恣意的に名前付けされたオブジェクトの識別手法

高橋　良平† 　小山　聡†† 　田中　克己††

† 京都大学工学部情報学科　〒 606–8501 京都府京都市左京区吉田本町
†† 京都大学大学院情報学研究科社会情報学専攻　〒 606–8501 京都府京都市左京区吉田本町
E-mail: †{takahasi,oyama,tanaka}@dl.kuis.kyoto-u.ac.jp

あらまし　料理や事件といったオブジェクトは，書き手によって恣意的に名前が付けられるため，同一オブジェクトに対して無数の名前が存在する．そのため，あるページがどのオブジェクトについての記述であるかを識別することは容易ではない．そこで本研究では，まず多くの人が使用している名前のパターンをもとにオブジェクトを抽出し，次にそのオブジェクトに分類されるための規則を作成し，最後にその規則をもとに各ページが指しているオブジェクトを同定する，という三段階でこの問題を解決する手法を提案する．

キーワード　オブジェクト識別，クラスタリング

1. はじめに

近年，Yahoo! [1] や Google [2] のようにページ単位で検索するのではなく，オブジェクト単位で検索する“ オブジェクト検索 ”に関する研究が多数行われている [3] [4]．オブジェクト検索では，オブジェクトについての記述を含む複数ページからスキーマに従って情報を抽出し，オブジェクト単位で情報を集約してユーザに提示する．例えば人物を対象としたオブジェクト検索の場合，同一人物に関する情報を Web から取得し，生年月日や連絡先などの属性を集約してユーザに提示する．人物だけでなく，論文や商品についても，このようなオブジェクト単位の検索は行われている．

このようなことを実現するためには，あるページの記述と，もう一つのページの記述が，同一オブジェクトについて書かれているかを判断する必要がある．これは,“ オブジェクト識別 ”と呼ばれている問題である．オブジェクト識別に関する研究は従来から行われているが，その多くは名前の曖昧性の解消に関するものであった．例えば，人物を対象としたオブジェクト識別の場合，同じ名前の人物について書かれているページであっても同姓同名の別人である可能性があり，それらを実際の人物ごとに分離するという問題である．ある人物に付けられる名前は，別名を持つ場合などの例外を除くと一つのみであるので,“ 名前が異なれば異なるオブジェクトである ”という仮定が成り立つ．そのため，まずデータを名前ごとに分け，その後，同じ名前を持つデータについてクラスタリングを行うことで同姓同名の問題を解消するという手法が一般的である [5]．また，同一人物が複数の名前をもつ場合には，別名は多くても数個程度のため，それぞれの名前ごとに集約した後に，属性情報が酷似していれば同一人物とみなすことができる．

人物などのオブジェクトを識別する際のの問題点は，主に名前の曖昧性の解消であった．しかし，それ以外のオブジェクトを識別する際には，別の問題が起こることもある．本研究で提起する問題は，名前が恣意的に付けられるオブジェクトの識別問題である．

例えば，料理レシピの場合，同じような料理レシピに対して,“ 本格タイ風グリーンカレー ”や“ 我が家のグリーンカレー ”といったように，様々な名前が付けられている．また，ニュースサイトでは，同一の事件に対して,“ ○○市の事件 ”や“ △△さんの事件 ”といったようにニュースサイトごとに異なる名前が付けられる．これらの名前は，書き手によって恣意的に付けられるものであるので，同一のオブジェクトに対して名前が多数存在する．このような恣意的に名前付けされたオブジェクトを識別する際には，次の二つの問題点がある．

一つ目は，名前が恣意的であるために，付けられた名前の中に，集約されるオブジェクトとしてふさわしいくないものも含まれるという問題である．例えば,“ グリーンカレー ”と名付けられた料理レシピは多数あるが，グリーンカレーはタイ料理の一つであり，オブジェクトとして集約するのが妥当であると考えられる．一方,“ 我が家のカレー ”と名付けられた料理レシピもいくつかあるが,“ 我が家 ”というのは単にその人の家庭で作ったという意味であり，オブジェクトとして集約されるべきではないと考えられる．

二つ目は，同一オブジェクトに多数の名前が存在するという問題である．例えば,“ 我が家のカレー ”と名前が付けられていても，実際にはグリーンカレーである場合もある．このような場合でも正しく判定できないと，実際にはグリーンカレーのものが正しくグリーンカレーと判断されず，集約の際に情報が少なくなってしまったり，検索の際に再現率が下がってしまうなどの問題が起こってしまう．

本研究の目的は，恣意的に名前が付けられたオブジェクトを対象として扱い，それぞれのページがどのオブジェクトについて書かれたものであるかを同定することである．そのためには，世の中にどのようなオブジェクトが存在するのかということと，それぞれのオブジェクトの持つべき性質は何であるのかの２つを自動で取得する必要がある．

料理レシピなどの場合，どの程度似ていたら同一オブジェクトとみなすかが，オブジェクトにより異なる．例えば，肉の種類はグリーンカレーであるかどうかには重要ではないが，チキ

ンカレーであるかどうかには重要である．そのため，あらかじめ材料などの属性の類似度の閾値を決めて、同一かどうかを判断するということは難しい．

そこで，本研究では，多くの人が同一と考えているものを同一オブジェクトとみなす．具体的には，多くの人の名前の付け方のパターンをもとに分類階層を抽出し，これ以上細かく分類されていないようなものを，オブジェクトとして集約の単位とする．

本研究で提案する手法では，以下の三段階からなる．まず，名前のパターンをもとに，分類階層を抽出し，その中からオブジェクトとなるべき分類を決定する．次に，それぞれの分類に入るために，属性が満たすべき規則を求める．最後にその規則をもとに，各ページがどのオブジェクトについての記述であるかを同定する．

本研究では，恣意的に名前付けされたオブジェクトの具体例として，料理レシピについて取り上げ説明する．

2. 関 連 研 究

オブジェクト識別に関する研究のうち，名前の曖昧性の解消についての研究は従来から多数行われている．例えば, [6] [7] [8] では，属性情報などをもとに同姓同名を分離し，実際の人物ごとに集約を行っている．

また，同一オブジェクトの別名について扱っている研究もある. [9] では，ある人物の正式な名称以外の呼び名（ニックネームなど）を Web から抽出している．

本研究で扱う，恣意的に名前付けされるオブジェクトは，同一オブジェクトに名前が多数存在するという点で，別名や表記ゆれの研究などに似ている．しかし，別名や表記ゆれは，同一の名前が多くの人によって使われるのに対し，恣意的に付けられた名前の場合は，一人の人しか使っていない場合も多いという点で異なる．例えば,“ 京都大学 ”の別名である“ 京大 ”は多くの人が使用する名前であるが,“ グリーンカレー ”のレシピに付けられた“ 本格タイ風グリーンカレー ”という名前は一人しか使用していないということもある．また，別名の場合はせいぜい数個しか存在しないのに対し，恣意的に付けられた名前の場合は，名前が無数に存在しうるという点でも異なる．

名前が恣意的に付けられるオブジェクトのうち，事件についてのオブジェクト識別についての研究は既に行われている. [10] [11] では，全ニュース記事に対して，名前の情報を用いず，記事の内容だけで階層的にクラスタリングを行うことで同一の事件に関する記事を集め，さらに要約の自動生成も行っている．具体的には，様々なクラスタリング手法について，閾値を動かしながら実験することで，ニュース記事に最適な手法を見つけている．また，名前については，クラスタの中で最も典型的な記事についている名前を与えている．

本研究では，このようにページの内容だけを用いるのではなく，付けられた名前のパターンと，ページの内容の両方を用いて，オブジェクトの識別を行う．次章以降では料理レシピを具体例として説明する．

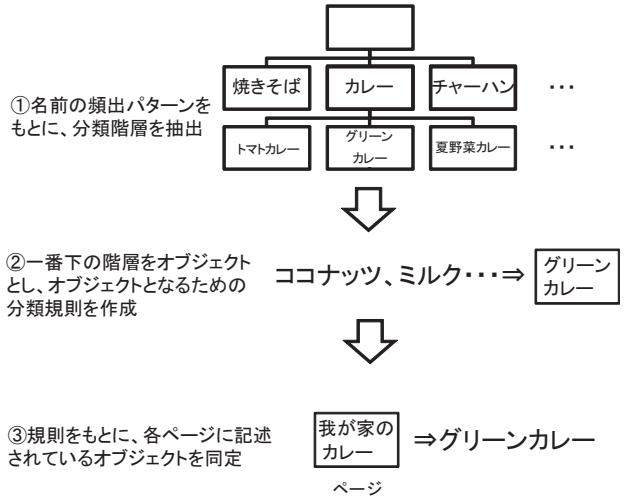

図 1　提案手法の流れ

3. 本研究の目的

本研究の目的は，オブジェクトについて書かれたページが多数あり，それぞれのページでオブジェクトに名前が恣意的に付けられている場合に，各ページに書かれたオブジェクトが実際には何であるのかを同定することである．料理の場合，Web 上の多数の料理レシピについて，それぞれのレシピがどの料理について書かれたものかを同定するということである．

3.1 提案手法の流れ

１章で述べたように，どの程度似ていれば同一オブジェクトとみなされるかがオブジェクトによって異なるため，あらかじめ属性の類似度の閾値を決めてどのオブジェクトであるかを同定することは難しい．そこで本研究では，まず多くの人の名前の付け方のパターンをもとに分類階層を抽出し，これ以上細かく分類されていないものをオブジェクトとして扱う．

本研究で提案する手法の流れは，大きく分けて以下の三段階から構成される．(図 1)

（1） 名前の頻出パターンをもとに，分類階層を抽出する

（2） 一番下の階層をオブジェクトとし，オブジェクトとなるために属性が満たすべき規則を作成する

（3） 規則をもとに，各ページがどのオブジェクトであるかを同定する

3.2 抽出対象とする分類

本稿では，以下の３つの性質を持つものを“ 分類 ”として抽出する．

＜抽出対象とする分類＞

（1） その分類に含まれるデータ数が十分にある

（2） 分類に含まれるデータ同士にある程度共通の特徴がある

（3） その特徴が既存の分類の特徴とは異なっている

１．の条件は，その分類がオブジェクトとしてどの程度認識されているかの指標である．その分類があまり認識されていないということは，その分け方が意味のある分け方だと考えている人が少ないということを表していると考えられる．

２．の条件は，共通性の低いものを取り除くためである．分散の大きいものも分類と考えることもできるが，オブジェクトの識別という観点からこのようなものはふさわしくないと考えられる．

３．の条件は，冗長な分類を取り除くためである．すでに分類として認識されているものがある場合，それと同じ特徴をもつ分類を新たに作る必要はないためである．

たとえば料理の場合,“ グリーンカレー ”に分類されるレシピにはいくつかの共通の性質があり，通常のカレーとも異なっており，数も十分にあるため，分類名としてふさわしいとして扱う．一方,“ 我が家のカレー ”の場合，それぞれのレシピは通常のカレーとは異なり，数も十分にあるが,“ 我が家のカレー ”の間には共通の性質はほとんどないため,“ 我が家のカレー ”は分類名としてはふさわしくないとして扱う．

4. 提 案 手 法

4.1 分類階層の抽出

分類の抽出は以下の２つの Step によって行う．

＜抽出手順＞

Step1:名前の一部が共通であるものを集め，分類の候補とする

Step2:分類の候補に含まれるデータについて，分類内の相関と，他の分類との差異を計算することで，分類であるかどうかを決定する

まず Step1 で，名前をもとにデータ数が十分にあること（分類の条件１.）を満たすものを集めてきて分類の候補とし，Step2 で共通の性質を持つか（条件２.），既存の分類と異なる性質であるか（条件３.）を満たしているかを調べ，全てを満たしているものを分類として抽出する．これらについて，次節以降で順番に説明していく．

なお，以下では,“ データ ”という語を用いるが，各ページに書かれているオブジェクトの名前と属性情報の組という意味で用いる．つまり，一つのページが一つのデータに対応する．

4.1.1 分類候補の抽出

まず，データの名前に出現する部分文字列の出現頻度を用いて分類名候補を抽出する．これは，ある程度のデータには正しい分類名が含まれているという仮定に基づいている．また,“ 複数語からなる分類名の場合，後ろに出現する語ほど上位階層を表す語である ”という仮定を用いて，階層的に分類する．たとえば,“ グリーンカレー ”は“ カレー ”の階層の一つ下の階層になるように分類する．具体的な手法は以下のとおりである．

＜抽出手順＞

（1） 木の根にすべてのデータを入れる

（2） それぞれのデータの名前 N_i を単語ごとに分け，$n_{ik}n_{i(k-1)}\cdots n_{i1}$ とする

（3） $n_{i1}, n_{i2}n_{i1}, \cdots, n_{ik}n_{i(k-1)}\cdots n_{i1}$ の節点にそれぞれにデータを入れる（節点がなければ新しく作る）

（4） n_{i1} の子に $n_{i2}n_{i1}$ ，$n_{i2}n_{i1}$ の子に $n_{i3}n_{i2}n_{i1}$，$\cdots$ となるように枝を張る

（5） 以上のことを全データの名前について行った後，節点に含まれるデータの数が閾値 θ 以上の節点だけを残す

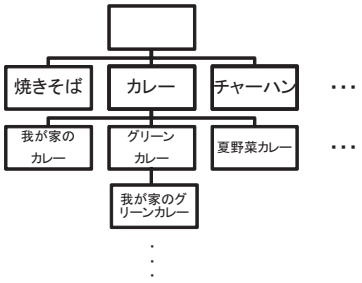

図 2 階層的な分類候補

	属性１	属性２	属性３	属性４	計
集合 A	w_1	w_2	w_3	w_4	b_1
集合 B	w'_1	w'_2	w'_3	w'_4	b_2
計	a_1	a_2	a_3	a_4	S

表１ 集合Aと集合Bに有意差があるかを調べる際のカイ 2 乗検定（図は自由度３の例）

このようにすることで，図 2 のような階層構造が得られる．そのそれぞれの節点が分類候補となる．

4.1.2 分類の判定

Step1 で抽出される分類候補に含まれるデータは，名前に共通性があるだけである．次に，それぞれの分類候補が条件２．と条件３．を満たしているかを調べる．

階層の上位のものから順番に分類候補を選択し，分類であるかを順次決定していく．以下では，選択された分類候補が，分類かどうかを判定するアルゴリズムを示す．

まず，選択した分類候補（集合 A と呼ぶ）に含まれる全てのデータの属性から特徴ベクトル$\boldsymbol{w}$を求める．特徴ベクトル$\boldsymbol{w}$の属性 t_j の重み w_j の値は以下の式で求める．

$$w_j = \begin{cases} \mathrm{DF}_A(t_j)(\mathrm{DF}_A(t_j) > \alpha N) \\ 0(otherwise) \end{cases} \quad (1)$$

ここで，$\mathrm{DF}_A(t_j)$ は集合 A の中で属性 t_j を含むデータの総数，N は集合 A に含まれる全データの総数，α は 1 未満の係数である．つまり，集合内の属性に一定の割合 α より多く出現するものだけを取り出し，共通の特徴としている（条件２）．

もう一つ，比較対象として，親ノードのデータから集合Aに含まれるデータを除いたもの（集合 B と呼ぶ）に含まれるすべてのデータの属性に関する特徴ベクトル$\boldsymbol{w}$’ を求める．特徴ベクトル$\boldsymbol{w}$’ の属性 t_j の重み w'_j の値は以下の式で求める．

$$w'_j = \begin{cases} \mathrm{DF}_B(t_j)(w_j > 0) \\ 0(otherwise) \end{cases} \quad (2)$$

ここで，$\mathrm{DF}_B(t_j)$ は集合 B の中で属性 t_j を含むデータの総数である．つまりここでは，集合 A の中に一定数以上出現する属性だけについて，集合 B 内での DF 値を求めている．

次に，集合 A と集合 B の間に有意差があるかどうかを，有意水準 p，自由度 d-1 のカイ２乗検定を用いて調べる．なお，d

は，特徴ベクトル$\boldsymbol{w}$の中の 0 でない要素の数のことである．そして，ここで有意差があると判断されれば，集合 A は適切な分類と判断できる（条件３）．具体的には次式によりカイ２乗値を求める.(表 1)

$$\chi^2 = \sum_{i=1}^{d}\sum_{j=1}^{2}\frac{(w_{ij} - a_i b_j/S)^2}{a_i b_j/S} \tag{3}$$

ここで,

$$w_{i1} = w_i, w_{i2} = w_i', a_i = w_i + w_i',$$
$$b_1 = \sum_{k=1}^{d} w_k, b_2 = \sum_{k=1}^{d} w_k', S = b_1 + b_2$$

である.

この式で求めたカイ２乗値が有意水準 p，自由度 d-1 のカイ２乗値よりも大きければ A と B の集合の間に有意差があると判断できる．集合 A と集合 B の間に有意差がないと判断された場合は，対象としている分類候補が一つ上の階層のものとはあまり変わらないという意味であるので，新たにこの分類を作る必要はないと判断され，子ノードとともに分類木から削除する．これを繰り返し，最終的に残ったものが分類階層となる．

4.2 分類規則の作成

前節で得られた階層木のうち，一番下の階層をオブジェクトとして扱う．次に，各オブジェクトに分類されるための規則を求める.

前節で集合間の有意差を求めたが，今度はどの出現頻度において有意差があったのかを調べるために，各属性ごとに集合 A と集合 B とでの出現頻度の違いを表 2 のような有意水準 p，自由度１のカイ 2 乗検定で調べ，有意差があった属性だけを分類規則として記憶しておく．具体的には次式によりカイ 2 乗値を求め，各属性が分類規則に含まれるかを判定する．比較対象となる集合Bは先ほどと同様，一つ上の階層のものである.

$$\chi^2 = \sum_{i=1}^{2}\sum_{j=1}^{2}\frac{(w_{ij} - a_i b_j/S)^2}{a_i b_j/S} \tag{4}$$

ただし,

$$w_{i1} = w_i, w_{i2} = w_i', a_i = w_i + w_i',$$
$$b_1 = w_1 + w_2, b_2 = w_1' + w_2', S = b_1 + b_2$$

である.

この式を満たす属性を，すべて分類Aの分類規則として記憶しておく．この分類規則を，すでに分類と判定されたものの分類規則と比較し，互いに全く同じであれば，その２つは同一の分類であるとみなして統合する.

一番上位の階層の分類について調べた後は，一つ下の階層についても先ほどと同様に調べる．これを一番下の階層まで再帰的に繰り返すと，適切な分類だけが残る.

前節と本節の内容を，図 2 の階層木から,“ グリーンカレー ”が分類として判定される過程で説明する（図 3).

	属性１を含む	属性１を含まない	計
集合 A	w_1	w_2	b_1
集合 B	w_1'	w_2	b_2
計	a_1	a_2	S

表 2　属性１が分類規則に含まれるか判定する際のカイ 2 乗検定

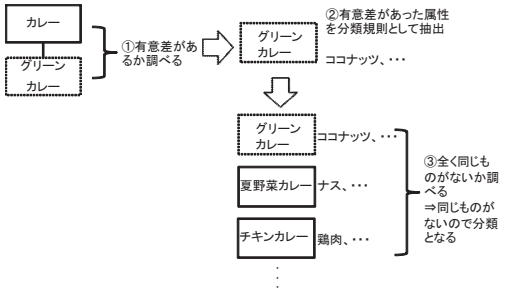

図 3　グリーンカレー（点線）が分類として判定される過程

まず,“ グリーンカレー ”に分類されているすべてのレシピを集め（集合 A），その中で一定の割合以上のレシピに含まれる語をもとに特徴ベクトルを作る．なお，Step1 での木の作り方から,“ 我が家のグリーンカレー ”のように“ グリーンカレー ”の下の階層に含まれるレシピも“ グリーンカレー ”の分類に含まれている.

“ グリーンカレー ”の比較対象となるのは,“ グリーンカレー ”以外のすべてのカレーであり（集合 B），集合 A の中で一定数以上出現する語だけを用いて，集合 A と B の間に有意差があるかどうかを調べる（図 3① ）．有意差がなければ集合 A は分類ではないとみなして削除する．実際には有意差があるので，どの属性について有意差があったのかを調べ，有意差があった属性を分類規則として記憶する．例えば，ココナッツやミルクなどが分類規則となる.（図 3② ）．次に,“ グリーンカレー ”の分類規則を，兄弟ノードである“ 夏野菜カレー ”や“ チキンカレー ”の分類規則と比較する（図 3③ ）．もしそのうちのどれかと完全に一致すれば二つを同一の分類とみなし統合されることとなるが，実際は同一の分類規則をもつノードは存在しないので，最終的にグリーンカレーは分類として判定される.

4.3 分類規則を用いたデータの同定

これまでの段階により，オブジェクトとそれに分類されるための規則が求まった．最後に，各ページがどのオブジェクトについての記述であるのかを属性の情報のみで同定する．この際，データに付けられた名前の情報は利用しない．それは，名前が恣意的に付けられているため，あるオブジェクト名が付けられているページであっても，それが間違いである可能性があるためである.

前節で求めた分類規則は，その分類を特徴づける属性が抽出されている．そこで，各ページが，分類規則に含まれる属性を全て含んでいるかによって，その分類に分類されるべきかどうかを判定する．例えば“ グリーンカレー ”の分類規則には，ココナッツやミルクなどがある．そこで，それぞれのレシピがこれらの属性を含んでいれば“ グリーンカレー ”と判断する．以

上のようにして同定を行う．

なお，その際，あるページが複数の分類の分類規則を満たすことも考えられるが，その場合も複数の分類に同定してよい．例えば，ある料理レシピが“ トマトカレー ”でもあり“ チキンカレー ”でもある場合もあるからである．

5. 実験と考察

実験では，恣意的に名前付けされたオブジェクトの具体例として，料理について取り上げ，得られる料理名の数とその精度の関係に関する実験と，各レシピがどの料理について書かれたものなのかを同定する実験の２つを行った．

実験で使用したデータは，投稿型レシピサイトの一つであるクックパッド [12] から，カレー，ハンバーグ，ヤキソバのレシピ計 5,894 件のレシピを取得したものである．その中には全部で 4,826 種類の名前が存在した．それぞれのレシピは名前，材料や作り方などで構成されている．このうち，食材，調理器具，調理動作などが各レシピの属性となるように，各レシピの名前以外の本文を，形態素解析器 MeCab [13] を用いて名詞と動詞を切り出し，それらをレシピの特徴ベクトルとした．つまり，“ ナス ”などの材料や“ 煮る ”などの動作などの語が，特徴ベクトルの属性となる．なお，必要に応じて辞書を作成した．

5.1 得られる料理名の数と精度の関係

一つ目の実験では，4 章で挙げた手法に使われる各種パラメータ α ，θ ，p を変えることによって，得られる料理名の数がどのように変わるかについて調べた．なお，α は各分類の何割以上に含まれていれば分類共通の属性とみなすかの値，θ は分類とみなす最低のインスタンス数，p はカイ 2 乗検定を行う際の有意水準である．θ を動かすことによって得られる分類数の違いは，他のパラメータとは無関係であるので，今回は θ を 5 に固定して α と p を動かし，得られる料理名の数の違いを調べ，図 4 と図 5 に示した．また，$\alpha = 0.4$, 有意水準 0.1 %の時に得られたカレーを図 6 に示す．

図 4 の縦軸は，全レシピについて提案手法を適用し，最終的にオブジェクトと判断されたもののうち，人手で正解と判断されたものの数のことである．また，図 5 の縦軸の適合率とは，提案手法が出力した結果のうち，人手で正しいと判断された割合のことである．有意水準は，一般的に統計学でよく使われる 1 %と 0.1 %の２つについて行い，α は 0.2，0.4，0.6，0.8 の４つについて行った．

まず，有意水準 p に関してであるが，有意水準を低くする（＝判定を厳しくする）と適合率は上がるが得られる料理名の数が少なくなるという傾向はあるものの，その数字にそれほど差は出ないことがわかる．

次に，α について考察する．α の値は，分類の何割以上に含まれていれば分類共通の属性とみなすかについての値である．つまり，α より低い割合でしか現れない属性語は，検定の際には使用されないということである．α が大きい場合には，その分類にかなり頻繁に現れる属性しか使用しないため，適合率は上がるものの得られる料理名の数は少ない．一方，α が小さい場合には，検定に使用する語が増えるために，有意差が現れやすくなるが，その分類の特徴とは直接関係のない属性に影響されて精度が下がってしまうことがある．そのため，適合率が少し下がってしまう．

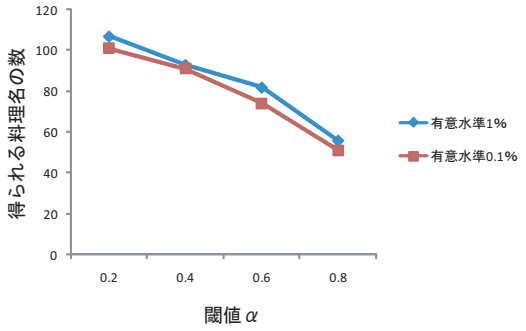

図 4 α ，p と得られる料理名の数

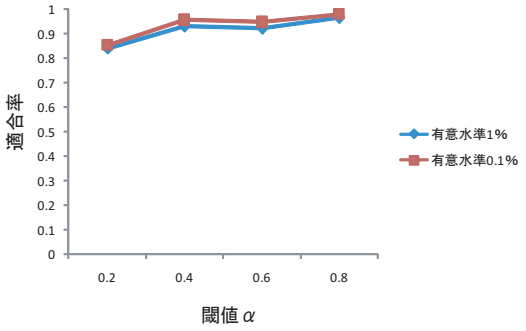

図 5 α ，p と適合率の関係

ドライカレー	エビカレー
チキンカレー	ゴーヤカレー
キーマカレー	大根カレー
グリーンカレー	イエローカレー
スープカレー	ミンチカレー
トマトカレー	ヨーグルトカレー
タイカレー	ツナカレー
豆カレー	キャベツカレー
夏野菜カレー	キノコカレー
ヒキ肉カレー	ミートボールカレー
牛スジカレー	トウフカレー
ナスカレー	チーズカレー
和風カレー	オムカレー
シーフードカレー	ミソカレー
焼カレー	オカラカレー
ホウレンソウカレー	ゴボウカレー
ココナッツカレー	サバカレー
ポークカレー	甘口カレー
レッドカレー	豆ドライカレー
インドカレー	オカラドライカレー
ビーフカレー	和風ドライカレー
ヒキニクカレー	豆キーマカレー
カボチャカレー	

図 6 $\alpha = 0.4$, 有意水準 0.1 %の時に得られたカレー

また，本手法と比較するために，簡単なベースライン手法を用意した．具体的には，単純に５回以上出現する名前のパターン全てを，料理名として出力するものである．ベースライン手法の適合率は 73.8 %であったため，本手法を適用することにより適合率が約 20 %ほど上がると言える．一方，ベースライン手法で得られた料理名の数は 124 であったため，本手法では 20 個程度得られなかったことがわかる．その理由については，次々節で説明する．

5.2 各ページに書かれているオブジェクトの同定

各ページに書かれているオブジェクトを同定する際には，分類規則を使用する．今回は，比較の際に出現頻度に有意差があった属性を取り出し，あるページの中にその属性がすべて現

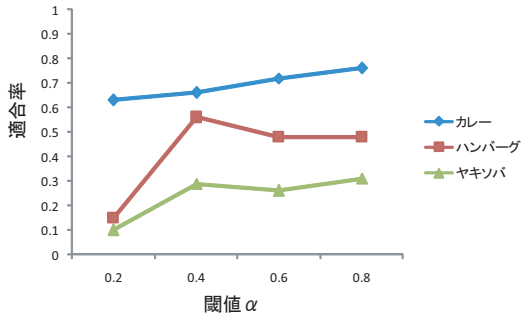

図 7　各分野における，レシピの適合率の平均

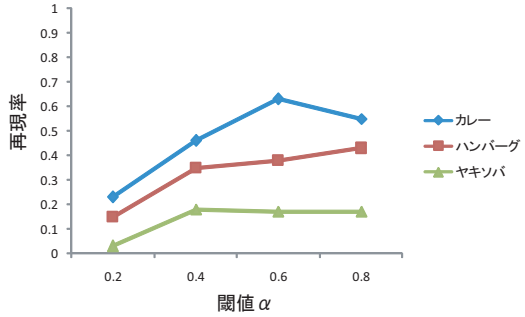

図 8　各分野における，レシピの再現率の平均

れれば，そのオブジェクトとして同定するということを行う．前節での実験により，有意水準はそれほど結果に影響を与えないということがわかったので，有意水準は 0.1 %に固定し，α の値を先ほどと同様に動かし，同定の精度がどのように変化するのかを調べた．精度の評価方法は以下のようにして行った．

まず，カレー，ハンバーグ，ヤキソバの 3 つの分野について，それぞれレシピを 30 件ずつ用意する．次に，各レシピに書かれているオブジェクトを人手で同定し，それを正解とする．この際，例えばあるカレーのレシピが，“ ポークカレー ”でもあり，“ ゴーヤカレー ”でもあるという場合もあるが，その際にはその全てを正解とする．その後，システムにそれぞれのレシピを入力として与え，提案手法を使って同定させる．そして，以下の 2 つの尺度により評価する．

- レシピの適合率
  - システムが同定したオブジェクト名のうち，正解の割合
- レシピの再現率
  - 正解のうち，システムが同定したオブジェクト名の割合

そして，各分野ごとに，30 個のレシピの適合率・再現率の平均を取り，その結果を図 7，8 に示した．図から分かる通り，カレーの分野では，$\alpha = 0.6$ のときに，適合率の平均が約 7 割，再現率の平均が約 6 割と，ある程度の精度となっている．しかし，他の二つの分野，特にヤキソバでは適合率の平均が 3 割程度，再現率の平均が 2 割程度とかなり低いものとなってしまっている．その理由については次節で説明する．

5.3　実験結果の考察

本論文での提案手法では，それぞれの分類候補が分類であるかどうかを判定する際に，階層木における一つ上の階層の分類と，対象となっている分類候補の 2 つの集合の間に，有意差があるかどうかをカイ 2 乗検定を用いて判定を行った．そして，その際，分類候補に比較的頻繁に現れる属性のみを用いて検定を行っている．そのため，例えば，“ 肉が入っていないこと ”が条件である“ 野菜カレー ”のように，特定の属性が含まれていないということが条件となるような分類の場合，分類規則はうまく抽出することができない．それは，“ 野菜カレー ”には“ 肉 ”という属性がほとんど存在せず，現在の手法のままでは比較の際の属性として用いられないためである．

この問題を解決するために，2 つの集合間の比較をする際に使用する語を増やすということが考えられる．具体的には，分類候補内の頻出属性だけを用いるのではなく，比較対象となる分類の頻出属性も用いるということである．このようにすれば，例えば先ほどの例では，“ 肉 ”も比較の際の属性に含まれるようになるので，“ 野菜カレー ”が分類として抽出されるようになる可能性がある．

また，現在では，各記述がどの分類についての記述であるかの判定を行うために，分類規則を求めている．現在の手法では，比較の際に有意差があった属性を分類規則とし，そのすべてを含んでいるページのみをその分類に同定している．しかし，前節で見たとおり，その同定の精度は，分野によっては低くなってしまう．

分類規則に含まれる属性が一つのときには，その属性が含まれるかどうかによって分類の同定ができる．例えば，“ トマトカレー ”はカレーにトマトが入っていればよく，このような例は本手法で正しく同定できる．カレーの分野では，このような例が多かったために，他の分野と比べて適合率・再現率が高くなったと考えられる．

しかし，現実の分類規則は，このような単純なものばかりではない．例えば，“ 和風ハンバーグ ”では，醤油を使うものと大根おろしを使うものの大きく 2 つが存在する．このような場合，これらの材料のうちどちらかが含まれていればよいのであるが，現在の手法では，どちらも含んでいなければ和風ハンバーグと同定されないので，規則が厳しすぎる．そのため，レシピの再現率が下がってしまう．また，“ 塩ヤキソバ ”の特徴は塩であるが，塩が入っているヤキソバであれば塩ヤキソバと呼べるというわけではなく，ソースが入っていてはいけない．現在の手法では，塩が入っているヤキソバをすべて塩ヤキソバとしているため，適合率が下がってしまっている．

ここで具体例として，表 3 から表 5 に，α の値が 0.8 と 0.6 場合で，実際に抽出された分類規則に含まれる属性を掲載する．また，参考のために，それぞれの属性について，有意差があるかどうかを判定した際のカイ 2 乗値も載せている．この値が大きいほど，その属性が分類を特徴づけているということを意味する．

一つ目の“ トマトカレー ”の場合，α の値がどちらの場合も抽出された属性はトマトだけであった．これをそのまま分類規則にすると，トマトが入っていればトマトカレーであると判断することになるが，これは妥当であると考えられる．

二つ目の焼カレーと三つ目のタイカレーの場合，α の値によって得られる属性が異なった．

二つ目の“ 焼カレー ”の場合，$\alpha = 0.8$ のときに得られた

$\alpha = 0.6$		$\alpha = 0.8$	
属性	カイ 2 乗値	属性	カイ 2 乗値
トマト	675.3	トマト	675.3

表 3　トマトカレーの分類規則

$\alpha = 0.6$		$\alpha = 0.8$	
属性	カイ 2 乗値	属性	カイ 2 乗値
オーブン	1049	オーブン	1049
チーズ	336.4	チーズ	336.4
焼く	131.7	焼く	131.7
皿	83.8	米	54.4
米	54.4		
かける	25.3		

表 4　焼カレーの分類規則

$\alpha = 0.6$		$\alpha = 0.8$	
属性	カイ 2 乗値	属性	カイ 2 乗値
タイ	690.6	ココナッツ	332.0
ココナッツ	332.0	ミルク	310.8
ミルク	310.8	ペースト	292.3
ペースト	292.3		

表 5　タイカレーの分類規則

“ オーブン ” チーズ ” 焼く ” 米 ”などの属性語は，焼カレーに必須である属性だと考えられ，妥当である．ところが，$\alpha = 0.6$ のときに得られる“ 皿 ”や“ かける ”という語は，特に必須であるというわけではない．つまり，偶然出現頻度に差があったために抽出されてしまった語である．そのため，このような属性を規則に入れてしまうと，本来焼カレーであるにも関わらず正しく同定されないということがおこる．

三つ目の“ タイカレー ”の場合，$\alpha = 0.8$ のときに得られた“ ココナッツ ” ミルク ” ペースト ”などの属性はほぼ必須の属性であり，妥当である．しかし，$\alpha = 0.6$ のときに得られる“ タイ ”という語は確かにタイカレーを説明する際によく出てくる語ではあるだろうが，属性とは言えない．単純に形態素解析を行っているだけなので、属性ではないものまで結果として出てきてしまっている．このような場合も先ほどと同様に,“ タイ ”という語を含んでいないと“ タイカレー ”とみなされず，本来の目的とは異なるものとなってしまう．

α の値は，分類の何割以上に含まれている属性を共通の属性とみなすかについてのパラメータであった．そのための値を小さくすると，分類にそれほど出現しないような属性であっても，出現頻度に有意差があれば分類規則として抽出されてしまう．そのため，α の値を小さくする場合，分類規則であるかを判定する際には，単純に有意差がある属性を取るだけではなく，分類内での属性の出現頻度と組み合わせるなどの工夫が必要であると考えられる．

6. 本手法と従来手法との違い

本研究では，付けられた名前の頻出パターンを利用して，分類階層を抽出するということを行った．一方，データを分類する代表的な従来手法として，クラスタリングと分類問題の二つの手法がある．本章では，これら二つの手法との違いについて考察する．

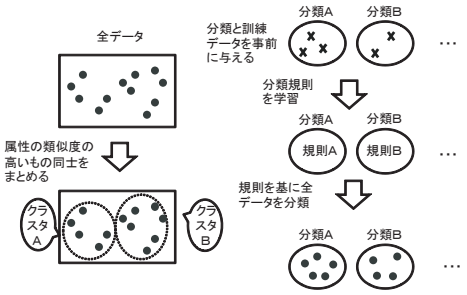

図 9　従来のクラスタリング（左）と分類問題（右）

クラスタリングは，類似度の最も高いクラスタ同士を再帰的に集約していくことによってクラスタを生成する（図 9 左）．クラスタリングの手法の一つである，階層的なクラスタリング手法を利用すれば，属性の類似度に基づくデータの階層構造を抽出することができる．しかし，クラスタリングには，主に以下の二つの問題点があると考えられる．一つ目の問題点は，属性の類似度だけに基づいてクラスタを形成しているため，人々が一般的に考えている実世界の分類方法と，クラスタリングによって得られる結果が，必ずしも一致するとは限らないということである．二つ目の問題点は，クラスタリング結果から階層構造を抽出する際には，あらかじめ閾値を決めておかなければならない．そのため，ある分野では細かく分けられているが，違う分野ではあまり細かく分けられていないといったような場合には，全分野を同じ閾値で分割する方法では，クラスタの粒度を自動的に変えることができないためうまくいかないと考えられる．

もう一つの従来手法である分類問題は，人手で分類を設定し，少数の訓練データから分類規則を学習することで，残りのデータを分類する（図 9 右）．分類問題の問題点は，事前に人手で分類や訓練データを与えないといけないため，世の中にどのような分類があるのかということを，データを与える人があらかじめ決めなければならないということであり，この手法は本研究の目的にそぐわない．

一方，本研究の手法の場合は，名前と属性の情報を利用して分類であるかどうかを決定しているため，二つ目の分類問題とは違い，あらかじめ何を分類とするのかを決めておく必要はない．また，どのくらいの細かさでオブジェクトとするかは，付けられた名前の情報をもとに決定しているため，クラスタリングのように粒度をあらかじめ決めておくわけではなく，名前に沿った分類ができるのではないかと期待できる．

逆に本手法を用いることの問題点は，分類の抽出方法が名前の付けられ方に依存しているということである．そのため，名前には出現しないような隠れた分類というのは取り出すことができないということであり，この点は今後の課題である．

7. まとめと今後の課題

本研究では，恣意的に名前付けされるオブジェクトの識別問題という新しい問題を提起した．

このようなオブジェクトについての識別を行う場合，オブジェクトに付けられた名前が恣意的であり，同一のオブジェク

トに対して多種多様な名前が存在するため，名前の情報だけを用いて２つの記述が同一オブジェクトについての記述であるかどうかということは判断することができない．また，オブジェクトの粒度も容易に決定することができない．そこで本手法では，多くのユーザの名づけパターンから、分類の粒度を決定する手法を提案した．

実験は料理レシピについて行い，オブジェクトを抽出する実験と，各レシピに書かれているオブジェクトを同定する実験の２つを行った．前者は，抽出漏れは多少あるものの，抽出されたオブジェクトについては 90 %程度の精度が得られた．一方，後者の実験では，一番精度の低かったヤキソバの分野では，20 %程度となってしまった．その原因は，同定のための規則の作り方が単純すぎるためであり，今後の課題である．

また，今回は料理の例だけで説明したが，手法自体は料理に限定されたものではなく，名前と属性の組が与えられれば識別を行うことができる．今後は，事件などの例に応用していきたいと考えている．

謝辞

本研究の一部は，科学研究費補助金（課題番号 18049041, 18049073, 19700091），文部科学省「知的資産のための技術基盤」プロジェクト，京都大学 GCOE プログラム「知識循環社会のための情報学教育研究拠点」，およびマイクロソフト産学連携研究機構 CORE 連携研究プロジェクト）によるものです．ここに記して謝意を表します．

文　　献

[1] Yahoo! JAPAN,http://www.yahoo.co.jp/
[2] Google,http://www.google.co.jp/
[3] Zaiqing Nie, Ji-Rong Wen, Wei-Ying Ma“Object-level Vertical Search” ,CIDR 2007,pp.235-246
[4] Muyuan Wang,Zhiwei Li, Lie Lu, Wei-Ying Ma, Naiyao Zhang,“Web Object Indexing Using Domain Knowledge”, ACM 2005,pp.294-303
[5] Satoshi Oyama, Katsumi Tanaka, “Distance Metric Learning from Cannot-be-linked Example Pairs, with Application to Name Disambiguation ”, Sugato Basu, Ian Davidson and Kiri Wagstaff Eds., In Constrained Clustering: Advances in Algorithms, Theory, and Applications, Chapter 15, pp. 357-374, Chapman & Hall/CRC Press, 2008.
[6] 森純一郎, 松尾豊, 石塚満,“ Web からの人物に関するキーワード抽出 ”, 人工知能学会論文誌, Vol.20, No.5, pp.337-345, 2005.
[7] Rui Kimura, Satoshi Oyama, Hiroyuki Toda, Katsumi Tanaka, “Creating Personal Histories from the Web using Namesake Disambiguation and Event Extraction ”, Proceedings of the Seventh International Conference on Web Engineering (ICWE 2007), Lecture Notes in Computer Science, Vol.4607, pp.400-414, July 2007.
[8] 上田洋, 村上晴美, “ Web 上の同姓同名人物を分離して人物属性情報を表示するシステム ”, 2007 年度人工知能学会全国大会 (第 21 回) 論文集,2007.
[9] 外間智子, 北川博之.“ Web データを用いた人物の呼称抽出 ”. DBSJ Letters, 2006.
[10] McKeown, K., R. Barzilay, D. Evans, V. Hatzivassiloglou,J. L. Klavans, A. Nenkova, C. Sable, B.Schiffman, S. Sigelman,“Tracking and Summarizing News on a Daily Basis with Columbia’s Newsblaster”,In Proceedings of Human Language Technology Conference 2002 (HLT 2002),pp.280-285, San Diego, CA, 2002.
[11] V.Hatzivassiloglou,L.Gravano,and A.Magnati,“An investigation of linguistic features and clustering algorithms for topical document clustering”,In Proceedings of the 23rd Annual ACM SIGIR Conference on Research and Development in Information Retrieval(SIFIR-00),pp.224-231,Athens,Greece,July 2000.
[12] クックパッド,http://cookpad.com/
[13] MeCab,http://mecab.sourceforge.net/