ブートストラップ法のための能動学習

江原遥 †，佐藤一誠 ‡，中川裕志 ‡
† 東京大学大学院情報理工学系研究科，‡ 東京大学情報基盤センター
{ehara,sato}@r.dl.itc.u-tokyo.ac.jp, n3@dl.itc.u-tokyo.ac.jp

1　はじめに

近年，自然言語処理（NLP）におけるラベル付けコストを削減する方法の1つとして，半教師あり学習の1つであるブートストラップ法[1]が注目を集めている．ブートストラップ法は，クラス（ラベル）に属する少数のインスタンスの集合（シード集合，例：{ シリウス, ヴェガ }）の所与の元，ラベルなしデータ中でシードと強く共起するパターン（素性）を抽出し，抽出された素性と強く共起するインスタンスを再度抽出する，という手順を反復することによって，シード集合と類似したインスタンス（例：アンタレス）を抽出する方法である．このシード集合の選び方が精度に影響することは知られているが，どのような基準でシード集合を選択すれば良いのかについてはあまり知られていない．

本研究では，**逐次シーディングフレームワーク**というシード集合の選択方法を提案する．逐次シーディングフレームワークは，現在のラベルなしインスタンス集合から，ある基準に従って**シード候補値**を計算し，それが最大のシード候補を1つ選び，シード集合に移していくことを逐次的に繰り返す手法であり，半教師あり能動学習の一種と見なせる．逐次シーディングフレームワークにおける基準は，元となるブートストラップ法から理論的に導出できる．本研究の貢献は，以下のとおりである．

1. 逐次シーディングフレームワークの提案
2. シード候補値の高速計算

表1に本研究の位置づけを示す．シード集合の選択問題は，近年，Kozareva ら [2] が問題提起しているが，実際にシード集合を選択してブートストラップ法の性能改善を確認していない．近年の代表的なブートストラップ法の理論化として，Komachi ら [1] による Espresso 型ブートストラップ法の理論化があげられる．Espresso[4] は Pantel らによる代表的なブートストラップ手法であるので，この理論化から議論を始めることは妥当である．

[1]統計分野のブートストラップ法とは関係がない．

手法	シード選択法	シード候補値とモデルの関係	シード候補値の高速計算
Espresso[4]	ランダム	-	NA
LLP [1]	ランダム	-	NA
Kozareva[2]	SVR	-	-
本研究	逐次シーディング Algorithm 1	新規．§3.1 で説明．	新規．§4 で説明．

表 1: 既存手法との比較．SVR: Support Vector Regression.

2　単純ブートストラップアルゴリズム

本節では [1] に従い，ブートストラップ法のモデル化である**単純ブートストラップアルゴリズム**とに対する**ラプラシアンラベル伝搬法** (Laplacian label propagation, *LLP*) の優位性を示す．

$\mathcal{D} \stackrel{\text{def}}{=} \{(y_1, \boldsymbol{x}_1), \ldots, (y_l, \boldsymbol{x}_l), \boldsymbol{x}_{l+1}, \ldots, \boldsymbol{x}_{l+u}\}$ をデータセットとする．各インスタンス $\boldsymbol{x}_i \in \mathbb{R}^m$ は，m 次元の素性ベクトルとして表現され，各ラベル $y_i \in C$ を $\boldsymbol{x}_i$ に対応するラベルとする．全 $n = l + u$ のインスタンスのうち，最初の l 件のデータにはラベル y が与えられており，残りの u 件はラベルなしインスタンスである．$k \in C$ に対して，$y_{ik} \in \{0, 1\}$ であるような二値ベクトル $\boldsymbol{y}_k \stackrel{\text{def}}{=} (y_{1k}, \ldots, y_{nk})^\top$ シードベクトルと定義する．ここで，i 番目のインスタンス $\boldsymbol{x}_i$ がラベル付されており，かつ，$\boldsymbol{x}_i$ のラベルが $k \in C$ であるならば $y_{ik} = 1$ であり，それ以外の場合は，$y_{ik} = 0$ である．以上は多クラス問題の設定であるが，この定義は広く使われているランキングの問題設定をも $|C| = 1$ の特殊な場合として含んでいる．インスタンス-素性行列を $X \stackrel{\text{def}}{=} (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^\top$ と定義する．言語の疎性により，X は通常疎行列である．

単純ブートストラップアルゴリズムは，ブートストラップ法のモデル化であり，以下の2つを $\boldsymbol{f_c}$ が収束するまで反復するアルゴリズムとして定義される．ただし，初期値 $\boldsymbol{f}_0 = \boldsymbol{y}$ である．

1. $\boldsymbol{a}_{c+1} = X^\top \boldsymbol{f}_c$ を計算し，$\boldsymbol{a}_{c+1}$ を正規化．
2. $\boldsymbol{f}_{c+1} = X\boldsymbol{a_{c+1}}$ を計算し，$\boldsymbol{f_{c+1}}$ を正規化．

以上を c 回反復した時，$\boldsymbol{f}_c = \left(\frac{1}{m}\frac{1}{n}XX^\top\right)^c \boldsymbol{y}$ と表せる．特に Espresso の簡易版である Simplified Espresso は，$X_{ij} = \frac{pmi(i,j)}{\max_{i,j} pmi(i,j)}$ と置いた場合に相当する．

単純ブートストラップアルゴリズムに帰着する手法では，反復回数 c が大きい時，$\left(\frac{1}{m}\frac{1}{n}XX^\top\right)^c$ の主固有ベクトルが支配的になるためスコアベクトル $\boldsymbol{f}$ が教師情報であるシードベクトル $\boldsymbol{y}$ に依存しなくなってしまい，半教師あり学習としてのの目的を果たせない．意味ドリフトの要因であるこの問題は，次式の LLP で解決される．

$$\mathbf{f} = (I + \beta L)^{-1}\mathbf{y} \tag{1}$$

数式 (1) において，$L \stackrel{\text{def}}{=} I - D^{-1/2}XX^\top D^{-1/2}$ であり $D_{ii} \stackrel{\text{def}}{=} \sum_j (XX^\top)_{ij}$ である．行列のスペクトル半径を返す関数を ρ とすると，$0 < \beta < \frac{1}{\rho(L)}$ であり，$(I + \beta L)^{-1}\boldsymbol{y} = \sum_{c=1}^{\infty} \beta^c (-L)^c \boldsymbol{y}$ と無限和分解することができる．数式 (1) は無限にラベルを伝搬させた場合のスコアを β^c で重み付けしているため，LLP では主固有ベクトルが支配的になる問題は起こらない．

3 提案：逐次シーディング

Algorithm 1 に，提案する**逐次シーディングフレームワーク**を示す．逐次シーディングの 1 反復では，まず，全ラベルなしインスタンスに対してシード候補としての良さ g_i を計算する．この g_i を**シード候補値**と呼ぶことにする．次に，全インスタンスのうち g_i の値が最高のインスタンス $\hat{i} \stackrel{\text{def}}{=} \arg\max_i g_i$ を 1 つ選んでシードに追加していく．g_i の定義の仕方を基準と呼び，この基準を変える事によって性質の異なるシードの選び方を 1 つの同じ逐次シーディングフレームワークの元で統一的に表現できる．

シード候補値 g_i を計測するためには，インスタンス i がモデルに与える影響の大きさを計測する必要があるが，そのモデルは数式 (1) では隠れてしまっている．

3.1 マージンとしてのスコア

数式 (1) では隠れているモデルパラメータ $\hat{\mathbf{w}}$ は，数式 (1) の双対問題を考えることによって明らかになる [7]．主な結果を説明する．数式 (1) を実行したときのインスタンス i のスコア f_i は，数式 (2) に示すリッジ回帰問題における**マージン** $\|y_i - \langle \hat{\mathbf{w}}, \phi(\mathbf{x}_i)\rangle\|$ を用いて，$f_i \propto \|y_i - \langle \hat{\mathbf{w}}, \phi(\mathbf{x}_i)\rangle\|$ と表せる．マージンは，図 1 に図示したように，データ点と識別平面との距離として解釈する事ができる．モデルパラメータ $\hat{\mathbf{w}}$ はこの識別平面の方向を表す．定数倍の違いを捨象すると $\hat{\mathbf{w}}$ と $\mathbf{f}$ の関係は，簡単に $\hat{\mathbf{w}} = \Phi^\top \mathbf{f}$ と表せる．ここで，$\Phi \stackrel{\text{def}}{=} (\phi(\mathbf{x}_1), \ldots, \phi(\mathbf{x}_n))^\top$

Algorithm 1 逐次シーディングフレームワーク

Require: $\boldsymbol{y}$, X, ラベルなしインスタンス集合 U, クラス集合 C.

$\forall k \in C, \forall i' \in U$ に対して $g_{i',k}$ を初期化.

repeat

  $\hat{i} \stackrel{\text{def}}{=} \arg\max_i g_i$ に従って，$\hat{i}$ を選択する.

  $\hat{i}$ をラベル付けする．$\hat{i}$ のクラスを k' とする.

  $U \leftarrow U \backslash \{\hat{i}\}$

  (以下，§4 に述べる高速化が必須．)

  for all $i' \in U$ **do**

    $g_{i',k'}$ を再計算する.

  end for

until 十分な数のシードが集まるまで.

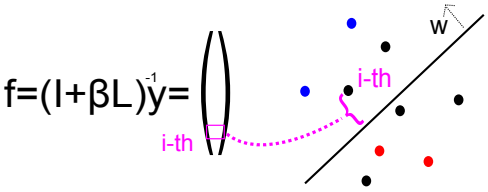


図 1: マージンとしてのスコア

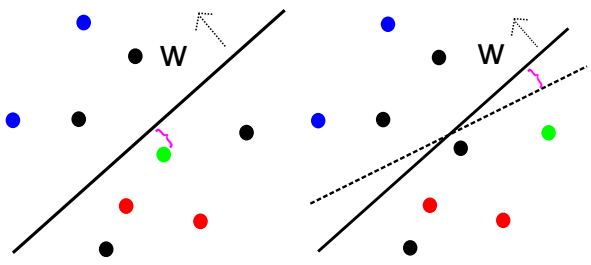
であり，$\Phi\Phi^T = I - D^{-\frac{1}{2}}XX^TD^{-\frac{1}{2}}$ を満たす．Φ の計算量は大きいため計算を避けるべきである．

$$\min_{\mathbf{w}} \sum_{i=1}^{n} \|y_i - \langle \mathbf{w}, \phi(\mathbf{x}_i)\rangle\|^2 + \|\mathbf{w}\|^2 \tag{2}$$

3.2 マージン基準

マージン基準は，現在のモデルにとって判別困難であるようなインスタンスを，良いシード候補とみなす基準である．これは，§3.1 の結果を用いれば，図 2a に示すように，マージンが最小のインスタンスを選択していくことと解釈できる．

図 2a に示したのは 2 クラスの場合であるが，一般の



(a) マージン (Margin)　(b) 期待回転 (EMR)

図 2: 提案する基準

多クラスの場合については次のように表せる．まず，各クラスごとのシード候補値を $g^{\text{Margin}}_{i',k} \stackrel{\text{def}}{=} (\mathbf{f}_k)_{i'}$ と定義する．ただし，以後，$\mathbf{f}_k$ はクラス k のシード数で割られ正規化されているものとする．この $g^{\text{Margin}}_{i',k}$ を用いると，マージン基準によるインスタンス i のシード候補値 g^{Margin}_{i} は，次のように表せる．ただし，$2^{\text{nd}}\text{largest}$ は範囲中で 2 番目に大きな値を返す関数である．

$$g^{\text{Margin}}_{i'} \stackrel{\text{def}}{=} -\left(\max_{k\in C} g^{\text{Margin}}_{i',k} - 2^{\text{nd}}\text{largest}_{k\in C} g^{\text{Margin}}_{i',k}\right)$$

3.3 期待回転基準

期待回転基準は，モデルへの影響が最も大きいと予測されるインスタンスを良いシード候補とみなし，高いシード候補値を与える基準である．モデルへの影響は，図 2b のように，現在の識別平面と，ラベルなしインスタンス i' を加えた時に期待される識別平面との間の角度の大きさ（期待回転量[2]）から計算できる．

この期待回転量は，一般の多クラスの場合については次のように計算できる．まず，全体の期待回転量 $g^{\text{EMR}}_{i'}$ は，i' のラベルが k であった時の回転量 $g^{\text{EMR}}_{i',k}$ に i' のラベルが k である確率をかけて期待値を取ることで，$g^{\text{EMR}}_{i'} \stackrel{\text{def}}{=} \sum_{k\in C} p_{i'}(k)\, g^{\text{EMR}}_{i',k}$ と計算できる．ここで，i' のラベルが k である確率は，$p_{i'}(k) \stackrel{\text{def}}{=} \frac{|(\mathbf{f}_k)_{i'}|}{\sum_{k\in C}|(\mathbf{f}_k)_{i'}|}$ で求める．次に，各 $g^{\text{EMR}}_{i',k}$ は，数式 (3) で求める．

$$g^{\text{EMR}}_{i',k} = 1 - \left|\frac{\mathbf{w}_k}{||\mathbf{w}_k||}^{\top} \frac{\mathbf{w}_{k,+i'}}{||\mathbf{w}_{k,+i'}||}\right| \quad (3)$$

ただし，$\mathbf{w}_k$ はクラス k の識別平面の法線，$\mathbf{w}_{k,i'}$ はラベルなしインスタンス i' がクラス k に分類された時の識別平面の法線である．それぞれ，次のように表される．

$$\mathbf{w}_k = \Phi^{\top}\mathbf{f}_k = \Phi^{\top}(I+\beta L)^{-1}\mathbf{y}_k \quad (4)$$

$$\mathbf{w}_{k,+i'} = \Phi^{\top}\mathbf{f}_{k,+i'} = \Phi^{\top}(I+\beta L)^{-1}(\mathbf{y}_k + \mathbf{e}_{i'}) \quad (5)$$

ただし，$\mathbf{e}_{i'}$ は要素 i' のみ 1 で他は 0 であるような単位ベクトルである．これらの式には Φ が現れるが，Φ を直接計算することなく，次のようにして計算することができるように数式 (3) を設計した．

$$\mathbf{w}_k^{\top}\mathbf{w}_{k,+i'} = \mathbf{f}_k^{\top}\left(I - D^{-\frac{1}{2}}XX^TD^{-\frac{1}{2}}\right)\mathbf{f}_{k,+i'} \quad (6)$$

$$||\mathbf{w}|| = \sqrt{\mathbf{f}^{\top}\left(I - D^{-\frac{1}{2}}XX^TD^{-\frac{1}{2}}\right)\mathbf{f}}$$

[2]Expected Model Rotation, EMR.

4 計算の高速化

4.1 逆行列計算回数の削減

Algorithm 1 においては，毎反復で $\{g_i|i\in U\}$ を計算する必要があるが，$g_{i',k}$ を導入することでこの計算量を削減可能である．例えば期待回転基準においては，$\{g_i|i\in U\}$ の計算に $|U||C|$ 回の逆行列計算を必要とする．しかし，$g_{i',k}$ の導入により，$\hat{i}$ をシードに追加する前後で変化するのは $\{g_{i',k'}|i'\in U\}$ のみとなるので（k' は $\hat{i}$ の真のラベル），$|U|$ 回まで削減できる．さらに，適当な前計算を施しシード追加前後で使いまわせる値を全てキャッシングすることにより，最終的には，k' のスコアベクトルを再計算するための 1 回にまで削減できる．

4.2 逆行列計算自体の高速化

数式 (1) で，$(I+\beta L)^{-1}$ を直接計算することは次の理由により望ましくない．(1) 逆行列の保持には $\mathrm{O}\left(n^2\right)$ サイズのメモリが必要であるため空間計算量が大きい．(2) 時間計算量の観点からも，L は密行列であるため，言語の特性である X の疎性を生かせない．これらの問題は，*splitting* と呼ばれる逆行列計算法を応用する事によって解決できる（Eq. 15.1.21, [3] ）．

今，$\mathbf{f} = A^{-1}\mathbf{y}$ を計算したいとする．この時，$A = M - N$ の形に行列を分解 (split) する．ただし，M^{-1} が存在するとし，$\rho\left(M^{-1}N\right) < 1$ であるとする．この時，$\mathbf{f} \leftarrow M^{-1}N\mathbf{f} + M^{-1}\mathbf{y}$ は，10 進数で毎実行約 $-\log_{10}\rho\left(M^{-1}N\right)$ 桁の収束速度で $A^{-1}\mathbf{y}$ に近づく．$M = I$，$N = \frac{\beta}{1+\beta}\left(D^{-\frac{1}{2}}X\right)\left(D^{-\frac{1}{2}}X\right)^{\top}$ と置けば数式 (1) を疎行列 $\left(D^{-\frac{1}{2}}X\right)$ と密ベクトルの積だけで解くことが可能であり，収束速度は $-\log_{10}\rho\left(\frac{\beta}{1+\beta}\right)$ である．多くの場合高い精度を達成する $\beta = 0.01$ の付近 [6] では，高速に数式 (1) が解けることが分かる．

5 評価

ブートストラップに典型的な情報抽出と，文書分類の 2 タスクで評価した．

5.1 情報抽出タスクによる評価

情報抽出タスクによる評価は，[5] の 3.1 節の実験設定に従った[3]．データは[4]から入手した．この実験は，表形式の知識データベースである Freebase の一部より変換されたグラフから，Wikipedia から取得した星名や作

[3]*Freebase-1 with Pantel Classes*

[4]Freebase-1, `http://www.talukdar.net/datasets/class_inst/`. ただし，Pantel の正解集合は，39 個の重複するインスタンスを除き 1,096 個の多クラス問題とした．

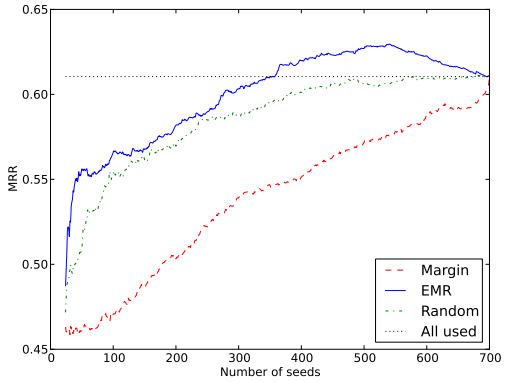


図 3: Freebase-1 の結果（30 回の平均）．提案手法：マージン基準 (Margin)，期待回転基準 (EMR)．ベースライン：Random（ランダムにシードを選択していった場合）．All used: シードプールの全データをシードに用いた場合．

曲家名など $|C| = 23$ クラスを正解集合として抽出するタスクである．グラフは $n = 31,143$ 個のインスタンスと $m = 1,529$ 個の素性からなり，正解集合は 1,096 個である．1,096 個から 700 個のシードプールと，300 個のテスト集合をランダムに作った．シードはシードプールからのみ選んだ．全ての判別は $\beta = 0.01$ に固定した数式 (1) を用い，シード集合の差のみが性能に影響するようにした．データセット分割時の乱数の種を変えて 30 回実行した．シード集合の初期値は，各クラスにつき 1 インスタンスからなる 23 個の集合とした．検定はマン・ホイットニー検定を用いた．評価指標も，[5] に従い，Mean Reciprocal Rank (MRR) を用いた[5]．

シードプール中のシード数に対する MRR を図 3 に示した．図 3 より，EMR は横軸 350 付近で All used を追い越しており，Random は All used を追い越していないので，Random が All used の精度を達成するのに必要な 700 個のシードのラベル付コストが，EMR の使用で約半減したと解釈できる．シード数（横軸）46，シード数 460 の 2 点において，Random と EMR 間に p 値 <0.01 の有意な MRR の差が認められた．

5.2 文書分類による評価

データセットによっては Margin の性能が EMR を上回る．文書分類で標準的な 20 Newsgroup を元に，ラベルの偏りやデータの分離の度合いなどを調整して作成されている 20 Newsgroup Subsets[6]を用いて評価した．最も分離しやすくラベルの偏りがないデータセットである sb-8-1 では，図 4 より，Margin は Random も EMR も

[5]テストセットを Q，全 $|C|$ クラス中の，インスタンス v の真のクラスの順位を r_v とすると，$MRR \stackrel{\text{def}}{=} \frac{1}{|Q|}\sum_{v\in Q}\frac{1}{r_v}$ と定義される．

[6]http://mlg.ucd.ie/datasets/

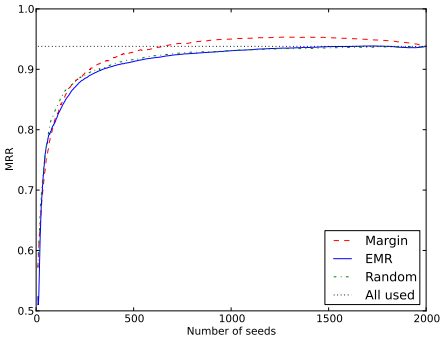


図 4: sb-8-1 の結果（30 回の平均）．凡例は図 3 に同じ．

上回る[7]．§5.1 と §5.2 の結果からは，クラスの偏りがあり分離がしにくい場合には EMR が優位であり，クラスの偏りがなく分離しやすい時は Margin が優位であるという傾向が考察される．

6 おわりに

ブートストラップ法に対する能動学習の枠組みとして，逐次シーディングフレームワークを提案した．シード候補の選択基準として，ラプラシアンラベル伝搬法 [1] のスコアベクトルがマージンと見なせることを利用し，マージン基準と期待回転基準，さらにその高速な計算方法を提案した．評価実験の結果，逐次シーディングを用いることによってラベル付コストが有意に削減できることを確認した．将来の課題としては，マルチラベル問題への拡張などが挙げられる．

参考文献

[1] M. Komachi, T. Kudo, M. Shimbo, and Y. Matsumoto. Graph-based analysis of semantic drift in Espresso-like bootstrapping algorithms. In *Proc. of EMNLP*, pp. 1011–1020, 2008.

[2] Z. Kozareva and E. Hovy. Not all seeds are equal: Measuring the quality of text mining seeds. In *Proc. of NAACL-HLT*, pp. 618–626, 2010.

[3] A. N. Langville and C. D. Meyer. *Google's Pagerank and Beyond: The Science of Search Engine Rankings*. Princeton Univ. Pr., 2006.

[4] P. Pantel and M. Pennacchiotti. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Proc. of ACL-COLING*, pp. 113–120, 2006.

[5] P. P. Talukdar and F. Pereira. Experiments in graph-based semi-supervised learning methods for class-instance acquisition. In *Proc. of ACL*, pp. 1473–1481, 2010.

[6] X. Zhou, M. Belkin, and N. Srebro. An iterated graph laplacian approach for ranking on manifolds. In *Proc. of KDD*, pp. 877–885, 2011.

[7] 江原, 佐藤, 中川. ガウス過程を用いたラプラシアンラベル伝搬法の拡張. NLP 若手の会 第 6 回シンポジウム, 2011.

[7]EMR は Random に対して横軸 500 地点で p 値 <0.01 で有意．