conference on Artificial Intelligence (AAAI), pp.1419-1424 (2006).

**白川 真澄 Masumi SHIRAKAWA**

大阪大学大学院情報科学研究科博士後期課程在学中．2010 年大阪大学大学院情報科学研究科博士前期課程修了．Web マイニングに関する研究に従事．情報処理学会学生会員．

**中山 浩太郎 Kotaro NAKAYAMA**

東京大学知の構造化センター特任講師．2007 年大阪大学大学院情報科学研究科博士後期課程修了．同年 4 月から大阪大学大学院情報科学研究科特任研究員．人工知能，Web マイニングに関する研究に従事．IEEE，ACM，情報処理学会，電子情報通信学会，人工知能学会各会員．

**原 隆浩 Takahiro HARA**

大阪大学大学院情報科学研究科准教授． 1997 年大阪大学大学院工学研究科博士前期課程修了．工学博士．データベースシステム，分散処理の研究に従事．IEEE，ACM，情報処理学会，電子情報通信学会各会員．

**西尾 章治郎 Shojiro NISHIO**

大阪大学大学院情報科学研究科教授．1975 年京都大学工学部卒業．1980 年同大学院工学研究科博士後期課程修了，工学博士．京都大学工学部助手等を経て，1992 年大阪大学工学部教授となり，現職に至る．文部科学省科学官，大阪大学理事・副学長等を歴任．データベース，マルチメディアシステムの研究に従事．本会より功労賞，論文賞を受賞．本会理事，監事を歴任．