社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

エンティティ検索支援のための知識ベースを用いたWebページ分類

桐谷　雄介†　　馬　　強†　　吉川　正俊†

† 京都大学大学院情報学研究科
〒 606–8501 京都府京都市左京区吉田本町
E-mail: †y.kiritani@db.soc.i.kyoto-u.ac.jp, ††{qiang,yoshikawa}@i.kyoto-u.ac.jp

あらまし　本稿では，人物などのエンティティ検索のため，YAGO や Wikipedia などの知識ベースを用いた Web ページの分類手法を提案する．我々は，まず，Web ページにおけるエンティティの出現頻度や Wikipedia の記事との類似度などを用いて，Web ページをエンティティにマッピングする．そして，YAGO におけるエンティティのクラス情報を利用して，Web ページ，エンティティ，クラスそしてそれらの関係からなるグラフを作成する．このグラフから，Web ページのクラスを推定して分類を行う．また，本稿では，この手法に基づく，検索結果を分類・提示してユーザのエンティティ検索を支援する手法などの応用についても論じる．

キーワード　知識ベース，エンティティ検索，Web ページ分類

Classifying Web Pages by Using Knowledge Base for Entity Retrieval

Yusuke KIRITANI†, Qiang MA†, and Masatoshi YOSHIKAWA†

† Department of Social Informatics, Graduate School of Informatices, Kyoto University
Yoshidahonmachi, Sakyo-ku, Kyoto 606–8501, Japan
E-mail: †y.kiritani@db.soc.i.kyoto-u.ac.jp, ††{qiang,yoshikawa}@i.kyoto-u.ac.jp

1. 序　　論

多くの Web ページが人物の説明であったり，組織の紹介であったりする．また，観光名所の一覧や製品の比較などのページも見られる．このような人物、組織や場所に対する検索が多く行われている．これは，人物に関する検索が 5〜10%であるということからも伺える. [1]．つまり，あらゆる Web ページは何らかのエンティティに関する情報が記述されていて，そのエンティティに対して検索が行われると捉えることもできる．

現在の検索エンジンを利用してエンティティ検索をすると，検索質問との適合度合やリンク構造による重要度に基づいて順位付けされた Web ページのリストを検索結果として出力する．ユーザは各ページの URL，タイトル，スニペットなどから望んだエンティティについて述べられているページかどうかを判断する必要があるが，検索結果数が多いと適切なページに辿りつくのが困難である．そのため，検索質問を改善して検索結果を絞ることになるが，正しい検索質問を作れるとは限らず，本来適切なページまでも取り除くことになりうる．このため，検索結果を無暗に絞ることなくユーザの望んだエンティティを探す方法として，検索結果ページの分類が考えられる．

Web ページの分類には，あらかじめ人手で Web ページを階層的に分類した索引集である Web ディレクトリがあり，Yahoo! Directory [2] や Open Directory Project [3] などが実用されている．これは，正確な分類である一方で，コストがかかる上，全ての Web ページを網羅できない．また，エンティティに基づく分類ではないため，人物や組織の検索には必ずしも有効に機能するとは限らない．

近年，インターネット上のフリー百科事典 Wikipedia [4] が注目されている．Wikipedia には，人物などエンティティに関する情報やエンティティのカテゴリ情報が多数収録されている．これらの情報と WordNet を利用して，YAGO [5] や DBPedia [7] などの知識ベースが構築されつつある．

そこで，本研究では，人物などのエンティティ検索を支援するため，YAGO などの知識ベースを用いた Web ページの自動分類手法を提案する．まず Web ページが説明しているエンティティを，エンティティの出現頻度や Wikipedia のページとの類似度を用いて調べる．次に，その Web ページに対応しているエンティティやクラス情報を YAGO などの知識ベースから取り出して，Web ページ，エンティティ，クラスとそれらの関係から構成されるグラフ（PEC グラフ）を作成して，そのグラフを分析してページの分類を行う．具体的に，

1. Web ページがエンティティにどの程度対応しているかを表す対応度を計算し，Web ページとエンティティをマッピングする．我々は以下のような４つの対応度計算手法を提案している．
   (1) Web ページの本文とエンティティに関する Wikipedia ページとの類似度
   (2) Web ページのタイトル・スニペットとエンティティに関する Wikipedia ページとの類似度
   (3) Web ページの本文におけるエンティティの出現頻度
   (4) Web ページのタイトル・スニペットにおけるエンティティの出現頻度
2. Web ページに対応するエンティティのクラス情報をもとに，Web ページとエンティティとクラスからなるグラフ（PEC グラフ）を作成し，それを解析して，Web ページがクラスにどのくらい対応しているかを表す対応度を計算し，Web ページをクラスに分類する．

さらに，この手法を利用した検索支援システムを提案する．ユーザから検索語を受け取り，Google や Yahoo!などの検索エンジンからその検索結果集合と，YAGO からエンティティ集合を求める．上記手法を用いて，Web ページをクラスに分類してユーザに提示する．

以下，本論文の構成を示す．２章では，関連研究を紹介する．３章では，Web ページとエンティティ間のマッピング手法を，４章では，Web ページの分類方法を説明する．５章では，予備実験について述べる．６章は，本論文のまとめである．

2. 関 連 研 究

Web ページの自動分類手法は多く研究されていて，一般の文書分類手法に加えて，Web 文書独特の HTML 構造やリンクのアンカーテキストを利用している手法 [8] などが多い．また，Chakrabarti らは [9] では，ページのリンク構造を利用する手法を提案している．リンク先のページの情報を考慮することによって，分類精度を上げている．他にも，オントロジを利用した分類手法も提案されている [10] [11]．

それに対して，我々は Web ページにおけるエンティティに注目した分類手法を提案している．本手法では，エンティティの出現頻度やエンティティを説明する Wikipedia 記事との類似度を利用することで，エンティティの検索に重点をおいた分類が出来ると考えられる．

知識ベース YAGO を利用したセマンティックサーチエンジンとして，NAGA [12] があり，これはグラフベースのクエリによりエンティティの検索とそのランキングを行っている．

文献 [13] では Wikipedia から概念のベクトル化手法を提案していて，ネットワーク構造になっている有向グラフ上での２点間のベクトル値を経路の数と長さをもとに計算している．本研究では，２つの２部グラフと木構造のグラフの３つを統合した有向グラフ上での２点間のスコアを計算する．

3. Web ページとエンティティのマッピング

Web ページ集合とエンティティ集合が与えられたとき，Web ページがどのエンティティについて書かれているかを調べ，Web ページをエンティティにマッピングする．

3.1 PE 対応度

Web ページ p がエンティティ e にどのくらい対応しているかを表す PE 対応度 $PE(p,e)$ を定義する．PE 対応度は，Web ページ p におけるエンティティ e の出現頻度や，e に関する Wikipedia の記事との類似度などで計算される．Web ページは，各エンティティとの PE 対応度をもとにマッピングするエンティティを決定する．

Web ページ集合 $P = \{p_1, p_2, ..., p_n\}$ とエンティティ集合 $E = \{e_1, e_2, ..., e_m\}$ があるとき，ページ p とエンティティ e の PE 対応度 $PE(p,e)$ は以下の４種類の計算方法で計算する．

以下 $sim(a,b)$ は a と b それぞれの TF-IDF ベクトルのコサイン類似度，$wikip(e)$ はエンティティ e に関する Wikipedia の記事ページ，$text(p)$ はページ p の本文テキスト，$summary(p)$ はページ p のタイトル・スニペット，$ef(p,e)$ は p での e の出現頻度，$idf(e,P)$ は文書集合 P に対する e の逆文書頻度を表す．

S-TW 手法：$PE(p,e) = sim(text(p), wikip(e))$
これは p の本文 $text(p)$ と e に対応する Wikipedia のページ $wikip(e)$ の TF-IDF ベクトルのコサイン類似度を用いた対応度計算である．Web ページ p と e に対応する Wikipedia のページの記述内容が似ているほど値が高くなる．

S-SW 手法：$PE(p,e) = sim(summary(p), wikip(e))$
これは p のタイトル・スニペット $summary(p)$ と e に対応する Wikipedia のページ $wikip(e)$ の TF-IDF ベクトルのコサイン類似度を用いた対応度計算である．S-TW 手法において，Web ページの本文ではなく，タイトル・スニペットを利用するようにしたものである．

F-TE 手法：$PE(p,e) = ef(text(p), e) \times idf(e,P)$
これは p の本文 $text(p)$ の e の出現頻度 ef(text(p), e) に Web ページ集合 P に対する e の逆文書頻度を掛け合わせた tf-idf のような対応度計算である．Web ページ p に出現するが，他の Web ページにはあまり出現しないほど値が高くなる．

F-SE 手法：$PE(p,e) = ef(summary(p), e) \times idf(e,P)$
これは p のタイトル・スニペット $summary(p)$ の e の出現頻度 ef$summary(p)$ に Web ページ集合 P に対する e の逆文書頻度を掛け合わせた tf-idf のような対応度計算である．F-TE 手法に計算方法において，Web ページの本文ではなく，タイトル・スニペットを利用するようにしたものである．

類似度と出現頻度のどちらを利用するかで，マッピング結果は大きく異なると考えられる．類似度を用いる場合は，類似文書では記述内容が類似するだろうと推定からマッピングを行っており，エンティティに関する Wikipedia の記事の記述量がマッピングの精度に大きく関わっている．出現頻度を用いる場合は，Web ページにおけるエンティティの出現量がマッピングの精度に関わっていて，ページに全く無関係のエンティティにはマッピングされにくいと考えられる．また，本文を利用す

る場合に比べて，タイトル・スニペットを利用すると，実際のページを取得する手間が省け，少ないが重要なテキスト部分だけに対して類似度・出現頻度の計算がされるので，処理時間が大幅に少なくなる．しかし，その一方で精度は落ちると考えられる．5.1 節では，この 4 種類の手法の比較実験について述べている．

3.2 PE 対応度に基づく Web ページとエンティティのマッピング

3.1 節で求めた PE 対応度と 2 つの定数 $\alpha > 1, K \geqq 1$ をもとにマッピングするエンティティを以下のようにして決定する．

1. Web ページ p に対する E の各エンティティの PE 対応度を降順に $PE_1, PE_2, \cdots, PE_m$ と並べる．
2. k 番目と $k+1$ 番目の対応度が

   $$PE_k / PE_{k+1} > \alpha$$

   となるような最小の k をとる．
3. $k > K$ であれば，$k = K$ とする．
4. Web ページを上位 k 番目までの PE 対応度のエンティティにマッピングする．

閾値 α と最小値 K の取り方によって，Web ページがマッピングするエンティティの数を操作する．α によって，対応度が急変する部分の直前のエンティティまで取るようにして，K によって，エンティティの数が多くなりすぎないように制限する．

例えば，ページ p がエンティティ集合 $E = \{e_1, e_2, e_3\}$ のエンティティそれぞれに対して PE 対応度を求めて，

$PE(p, e_1) = 0.44$，$PE(p, e_2) = 0.32$，$PE(p, e_3) = 0.94$，

と計算されたとする．これを降順に並べて，

$PE_1 = PE(p, e_3) = 0.94$，$PE_2 = PE(p, e_1) = 0.44$，

$PE_3 = PE(p, e_2) = 0.12$

とおくと，PE_k / PE_{k+1} は以下のようになる．

$PE_1 / PE_2 = 2.14$，$PE_2 / PE_3 = 3.67$

2 つの定数がそれぞれ $\alpha = 2.0$，$K = 2$ とすると，$k = 1$ となる．また，$\alpha = 3.0$，$K = 2$ だとすると $k = 2$ となる．

4. Web ページの分類

4.1 PEC グラフ

G_{PE}

Web ページ集合 P とエンティティ集合 E と，Web ページとエンティティ間の対応関係 $R_{PE} \subseteqq P \times E$ の集合が与えられたとき，P と E 間の二部グラフ $G_{PE} = (P, E, R_{PE})$ が作成される．図 1 はこのグラフの例である．

G_{EC}

エンティティ集合 E と，知識ベースからエンティティがクラスに属するという情報を取得することで，エンティティ集合 E とクラス集合 C_0 と，エンティティとクラス間の所属関係 $R_{EC} \subseteqq E \times C_0$ から，E と C_0 間の二部グラフ $G_{EC} = (E, C_0, R_{EC})$ が作成される．図 2 はこのグラフの例である．

G_C

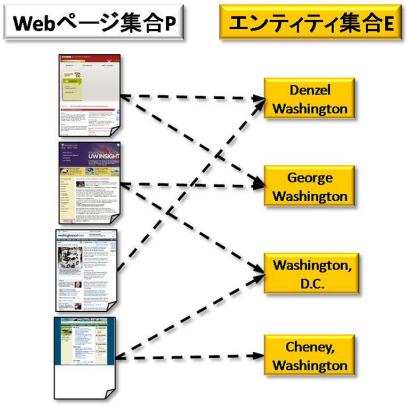

図 1　P と E の二部グラフの例

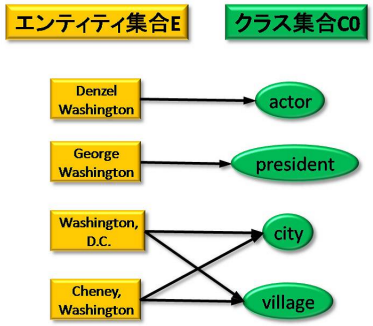

図 2　E と C の二部グラフの例

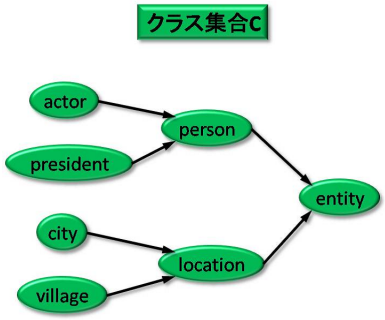

図 3　C の木の例

クラス集合 C_0 と知識ベースからあるクラスが別のクラスの子クラスであるという情報を再帰的に取得することで，クラス集合 $C(\supseteqq C_0)$ と 2 つのクラス間の親子関係 $R_{CC} \subseteqq C \times C$ から，C の木構造となるグラフ $G_C = (C, R_{CC})$ が作成される．図 3 はこのグラフの例である．

これらの 3 つのグラフ G_{PE}, G_{EC}, G_C を統合して作成されたグラフ $G_{PEC} = (P, E, C, R_{PE}, R_{EC}, R_{CC})$ を PEC グラフと呼ぶ．図 4 は PEC グラフの例である．

すなわちこのグラフは，P の要素である Web ページ，E の要素であるエンティティ，C の要素であるクラスをノードとして表現し，2 つのノード間の関係をエッジとして表現した有向グラフとして作成される．エッジを表す関係には，

- R_{PE}
  Web ページがエンティティにマッピングしている関係（向きは Web ページからエンティティへの方向）

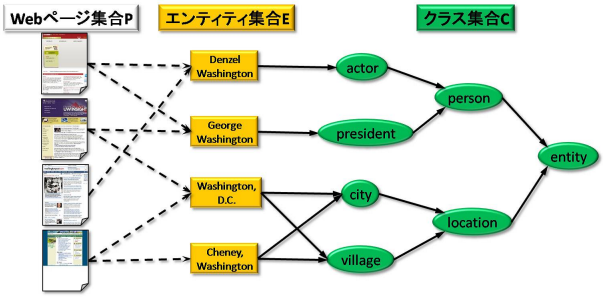

図 4　PEC グラフの例

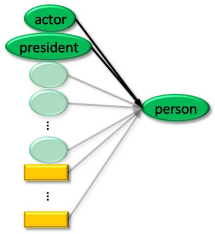

図 5　知識ベース上の潜在的な情報

- R_{EC}
  エンティティがクラスに属している関係（向きはエンティティからクラスへの方向）
- R_{CC}
  クラス A がクラス B の下位クラスであるというクラスの親子関係（向きはクラス A からクラス B への方向）

の 3 つの関係を用いる．

ただし図 4 のように PEC グラフでは描かれないが，図 5 のクラス $person$ のように各クラスは知識ベース上には，他にも多くの下位クラスやエンティティを潜在的に持っている．

4.2　Web ページとクラス間の対応度に基づく分類

作成した PEC グラフを解析することで，Web ページ p がクラス c にどの程度対応しているかを表す PC 対応度 $PC(p,c)$ 計算し，Web ページをクラスに分類する．

PC 対応度は以下の三つの尺度から計算される．

- **クラス c の下位クラス（エンティティ）とページ p の PC 対応度（PE 対応度）の和**：
  この和が大きいほど，p が c への対応度が高いと計算される．
- **クラス c の下位クラス（エンティティ）における p の関連クラス（エンティティ）の割合 r_c**：
  この割合は，c が p の関連クラス（エンティティ）をカバーする被覆率でもある．つまり，p の性質を最も多く表すクラス c が，p との PC 対応度が高い．
- **対象 Web ページ集合におけるクラス c が関連していないページの割合 r_p**：
  この割合が大きいほど，c を用いて p をその他のページと区別しやすい．つまり，c を用いて p を容易に識別できる．

具体的に，Web ページ $p \in P$ とクラス $c \in C$ 間の PC 対応度 $PC(p,c)$ は次のように計算する．ただし，以下 $subNode(c)$ は PEC グラフ上でノード c へ 1 だけのエッジを経て到達することができるノード（子ノード）の集合，$subNode*(c)$ は PEC グラフ上でノード c へ 1 つ以上のエッジを経て到達することができるノード（子孫ノード）の集合，$superNode*(p)$ は PEC グラフ上でノード p から 1 つ以上のエッジを経て到達することができるノード（先祖ノード）の集合，$|P|$ はページ集合 P の大きさとする．

$$PC(p,c) = \left(\sum_{x \in subNode(c)} PX(p,x) \right) \times r_c \times r_p$$

ここで，

$$PX(p,x) = \begin{cases} PE(p,x), & x \text{ がエンティティのとき} \\ PC(p,x), & x \text{ がクラスのとき} \end{cases}$$

であり，これは下位の対応度の和となっている．r_c は

$$r_c = \frac{|subNode(c) \cap superNode*(p)|}{\log |subNode(c)|}$$

で求められ，分母に c の下位クラス集合の大きさの対数，分子に c の下位クラス集合と p の関連クラス（エンティティ）の集合との共通集合の大きさの分数となっている．r_p は

$$r_p = \frac{|P - subNode*(c)|}{\log |P|}$$

で求められ，分母に集合 P の大きさの対数，分子に集合 P と c の子孫クラス集合の差集合の大きさの分数となっている．

5.　予 備 実 験

提案手法の精度を調べるために 2 つの予備実験を行った．4 種類の PE 対応度の計算方法で，Web ページのエンティティへのマッピングと，クラスへの分類の予備実験をそれぞれ行った．

5.1　エンティティへのマッピングの予備実験

Web ページのマッピングの正解エンティティ集合は各 Web ページを人手で見て判断し作成した．各ページごとのマッピング精度は「k=正解集合の大きさ」としたとき，

$$\text{マッピング精度} = \frac{\text{マッピング結果 } topk \text{ に含まれる正解数}}{k}$$

で計算される．例えば，あるページがエンティティA,B,C,D,E に対して PE 対応度を計算しマッピングを行った結果，A,D,B,C の順でマッピングされたとする．そのページの正解集合のエンティティが A,C,D であったとすると，このマッピング精度は 2/3 となる．

検索エンジンとして Yahoo!を用いて，検索語は”washington”で検索結果上位 5 0 件を Web ページ集合，”washington”という語を含むエンティティの集合を YAGO から取り出す．各ページに対して 4 つの PE 対応度の計算方法を適用し，3.2 節における 2 つの定数を $\alpha = 0.5$，$K = 5$ として，Web ページとエンティティのマッピングを行った．そして，4 種類の計算手

	S-TW	S-SW	F-TE	F-SE
マッピング精度	0.433	0.400	0.451	0.341
分類精度	0.80	0.55	0.74	0.48

表 1　予備実験結果

法それぞれの検索結果５０件のマッピング精度の平均を計算した結果が表 1 のマッピング精度である．

４種類の手法のうち，S-TW 手法と S-SW 手法は類似度計算のために Wikipedia の記事ページの取得に時間がかかり，S-TW 手法と F-TE 手法は Web ページ本文の利用のために Web ページの取得に時間がかかるので，計算処理時間の長さは S-TW 手法> S-SW 手法　F-TE 手法> F-SE 手法となっている．この計算処理時間を考慮すると，F-TE 手法が最もよいと考えられる．

S-TW 手法と S-SW 手法では，“Washington Nationals”や“Washington Wizards”などのエンティティが正しくマッピングされた．これは，特殊な単語（人名など）がともに出現していれば，それにより類似度が大きく反映されてマッピングされたと考えられる．一方で，例えば本来エンティティ“Washington Huskies”にマッピングされるべきページが，“Washington County, Kansas”といった全く関係ないものにマッピングされる場合が多々ある．また，“Washington State”にマッピングされるべきページが，地名という点では類似している“Washington, D.C.”にマッピングされる場合も多々見受けられる．これら要因の一つとして，Wikipedia 記事の記述の不足もしくは記述の過剰が影響していると考えられる．これらの特徴は，S-TW 手法の方が S-SW 手法に比べて顕著にみられた．

F-TE 手法と F-SE 手法では，これは本文やタイトル・スニペットからエンティティを抽出することで頻度を計算しているので，全く関係のないエンティティへのマッピングはほとんど起こらなかった．しかし，エンティティが一つもみつからない場合も存在した．また，F-TE 手法の方が F-SE 手法より安定してエンティティを抽出することができるのでよい結果が出ているが，一部では，F-SE 手法の方が優れている場面もあった．これは，タイトルやスニペット部に出現するエンティティが特に重要なエンティティであると捉える事が出来る．ただしこれらの手法は，エンティティ抽出の手法が改良すればもう少し精度が上がると考えられる．

5.2　クラスへのマッピングの予備実験

正解集合は各 Web ページの Yahoo! Directory におけるディレクトリ情報を参考に作成した．各ページは PC 対応度が最も高いクラス１つに分類されるとして，各ページのクラスの分類精度は

$$\text{分類精度} = \begin{cases} 1, & \text{分類が正解} \\ 0, & \text{分類が不正解} \end{cases}$$

というように，分類が正解か不正解かの二値で計算する．

5.2 節のエンティティのマッピング実験と同じ条件で，クラスの分類実験を行った．Web ページとエンティティのマッピングの際の計算方法により４種類を計算した．そして，４種類の計算手法それぞれの検索結果５０件の分類精度の平均を計算した結果が表 1 の分類精度である．

S-TW 手法と F-TE 手法は S-SW 手法と F-SE 手法に比べて，よい精度であった．つまり，Web ページのタイトル・スニペットだけでなく本文を利用するほどよりよい結果が得られたと言える．

また，いずれの手法でもエンティティへのマッピング精度よりも，クラスへの分類精度の方がよい結果であった．これは，例えばエンティティへのマッピングの際に，“Washington State”にマッピングされるべきページが地名という点では類似している“Washington, D.C.”にマッピングされたが，この二つのエンティティは同じクラスに属しているので，クラスへのマッピングは正解となったからではと考えられる．

しかし，正解集合が良くなかったということも考えられる．よりよい正解集合の作成方法も検討する必要がある．

6. 結　　論

本論文では，知識ベース YAGO を利用して，Web ページを分類した．その過程で，Web ページとエンティティ間の関連度および Web ページとクラス間の関連度を計算した．そして，その対応度の計算がどの程度有用であるかを確かめるための予備実験を行った．予備実験により，F-TE 手法が最も有効ではないかと考えられる．また，各手法により有効に適用できる部分があることが分かった．

応用としての検索支援で，分類した後にユーザへ見せるインターフェース部分の開発を進めていきたい．

謝　　辞

本研究の一部は，科研費 (20700084 と 20300036) 助成を受けたものである．

文　　献

[1] R. Guha, A. Garg. Disambiguating People in Search. Proc. of WWW2004, 2004.
[2] Yahoo! Directory: http://dir.yahoo.com/
[3] Open Directory Project: http://www.dmoz.org/
[4] Wikipedia: http://en.wikipedia.org/
[5] Fabian M. Suchanek, Gjergji Kasneci, Gerhard Weikum. Yago: A Core of Semantic Knowledge, Proc. of WWW2007, pp. 697-706, 2007.
[6] WordNet: http://wordnet.princeton.edu/
[7] S. Auer, C. Bizer, J. Lehmann, G. Kobilarov, R. Cyganiak, Z. Ives. DBpedia: A Nucleus for a Web of Open Data. Proc. of ISWC 2007 + ASWC 2007, LNCS4825, pp. 722-735, 2007.
[8] Aixin Sun, Ee-Peng Lim, Wee-Keong Ng. Web Classification Using Support Vector Machine. WIDM'02, pp. 96-99, 2002.
[9] S. Chakrabarti, B. E. Dom, P. Indyk. Enhanced hypertext categorization using hyperlinks. Proc. of the ACM SIGMOD1998, pp. 307-318, 1998.
[10] Rudy Prabowo, Mike Jackson, Peter Burden, Heinz-Dieter Knoell. Ontology-Based Automatic Classification for the Web Pages: Design, Implementation and Evaluation Proc. of WISE'02, pp. 182-191, 2002.
[11] Mu-Hee Song, Soo-Yeon Lim, Dong-Jin Kang, Sang-Jo Lee. Automatic Classification of Web Pages based on the Con-

cept of Domain Ontology. Proc. of APSEC'05, pp. 645-651, 2005.

[12] Gjergji Kasneci, Fabian M. Suchanek, Georgiana Ifrim, Maya Ramanath, Gerhard Weikum. NAGA: Searching and Ranking Knowledge. Proc. of ICDE2008, pp. 1285-1288, 2008.

[13] Masumi Shirakawa, Kotaro Nakayama. Concept Vector Extraction from Wikipedia Category Network. Proc. of ICUIMC-09, pp. 71-79, 2009.