グラフカーネルに基づく
非分かち書き文からの意味的語彙カテゴリの抽出

萩原 正人，小川 泰弘，外山 勝彦 (名古屋大学)
{hagiwara,yasuhiro,toyama}@kl.i.is.nagoya-u.ac.jp

1 はじめに

人名や地名などの固有表現やそれらの関係は，意味処理において重要な知識源である．これらの知識源の構築・維持には膨大なコストがかかるため，計算機による自動獲得手法が盛んに研究されている．その中でも特に，文脈パターンを手がかりとしたブートストラップを用いて，半教師あり学習により固有表現やその関係を抽出する手法が注目されている．例えば，国名や大統領の名前などの意味的語彙カテゴリ (以下単に意味カテゴリと呼ぶ) を抽出するもの [1, 7] や，is-a 関係や part-of 関係などの意味的関係を抽出するもの [9] が提案されている．

しかし，これらの手法は，英語などの分かち書きされた文や，日本語の検索ログなどの短い文を対象にしており，日本語の長い非分かち書き文に対しては，形態素解析等による分割を経なければ適用できないという問題がある．しかし，固有表現抽出においては，未知語・新語の問題が特に顕著であり，既存の形態素解析や構文解析に頼ることができない．したがって，非分かち書き文から直接，意味カテゴリを抽出する必要性が生じる．

この問題に対し我々は，形態素解析などを用いずに，非分かち書き文から意味カテゴリを直接抽出するブートストラップ法 *Monaka* を提案した [2]．*Monaka* は，隣接する文字 *n* グラムを文脈パターンとして直接用い，インスタンスを信頼度の高い文脈により挟むという制約を加えることにより，正しく分かち書きされた意味カテゴリを抽出することができる．

しかし，助詞の「で」など，他の多くのインスタンスと共起するパターンをひとたび抽出すると，シードと関連性の低いインスタンスを抽出してしまうという (1) **意味ドリフト**の問題があり，Komachi et al. [6] はこの問題が *Espresso* などのブートストラップ法にとって本質的に不可避であることを示した．また，(2) ブートストラップ法における信頼度の形式化はされておらず，さらに，(3) 獲得するインスタンスやパターンの数など，調節すべきパラメータ数が多いという欠点もある．

そこで本稿では，これらの問題に対処できるグラフカーネルを用い，非分かち書き文から意味カテゴリを抽出するアルゴリズム *g-Monaka* を提案する．本手法は，文字 *n* グラムの隣接関係を有向グラフにより表現し，グラフカーネルを適用することにより，意味ドリフト等の問題を回避することができる．比較実験により，*Espresso* や *Monaka* 等の従来手法よりも高い精度で意味カテゴリを抽出できることを示す．

2 ブートストラップ法

2.1 *Espresso* アルゴリズム

Espresso[9] は，ブートストラップを用いた半教師有り学習によって 2 項関係を抽出する手法であり，はじめに少数の正解をシードとして与えた後，パターン抽出とインスタンス抽出のステップを繰り返す．

パターン抽出ステップでは，インスタンスと共起する文脈パターンをコーパスから抽出する．続いて，抽出されたパターン p の信頼度 $r_\pi(p)$ を以下のように求める：

$$r_\pi(p) = \frac{1}{|I|}\sum_{i\in I}\frac{\mathrm{pmi}(i,p)}{\max \mathrm{pmi}} r_\iota(i) \tag{1}$$

ここで，I はインスタンスの集合，$\mathrm{pmi}(i,p)$ は i と p との自己相互情報量であり，

$$\mathrm{pmi}(i,p) = \log \frac{|i,p|}{|i,*||*,p|} \tag{2}$$

によって計算できる．ただし，$|i,p|$ は i と p の共起頻度を表し，*はワイルドカードを表す．

インスタンス抽出ステップでは，信頼度の高いパターンを用いてインスタンスを抽出する．インスタンス i の信頼度 $r_\iota(i)$ は，式 (1) と同様に，

$$r_\iota(i) = \frac{1}{|P|}\sum_{p\in P}\frac{\mathrm{pmi}(i,p)}{\max \mathrm{pmi}} r_\pi(p) \tag{3}$$

により求める．信頼度の高いインスタンスを新たな入力として上記ステップを反復することにより，インスタンスを増やしていく．

2.2 *Monaka* アルゴリズム

非分かち書き文からの意味カテゴリ抽出手法 *Monaka* [2] では，インスタンスに隣接する文字 *n* グラムをパターンとして用いる．例えば，「これを受け、日本政府は次期通常国会に協定の承認案を提出する考えだ。」という文からインスタンス「日本」に対応するパターンを抽出すると，「、#」「け、#」「受け、#」などの左側文脈および「# 政」「#政府」「#政府は」などの右側文脈が得られる．ここで，#はインスタンスのスロットを表す．インスタンス抽出ステップでも同様に，スロット位置に存在する文字 *n* グラムを抽出する．例えば，上述の文にパターン「#政府は」を適用した場合に抽出されるインスタンスは，「本」「日本」「、日本」「け、日本」などとなる．

このようにパターンを定義することにより，分かち書きに頼らない抽出が可能となる．しかし，上述のように正しく分かち書きされていないインスタンスが大量に抽出されてしまう．そこで，「信頼度の高いインスタンスは，信頼度の高い右側文脈と左側文脈に挟まれていなければならない」という**両側隣接制約**を導入した．具体的には，インスタンスの信頼度 r_ι を，右側文脈から計算される右側信頼度 r_R と左側文脈から計算

される左側信頼度 r_L の一般化平均を用いて，

$$r_\iota(i) = \sqrt[m]{\frac{1}{2}(r_L^m(i) + r_R^m(i))} \tag{4}$$

として求める．一般化平均は，算術平均，幾何平均など各種平均の一般化であり，m によって両側隣接制約の強さを調節することができる．m を 0 に近い値にすることにより，r_L と r_R の両者が高いときにのみ r_ι が高くなるという制約を表現することができ，信頼度の高いインスタンスに対して分かち書きの正しさが保証される．実験の結果，形態素解析等を用いていないにも関わらず，信頼度の上位では 100%近い精度で正しく分かち書きされたインスタンスを抽出できた [2].

3 ブートストラップのグラフ解析的解釈

Espresso などのブートストラップ法には，1 節で述べた 3 つの問題がある．そこで，Komachi et al. [6] は，ブートストラップ法をグラフ解析として定式化して，グラフカーネルを適用することにより，これらの問題を解決した．以下にその概要を示す．

3.1 ブートストラップの定式化

ここでは，*Espresso* を行列演算によって定式化する．まず，インスタンスとパターンの共起行列を M とし，シードの信頼度ベクトルを $\boldsymbol{i}_0$ とする．行列 M の (p,i) 要素 $M(p,i)$ は，$M(p,i) = \mathrm{pmi}(i,p)/\max \mathrm{pmi}$ とする．また，シードの信頼度ベクトルは，各シードのインデックスに対応する要素が 1，それ以外が 0 となるようなベクトルである．

このとき，*Espresso* の n 回目の反復のパターン抽出ステップは，$\boldsymbol{p}_n = M\boldsymbol{i}_n$ としてパターンの信頼度ベクトル $\boldsymbol{p}_n$ を求めた後，$\boldsymbol{p}_n \leftarrow \boldsymbol{p}_n/|I|$ として正規化する操作に対応する．同様に，インスタンス抽出ステップは，$\boldsymbol{i}_{n+1} = M^T\boldsymbol{p}_n$ としてインスタンスの信頼度ベクトルを求めた後，$\boldsymbol{i}_{n+1} \leftarrow \boldsymbol{i}_{n+1}/|P|$ として正規化する操作に対応する．以上をまとめると，*Espresso* の n 回目の反復後に得られるインスタンスの信頼度ベクトル $\boldsymbol{i}_n$ は，

$$\boldsymbol{i}_n = A^n\boldsymbol{i}_0, \quad A = \frac{1}{|I||P|}M^TM \tag{5}$$

と書ける．各ステップにおける更新は，I を節点集合，A を接続行列とした無向グラフ G において，インスタンスの信頼度がシードから伝播していく過程と見なすことができ，各種のグラフカーネルが適用できる．

3.2 グラフカーネルの適用

グラフ上における類似度の伝播を扱うモデルとして，ノイマンカーネル [4] K_β が提案されている：

$$K_\beta(A) = A\sum_{n=0}^{\infty}\beta^nA^n = A(I-\beta A)^{-1} \tag{6}$$

これは，グラフ G における無限長までの全ての経路を考え，その重み付き和を求めることに相当し，節点間の高次の相関関係を捉えられる．ただし，次数 n が高くなるにつれ，ジェネリック・インスタンスに対して高い重みが付くという問題点がある．そこで，ノイマンカーネル同様に全経路を考慮するモデルであるが，

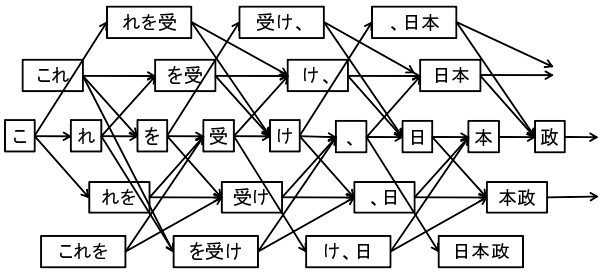

図 1: 例文に対する n グラムの接続グラフの一部

他の多くの節点と繋がっているようなインスタンスに対しては低い重みを与える**正則化ラプラシアンカーネル**を用いる [6]．まず，グラフ G のラプラシアン L を

$$L = D - A, \qquad D(i,i) = \sum_j A(i,j) \tag{7}$$

として求めた後，正則化ラプラシアンカーネルを

$$R_\beta(A) = A\sum_{n=0}^{\infty}\beta^n(-L)^n = (I+\beta L)^{-1} \tag{8}$$

として計算する．D の対角要素が各節点の度数を表しているため，$-L$ は A の節点に対し，その度数に応じて負の自己ループとしてペナルティを与えたものと解釈することができ，ジェネリック・インスタンスの影響を低く抑えることが可能である．

4 提案手法

本節では，2.2 節において紹介した *Monaka* アルゴリズムの問題点を，グラフ解析手法により解決した手法である *g-Monaka* アルゴリズムについて述べる．

4.1 隣接関係のモデル化

g-Monaka ではまず，文中の全ての文字 n グラムの隣接関係を，図 1 に示すような有向グラフによって捉える．コーパス中の全ての文に対してこの隣接関係を求めた後，n グラム u と v との接続頻度 $|u,v|$ を求め，行列 M を以下のように構築する：

$$M(u,v) = \frac{\mathrm{pmi}(u,v)}{\max \mathrm{pmi}}, \quad \mathrm{pmi}(u,v) = \log\frac{|u,v|}{|u,*||*,v|} \tag{9}$$

全ての n グラムの集合 V を節点集合，M を接続行列として表現される有向重み付きグラフ G_M を考える．V には全ての n グラムが含まれているため，*Espresso* のようにインスタンスとパターンを区別する必要がなくなり，G_M は一般に二部グラフにはならない．また G_M は，論文の引用関係や Web ページのリンク関係と同一の構造を持つため，HITS[5] などの引用解析手法が適用できる．例えば，n グラム u と v に対して，右側に接続する n グラムの集合，すなわち右側文脈が似ているほど u と v は意味的に類似していると考えられるが，これは各 n グラムを文書と見なしたときの書誌結合の概念に対応する．同様に，左側文脈による類似度は共引用の概念に対応する．したがって，引用解析手法の一つであるグラフカーネルにより n グラム間の類似関係を自然に表現することができる．

4.2 グラフカーネルの適用

右側文脈と左側文脈に基づく類似度行列は，接続行列 M を用いて，それぞれ

$$A_R = \frac{1}{|V|^2} MM^T, \quad A_L = \frac{1}{|V|^2} M^T M \tag{10}$$

として求められる．ここで，A_R,A_L は引用解析における書誌結合行列と共引用行列にそれぞれ対応する．

Monaka と同様に，n グラム u と v に対して，右側文脈 (書誌結合) と左側文脈 (共引用) の両者が類似していてはじめて u と v は類似していると言えるため，

$$A(i,j) = \sqrt[m]{\frac{1}{2}\left(A_R(i,j)^m + A_L(i,j)^m\right)} \tag{11}$$

として，要素毎の重み付き一般化平均によって A を求める[1]．式 (4) と同様，パラメータ m によって両側隣接制約の強さを調節できる．

接続行列 A によって表現される無向重み付きグラフ G_A は，接続行列が A_R である書誌結合グラフ G_R と接続行列が A_L である共引用グラフ G_L を重ね合わせたものになっている．前節で述べた正則化ラプラシアンカーネルをグラフ G_A に対して適用することにより，ブートストラップ法の問題を回避しながら，非分かち書き文からの意味カテゴリ抽出が可能になる．

4.3 近似アルゴリズム

本手法の問題点として，文中の全ての n グラムを対象としているため，グラフカーネルの計算量が膨大になるという問題点がある．そこで，ブートストラップを 1 回反復してもシードから類似度が伝播しないインスタンスやパターンは無視しても差し支えないと考え，それらを除いた部分グラフ上においてグラフカーネルを近似する．

具体的には，シード $\boldsymbol{v}_0$ に対して，まず *Monaka* を 1 回反復させ，インスタンス $\boldsymbol{v}_1$ を得る．次に，$\boldsymbol{v}_1$ に対して右側文脈 $\boldsymbol{v}_1^R = M^T\boldsymbol{v}_1$ と左側文脈 $\boldsymbol{v}_1^L = M\boldsymbol{v}_1$ を求める．$\boldsymbol{v}_1, \boldsymbol{v}_1^R, \boldsymbol{v}_1^L$ 中の信頼度上位の n グラムのみを取り出し，その集合を $V_S(\subset V)$ とする．実際には，インスタンスおよび右側/左側文脈の数が偏るのを防ぐため，まず $\boldsymbol{v}_1$ から N_{inst} 個の n グラムを選び，$|V_S| = N_{all}$ が満たされるまで，$\boldsymbol{v}_1^R, \boldsymbol{v}_1^L$ から信頼度の高い順に交互に n グラムを V_S に追加した．実験では，$N_{inst} = 2{,}000$, $N_{all} = 6{,}000$ としたが，これらの値は近似精度と計算量のトレードオフを考慮して決定する．最後に，V_S が節点であり，辺の重みが式 (9) で与えられるような部分グラフ G_S に対して正則化ラプラシアンカーネルを計算した．

5 性能比較実験

本節では，意味カテゴリ抽出タスクを用いて，提案手法 *g-Monaka* と従来手法の性能を比較する．

5.1 条件条件

コーパス コーパスには，2007 年版毎日新聞コーパス中，1 面，2 面，3 面，国際，経済カテゴリの記事本文を用いた．句点をまたいだ n グラムは使用しない．罫線のみの行や，記事末尾・先頭の記者名，行頭の空白と見出し用各種記号の連続は取り除いた．この前処理後のコーパスは，約 31 万文，1,341 万文字からなる．

文字 n グラム長は $1 \leq n \leq 8$ とし，出現頻度が 30,000 回以上および 20 回未満のものはストップワードまたはノイズと判断して使用しなかった．最終的に使用した n グラム数は 306,919 個であった．

正解セット 意味カテゴリの正解セットとして，世界の国・地域名 (CNT)，日本の現行法令名 (LAW)，国内外の自動車メーカー名 (CAR) の 3 つを用いた．CNT は Wikipedia の「国の一覧」から各国の「通称名・別名」を，LAW は法令データ提供システム[2]の「憲法・法律」の正式名称および「略称法令名」のリストから，CAR は Wikipedia の「自動車製造者の一覧」から収集し，V に含まれているものを正解セットとした．各セットのサイズはそれぞれ 140, 24, 21 である．シードとして，CNT については 5 個 (ただし「米」「仏」などの漢字一文字の略称を除く)，LAW, CAR については 3 個，正解セットから無作為にインスタンスを選び，この操作を各セットに対して 3 回繰り返した平均を各手法の性能とした．

比較手法 (1) 分布類似度 [8]，(2) 3.1 節で述べた *Espresso* にインスタンスとパターンの取捨選択を加えた *Filtered Espresso*, (3) 2.2 節で述べた *Monaka*, そして (4) *g-Monaka*(提案手法) の 4 手法を比較した．なお，(1) 分布類似度は，右側・左側文脈による類似度を別々に計算し，両側隣接制約と同様に両者の一般化平均により統合した．実験では $m = 1.0$ と $m = 0.1$ の場合を比較した．(2) *Filtered Espresso* と (3) *Monaka* では，n 回目の反復において獲得するインスタンス数を $5n+5$, パターン数を $5n+195$ とした．(4) *g-Monaka* の正則化ラプラシアンカーネルの拡散パラメータは $\beta = 5.0 \times 10^{-4}$ に設定したが，このカーネルは β に対して性能が安定していることが知られており [6]，予備実験においても同様の傾向を確認した．

5.2 結果

精度-再現率の変化 まず，*Monaka* において，獲得されたインスタンスの質が反復が進むにつれてどのように変化するかを，正解セット CNT に対して,「アルゼンチン」「エクアドル」「キルギス」「韓国」「バングラデシュ」の 5 個をシードとして調べた．その結果が図 2 であり，各反復におけるインスタンスの精度・再現率・F 値を示している．最初は再現率・F 値共に増加していくが，反復 25 回付近から再現率が低下している．これは，反復が進むにつれ，次の実験で示すようなジェネリック・インスタンスが獲得されているためであり，ここから，*Monaka* においても意味ドリフトが発生していると推測される．

各アルゴリズムの比較 次に，上述の各アルゴリズムによって獲得されたインスタンスを比較した．前実験と同じ正解セットとシードを用い，分布類似度 ($m = 1.0$), *Monaka*(30 回反復), *g-Monaka* によって獲得されたインスタンスの信頼度上位 50 個を表 1 に示す．分布類似度 ($m = 0.1$) を用いた場合については，抽出結果が *Monaka* と類似しているため比較から除いた．分布類

[1] Ito et al. [3] は，書誌結合と共引用を統合した類似度指標として $\alpha(M^TM) + \beta(AA^T) + \gamma(A + A^T)$ を提案しており，式 (11) は，その変種として両側隣接制約を考慮したものであると言える．

[2] http://law.e-gov.go.jp/

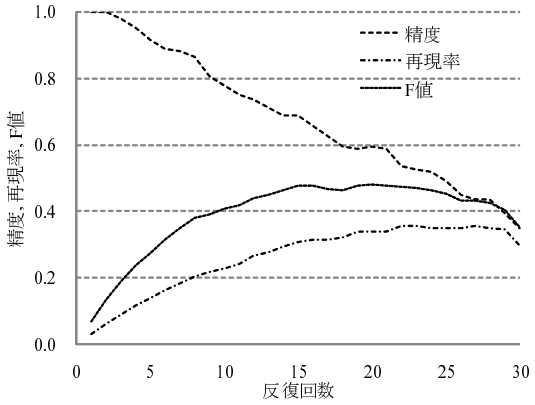

図 2: *Monaka* における精度-再現率の変化

似度 ($m = 1.0$) については，両側隣接制約を用いていないため,「イラ」「ロシ」など分割の不適切なインスタンスが獲得されているが，*Monaka* ではそれが少なくなっている．しかし,「地方」「地域」「各国」のようなジェネリック・インスタンスが獲得されており，これらが性能の低下を招いていると考えられる．一方，*g-Monaka* ではそのようなインスタンスは信頼度 50 位近くにも見られず，グラフカーネルによって意味ドリフトが抑えられていることが分かる．

最後に，各手法の F 値の最大値を表 2 に示した．*Filtered Espresso* は，両側隣接制約を用いていないため，反復に従い精度が急激に低下し，いずれの手法よりも性能が低くなっている．一方，提案手法の *g-Monaka* は，いずれの正解セットにおいても従来手法の性能を 20%以上上回っている．なお，正解セットによっては，*Monaka* の性能が単純な分布類似度の性能よりも低い場合があるが，これも意味ドリフトによる悪影響によるものであると考えられる．

6 おわりに

本稿では，非分かち書き文から意味的語彙カテゴリを抽出するアルゴリズム *g-Monaka* を提案した．本手法では，文字 *n* グラムの隣接関係グラフ上における引用解析としてブートストラップを形式化した．このことにより，本手法自体および従来手法との比較に関して見通しが良くなった．また本手法では，グラフカーネルを用いており，形態素や文字種などの情報に頼ること無く，従来手法よりも高い精度で意味カテゴリの抽出が可能であることを示した．

なお，KnowItAll[1] や SEAL[10] など，wrapper と呼ばれる前後の文字列をパターンとして用いる意味カテゴリ抽出法も提案されており，これら従来手法との比較は今後の課題である．また，中国語やタイ語など，分かち書きされない日本語以外の言語においては，高精度な形態素解析システムや固有表現に関する言語資源等も未整備の部分が多く，本アルゴリズムの貢献は大きいと考えられ，引き続き検討する．

表 1: 獲得されたインスタンス

手法	インスタンス
分布類似度 ($m = 1.0$)	韓国, 韓, 日本, 中国, 米国, 米, 、, 韓国, ロシア, 北朝鮮, イラン, バングラデシュ, イラク, 英国, 韓国の, インド, パキスタン, ドイツ, 英, 北, アルゼンチン, 同国, 政府, フランス, トルコ, 、, 日本, イスラエル, イラ, 欧州, 北朝, 韓国政府, ミャンマー, 東, 今, 地, 国内, エクアドル, 大統領, 民主党, 世界, フィリピン, 台湾, 東京, 自, 欧, ロシ, ペルー, アフガン, 日本の, 、, 中国, 外
Monaka	韓国, 日本, 中国, 米国, 米, ロシア, イラン, 北朝鮮, バングラデシュ, イラク, 韓, 英国, パキスタン, インド, ドイツ, アルゼンチン, 英, 政府, 同国, トルコ, フランス, 、, 韓国, イスラエル, 韓国の, 欧州, ミャンマー, 国内, 大統領, 台湾, フィリピン, 地, エクアドル, 民主党, 韓国政府, アフガン, ペルー, 、, 日本, 世界, 外, アフリカ, 北京, 地元, キルギス, タイ, コ, 首相, 各国, 地域, 地方, 今回
g-Monaka	韓国, 中国, 日本, 米国, ロシア, 米, 北朝鮮, イラン, 英国, 韓, イラク, パキスタン, ドイツ, インド, 英, イスラエル, 欧州, アフガン, ミャンマー, フランス, 韓国政府, 台湾, トルコ, 同国, 韓国の, ペルー, フィリピン, 欧米, カナダ, 政府, 、, 韓国, 香港, 朝鮮, 外国, 両国, 民主党, スーダン, 欧, オーストラリア, 地元, 中国、韓国, マカオ, アフリカ, ブッシュ米, 各国, パレスチナ自治, インドネシア, 国内, ベトナム, 北京

表 2: 各アルゴリズムの F 値 (最大値)

手法	CNT	LAW	CAR
分布類似度 ($m = 1.0$)	0.339	0.365	0.414
分布類似度 ($m = 0.1$)	0.408	0.464	0.463
Filtered Espresso	0.296	0.269	0.332
Monaka	0.479	0.391	0.477
g-Monaka	**0.645**	**0.578**	**0.570**

参考文献

[1] Oren Etzioni, et al. Web-scale Information Extraction in KnowItAll: (Preliminary Results). *Proc. of WWW 2004*, pp. 100–110, 2004.

[2] Masato Hagiwara, Yasuhiro Ogawa, and Katsuhiko Toyama. Bootstrapping-based Extraction of Dictionary Terms from Unsegmented Legal Text. *Proc. of the Second International Workshop on Juris-Informatics (JURISIN 2008)*, pp. 63–72, 2008.

[3] Takahiko Ito, Taku Kudo, Campbell Hore, Masashi Shimbo, and Yuji Matsumoto: Computing Citation Relatedness Using Kernels (preliminary report). *IEIC Technical Report*, Vol. 103, No. 305, pp. 25–30, 2003.

[4] Jaz Kandola, John Shawe-Taylor, and Nello Cristianini. Learning semantic similarity. *Neural Information Processing Systems (NIPS 15)*, pp. 657–664, 2002.

[5] Jon M. Kleinberg. Authoritative Sources in a Hyperlinked Environment. *Journal of ACM*, pp. 604–632, 1999.

[6] Mamoru Komachi, Taku Kudo, Masahi Shimbo, and Yuji Matsumoto: Graph-based Analysis of Semantic Drift in *Espresso*-like Bootstrapping Algorithms. *Proc. of the EMNLP 2008*, pp. 1011–1020, 2008.

[7] Mamoru Komachi and Hisami Suzuki: Minimally Supervised Learning of Semantic Knowledge from Query Logs. *Proc. of IJCNLP 2008*, pp. 358–365, 2008.

[8] Dekang Lin. Automatic retrieval and clustering of similar words. *Proc. COLING/ACL 1999*, pp. 786–774.

[9] Patrick Pantel and Marco Pennacchiotti: *Espresso*: Leveraging Generic Patterns for Automatically Harvesting Semantic Relations. *Proc. of ACL 2006*, pp. 113–120, 2006.

[10] Richard C. Wang and William W. Cohen. Language-Independent Set Expansion of Named Entities using the Web. *Proc. of ICDM 2007*, pp. 342–350, 2007.