MIRA
(Margin Infused Relaxed Algorithm)

京都大学大学院情報学研究科知能情報学専攻
中澤 敏明
nakazawa@nlp.kuee.kyoto-u.ac.jp

2009/7/31 勉強会

1 Introduction

MIRA(Margin Infused Relaxed Algorithm) はオンライン学習アルゴリズムの一つで、事例と超平面との距離 (*Margin*) をモデル更新に利用 (*Infused*) する。*Relaxed* の気持ちは、[3] で示されている multiclass version Perceptron のパラメータ更新時の制約を緩めているところから来ている (と思われる)。なお MIRA の初出は [3] であり、正確にはこの論文中での定義が MIRA なのだが、この MIRA は分離可能な問題にしか対応しておらず、これを発展させたものが PA(Passive-Aggressive) algorithm[2] となっている。また MIRA を利用したと謳っている研究のほとんどは実際には [2] を使っている。そこで今回は [2] を取り上げ、その内容を理解する。なお本稿の章立ては [2] に倣っているので、適宜元の論文を参照されたい。

一般にオンライン学習では、以下の処理を繰り返すことによりモデルの学習を行う:

1. 事例を一つ観測
2. 事例に対する出力をその時点でのモデルにより生成
3. 出力と正解とを比較してモデルを更新

この一連の処理をここでは *round* と呼ぶことにする。

PA の基本的なアイデアは、学習を「観測している事例が正しく判定できるようにモデルを更新するが、その更新の大きさは最小でなければならない」という最適化問題を解くことに置き換えるということである。この利点は

- 閉じた解が得られる
- 様々な問題を解くための様々なオンラインアルゴリズムの性能 (誤りの上限など) を統一的に分析できる

なおオリジナルの MIRA はモデルの更新の大きさを最小化するのではなく、更新後のパラメータのノルムを最小化するという点が異なる。PA より MIAR の方が計算が複雑なため速度が遅くなり、実装するのも MIRA の方が難しい。しかし最後の実験では精度は MIRA の方が若干よい。

2 Problem Setting

二値分類問題を考える。各 round ごとに、

1. ある事例を観測し、+1 か-1 かのラベルを付与する
2. 正解と比較し、誤りの度合を得る
3. 誤りの度合に応じてモデルを更新

を行う。ある round t における事例とその事例の正解ラベルのペアを $(\mathbf{x}_t, y_t)$ とし、*example* と呼ぶ。ただし $\mathbf{x}_t$ は n 次元の実数ベクトル ($\mathbb{R}^n$) とし、$y_t \in \{+1, -1\}$ とする。

事例を分類するための関数として、重みベクトル $\mathbf{w} \in \mathbb{R}^n$ を考え、round t において用いる重みを $\mathbf{w}_t$ と記す。この重みと事例ベクトル $\mathbf{x}_t$ との内積 $(\mathbf{w}_t \cdot \mathbf{x}_t)$ の符号 (sign$(\mathbf{w}_t \cdot \mathbf{x}_t)$) により事例を分類し、その大きさ ($|\mathbf{w}_t \cdot \mathbf{x}_t|$) により確信度を表す。また $y_t(\mathbf{w}_t \cdot \mathbf{x}_t)$ を *margin* と呼ぶ。sign$(\mathbf{w}_t \cdot \mathbf{x}_t) = y_t$ ($\Leftrightarrow y_t(\mathbf{w}_t \cdot \mathbf{x}_t) > 0$) ならば分類が正しいことになるが、確信度もある程度大きい方が望ましい。そこで確信度が 1 より小さい場合に、以下の hinge-loss function により損失を受けることにする:

$$\ell(\mathbf{w}; (\mathbf{x}, y)) = \begin{cases} 0 & y(\mathbf{w} \cdot \mathbf{x}) \geq 1 \\ 1 - y(\mathbf{w} \cdot \mathbf{x}) & \text{otherwise} \end{cases} \tag{1}$$

簡単のために、round t における損失を ℓ_t ($= \ell(\mathbf{w}_t; (\mathbf{x}_t, y_t))$) と書く。

3 Binary Classification Algorithm

重みベクトルの初期値を 0 ベクトル ($\mathbf{w}_1 = (0, ..., 0)$) とし、*Passive-Aggressive (PA) algorithm* により更新する。

重みベクトルの更新を以下の最適化問題の解として定義する:

$$\mathbf{w}_{t+1} = \underset{\mathbf{w} \in \mathbb{R}^n}{\operatorname{argmin}} \frac{1}{2} ||\mathbf{w} - \mathbf{w}_t||^2 \quad \text{s.t.} \quad \ell(\mathbf{w}; (\mathbf{x}_t, y_t)) = 0 \tag{2}$$

つまり $\mathbf{w}_{t+1}$ は $\mathbf{w}_t$ の hinge-loss が 0 である空間への写像である。

この式では $\ell_t = 0$ ならば何もせず $\mathbf{w}_{t+1} = \mathbf{w}_t$ となり (*passive*)、そうでなければ条件を満たす範囲で $\mathbf{w}_t$ を更新して $\mathbf{w}_{t+1}$ とする (*aggressive*)。つまり現在の事例を正しく分類できるように重みを更新するのだが、更新は最少であり、更新前の重みになるべく近い必要がある。

この最適化問題は以下のように、ラグランジュの未定乗数法を用いて解くことができる。この問題におけるラグランジュ関数は、ラグランジュ乗数を $\tau \geq 0$ として:

$$\mathcal{L}(\mathbf{w}, \tau) = \frac{1}{2} ||\mathbf{w} - \mathbf{w}_t||^2 + \tau(1 - y_t(\mathbf{w} \cdot \mathbf{x}_t)) \tag{3}$$

である。この関数の偏導関数が 0 になる点を求めればよいので、

$$\nabla_{\mathbf{w}} \mathcal{L}(\mathbf{w}, \tau) = \mathbf{w} - \mathbf{w}_t - \tau y_t \mathbf{x}_t = 0 \quad \Longrightarrow \quad \mathbf{w} = \mathbf{w}_t + \tau y_t \mathbf{x}_t \tag{4}$$

これを式 3 に代入すると、

$$\mathcal{L}(\tau) = -\frac{1}{2} \tau^2 ||\mathbf{x}_t||^2 + \tau(1 - y_t(\mathbf{w}_t \cdot \mathbf{x}_t)) \tag{5}$$

さらにこの式を τ について偏微分したものも 0 にならなければならないので、

$$\frac{\partial \mathcal{L}(\tau)}{\partial \tau} = -\tau ||\mathbf{x}_t||^2 + (1 - y_t(\mathbf{w}_t \cdot \mathbf{x}_t)) = 0 \quad \Longrightarrow \quad \tau = \frac{1 - y_t(\mathbf{w}_t \cdot \mathbf{x}_t)}{||\mathbf{x}_t||^2} \tag{6}$$

よって式 1, 4, 6 より、

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \tau_t y_t \mathbf{x}_t \quad \text{where} \quad \tau_t = \frac{\ell_t}{||\mathbf{x}_t||^2} \tag{7}$$

とすればパラメータの更新を行うことができる。

しかしこれだとデータにノイズが含まれていた際に、ノイズによって誤った方向にパラメータが更新されてしまい、その後の分類に悪影響を与えやすい。これを回避し、毎回のパラメータ更新を穏やかにするために、式 2 にスラック変数 ξ を導入する:

$$\mathbf{w}_{t+1} = \underset{\mathbf{w} \in \mathbb{R}^n}{\operatorname{argmin}} \frac{1}{2} ||\mathbf{w} - \mathbf{w}_t||^2 + C\xi \quad \text{s.t.} \quad \ell(\mathbf{w}; (\mathbf{x}_t, y_t)) \leq \xi \text{ and } \xi \geq 0 \tag{8}$$

ここで C はスラック変数がパラメータ更新に与える影響の大きさをコントロールする正のパラメータで、*aggressiveness parameter* と呼ぶ。この場合、パラメータの更新は PA と同じく $\mathbf{w}_{t+1} = \mathbf{w}_t + \tau_t y_t \mathbf{x}_t$ となり、

$$\tau_t = \min\left\{C, \frac{\ell_t}{||\mathbf{x}_t||^2}\right\} \qquad \text{(PA-I)} \tag{9}$$

つまり PA では損失が大きくなればなるほどパラメータ更新の割合も大きくなるが、その上限を C で抑えていることになる。

さらに ξ を二乗した以下の式も考えられる:

$$\mathbf{w}_{t+1} = \underset{\mathbf{w}\in\mathbb{R}^n}{\operatorname{argmin}} \frac{1}{2}||\mathbf{w} - \mathbf{w}_t||^2 + C\xi^2 \quad \text{s.t.} \quad \ell(\mathbf{w}; (\mathbf{x}_t, y_t)) \leq \xi \tag{10}$$

この場合は

$$\tau_t = \frac{\ell_t}{||\mathbf{x}_t||^2 + \frac{1}{2C}} \qquad \text{(PA-II)} \tag{11}$$

この場合はパラメータの更新を C によって若干和らげることになる (C が小さければ τ_t も小さくなり、C 大きければ τ_t も大きくなる)。

さらに式 7 を繰り返し用いることにより、

$$\mathbf{w}_t = \sum_{i=1}^{t-1} \tau_i y_i \mathbf{x}_i \tag{12}$$

であり、両辺に $\mathbf{x}_t$ をかけると、

$$\mathbf{w}_t \cdot \mathbf{x}_t = \sum_{i=1}^{t-1} \tau_i y_i (\mathbf{x}_i \cdot \mathbf{x}_t) \tag{13}$$

となり、この右辺のカッコ内は一般的な Mercer カーネルに置き換えることができる (SVM でいうカーネルトリックが使える)。

4 Analysis

前章の各アルゴリズムにおける損失の境界について議論する (それぞれの証明は元の論文を参照してください)。なおここでの境界の議論は以後出てくる様々な問題設定における PA の適用時にも全て同等に成り立つ (ので以後では省略)。

ℓ_t を round t における損失とし、ℓ_t^* を任意の重みベクトル $\mathbf{u} \in \mathbb{R}^n$ による損失とする。つまり、

$$\ell_t = \ell(\mathbf{w}_t; (\mathbf{x}_t, y_t)) \quad \text{and} \quad \ell_t^* = \ell(\mathbf{u}_t; (\mathbf{x}_t, y_t)) \tag{14}$$

まず以下の補助定理を導く。

Lemma 1 $(\mathbf{x}_1, y_1), ..., (\mathbf{x}_T, y_T)$ を example 列とする (ただし全ての t において $\mathbf{x}_t \in \mathbb{R}^n$、$y_t \in \{+1, -1\}$)。$\tau_t$ を前章のラグランジュ乗数とすると、任意の $\mathbf{u} \in \mathbb{R}^n$ に対して以下の式が成り立つ:

$$\sum_{t=1}^{T} \tau_t (2\ell_t - \tau_t ||\mathbf{x}_t||^2 - 2\ell_t^*) \leq ||\mathbf{u}||^2 \tag{15}$$

分離可能な問題での累積二乗損失の上界について以下の定理が成り立つ。

Theorem 2 $(\mathbf{x}_1, y_1), ..., (\mathbf{x}_T, y_T)$ を example 列とする (ただし全ての t において $\mathbf{x}_t \in \mathbb{R}^n$、$||\mathbf{x}_t|| \leq R$、$y_t \in \{+1, -1\}$)。すべての t において $\ell_t^* = 0$ となるベクトル $\mathbf{u}$ が存在すると仮定すると、累積二乗損失の上界は以下のようになる:

$$\sum_{t=1}^{T} \ell_t^2 \leq ||\mathbf{u}||^2 R^2 \tag{16}$$

これは分離可能な問題に対する Perceptron での上界 [8] と一致する (ただし [8] は誤り回数の上界であるのに対し、この定理では累積二乗損失の上界である)。

分離不能な問題での累積二乗損失の上界について以下の定理が成り立つ。

Theorem 3 $(\mathbf{x}_1, y_1), ..., (\mathbf{x}_T, y_T)$ を example 列とする (ただし全ての t において $\mathbf{x}_t \in \mathbb{R}^n$、$||\mathbf{x}_t|| = 1$、$y_t \in \{+1, -1\}$)。任意のベクトル $\mathbf{u}$ に対して、累積二乗損失の上界は以下のようになる:

$$\sum_{t=1}^{T} \ell_t^2 \leq \left(||\mathbf{u}|| + 2\sqrt{\sum_{t=1}^{T} (\ell_t^*)^2} \right)^2 \tag{17}$$

PA-I による分類誤り回数の上界について以下の定理が成り立つ。

Theorem 4 $(\mathbf{x}_1, y_1), ..., (\mathbf{x}_T, y_T)$ を example 列とする (ただし全ての t において $\mathbf{x}_t \in \mathbb{R}^n$、$||\mathbf{x}_t|| \leq R$、$y_t \in \{+1, -1\}$)。任意のベクトル $\mathbf{u}$ に対して、PA-I による分類誤り回数の上界は Aggressiveness Parameter C を用いて以下のようになる:

$$\max\{R^2, 1/C\} \left(||\mathbf{u}||2 + 2C \sum_{t=1}^{T} \ell_t^* \right) \tag{18}$$

これは分離不能な問題に対する Perceptron での誤り回数の上界 [5] と比較すると、どんなに悪くても高々2 倍である。

PA-II による累積二乗損失の上界について以下の定理が成り立つ。

Theorem 5 $(\mathbf{x}_1, y_1), ..., (\mathbf{x}_T, y_T)$ を example 列とする (ただし全ての t において $\mathbf{x}_t \in \mathbb{R}^n$、$||\mathbf{x}_t||^2 \leq R^2$、$y_t \in \{+1, -1\}$)。任意のベクトル $\mathbf{u}$ に対して、PA-II による累積二乗損失の上界は Aggressiveness Parameter C を用いて以下のようになる:

$$\sum_{t=1}^{T} \ell_t^2 \leq \left(R^2 + \frac{1}{2C} \right) \left(||\mathbf{u}||^2 + 2C \sum_{t=1}^{T} (\ell_t^*)^2 \right) \tag{19}$$

これは $C = ||\mathbf{u}||/(2R\sqrt{\sum_t (\ell_t^*)^2})$ とすることによって [4] による誤り回数の上界と一致する。

5 Regression

Regression(回帰) 問題に PA を適用する。問題設定として以下を考える。各事例 $\mathbf{x}_t \in \mathbb{R}^n$ には実数値の正解 $y_t \in \mathbb{R}$ が対応付けられている。重みベクトル $\mathbf{w}_t \in \mathbb{R}^n$ を用いて、事例に対する値を線形回帰関数 $\hat{y}_t = \mathbf{w}_t \cdot \mathbf{x}_t$ により推定する。この推定値と正解とを比較し、以下の ε-insensitive hinge loss 関数により損失を計算する:

$$\ell_\varepsilon(\mathbf{w}; (\mathbf{x}, y)) = \begin{cases} 0 & |\mathbf{w} \cdot \mathbf{x} - y| \leq \varepsilon \\ |\mathbf{w} \cdot \mathbf{x} - y| - \varepsilon & \text{otherwise} \end{cases} \tag{20}$$

ε は推定誤りに対する感度をコントロールする正のパラメータである。各 round t において $\mathbf{w}$ を式 2 と全く同様に更新する。この解は以下のようになる:

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \text{sign}(y_t - \hat{y}_t)\tau_t \mathbf{x}_t \quad \text{where} \quad \tau_t = \frac{\ell_t}{||\mathbf{x}_t||^2} \tag{21}$$

PA-I と PA-II についてもスラック変数を導入することにより全く同様に定義でき、$\mathbf{w}_{t+1} = \mathbf{w}_t + \text{sign}(y_t - \hat{y}_t)\tau_t \mathbf{x}_t$ により重みが更新される。なお τ_t についてはそれぞれ式 9 と式 11 と同じものになる。

6 Uniclass Prediction

単一クラス予測問題に PA を適用する。各 round t において、予測ベクトル $\hat{y}_t \in \mathbb{R}^n$ を出力する。ただし入力 $(\mathbf{x})$ はなく、t におけるベクトル $\mathbf{w}_t \in \mathbb{R}^n$ をそのまま出力するものとする ($\hat{y}_t = \mathbf{w}_t$)。正

解ベクトルと比較することにより損失を計算する:

$$\ell_\varepsilon(\mathbf{w};\mathbf{y}) = \begin{cases} 0 & ||\mathbf{w}-\mathbf{y}|| \leq \varepsilon \\ ||\mathbf{w}-\mathbf{y}|| - \varepsilon & \text{otherwise} \end{cases} \tag{22}$$

この損失を用いて $\mathbf{w}$ を更新する。これは $\mathbf{w}$ を中心とする半径 ε の球の内部により多くの $\mathbf{y}$ が入るように最適化する問題と考えることができる。

この問題においても、$\mathbf{w}$ を式 2 と全く同様に更新する。この解は $\tau_t = \ell_\varepsilon$ として以下のようになる:

$$\begin{aligned} \mathbf{w}_{t+1} &= \left(1 - \frac{\ell_t}{||\mathbf{w}_t - \mathbf{y}_t||}\right)\mathbf{w}_t + \left(\frac{\ell_t}{||\mathbf{w}_t - \mathbf{y}_t||}\right)\mathbf{y}_t \\ &= \mathbf{w}_t + \tau_t \frac{\mathbf{y}_t - \mathbf{w}_t}{||\mathbf{y}_t - \mathbf{w}_t||} \end{aligned} \tag{23}$$

これが求める最適解であることを示すためには、KKT 条件を全て満たせばよい。最適化問題のラグランジュ関数は:

$$\mathcal{L}(\mathbf{w},\tau) = \frac{1}{2}||\mathbf{w}-\mathbf{w}_t||^2 + \tau(||\mathbf{w}-\mathbf{y}_t|| - \varepsilon) \tag{24}$$

となるので、$(\mathbf{w}_{t+1}, \tau_t)$ が以下の KKT 条件を全て満たすことを示せば良い (証明は論文で):

- $\nabla_{\mathbf{w}}\mathcal{L}(\mathbf{w},\tau) = \mathbf{w} - \mathbf{w}_t + \tau\frac{\mathbf{w}-\mathbf{y}_t}{||\mathbf{w}-\mathbf{y}_t||} = 0$
- $\tau \geq 0$
- $\tau(||\mathbf{w}-\mathbf{y}_t|| - \varepsilon) = 0$

PA-I についても同様にパラメータの更新を以下のように定義できる:

$$\begin{gathered} \mathbf{w}_{t+1} = \underset{\mathbf{w}\in\mathbb{R}^n}{\operatorname{argmin}} \frac{1}{2}||\mathbf{w}-\mathbf{w}_t||^2 + C\xi \quad \text{s.t.} \quad ||\mathbf{w}-\mathbf{y}_t|| \leq \varepsilon + \xi \ \text{ and } \ \xi \geq 0 \\ \text{where} \quad \tau_t = \min\{C, \ell_t\} \end{gathered} \tag{25}$$

PA-II は以下のようになる:

$$\begin{gathered} \mathbf{w}_{t+1} = \underset{\mathbf{w}\in\mathbb{R}^n}{\operatorname{argmin}} \frac{1}{2}||\mathbf{w}-\mathbf{w}_t||^2 + C\xi^2 \quad \text{s.t.} \quad ||\mathbf{w}-\mathbf{y}_t|| \leq \varepsilon + \xi \\ \text{where} \quad \tau_t = \frac{\ell_t}{1 + \frac{1}{2C}} \end{gathered} \tag{26}$$

次にこれまであらかじめ固定していた ε も学習することを考える。つまり可能な限り

$$||\mathbf{w}_t - \mathbf{y}_t|| \leq \varepsilon_t \tag{27}$$

となるような $\mathbf{w}_t$ と ε_t を求める。ただし ε_t はなるべく小さくする。

まず ε の上限を B とし、この値は固定であるとする。式 27 は B を用いて以下のように書き換えられる:

$$||\mathbf{w}_t - \mathbf{y}_t||^2 + (B^2 - \varepsilon_t^2) \leq B^2 \tag{28}$$

ここで $\mathbf{y}_t$ を $n+1$ 次元に拡張し、$n+1$ 番目の座標を 0 とする。さらに $\mathbf{w}_t$ も $n+1$ 次元に拡張し、$n+1$ 番目の座標 ($w_{t,n+1}$ と書く) を $\sqrt{B^2 - \varepsilon_t^2}$ とすると、式 28 は $||\mathbf{w}_t - \mathbf{y}_t||^2 \leq B^2$ となり、半径を B に固定した場合と同様の問題に帰着する。$\varepsilon_1 = 0 (\Leftrightarrow w_{1,n+1} = B)$ とすると、各 round において ε_t は $\varepsilon_t = \sqrt{B^2 - w_{t,n+1}^2}$ と求められる。

なお $w_{t+1,n+1}$ は式 23 より $w_{t,n+1}$ と $y_{t,n+1}(=0)$) との凸結合[1]であることに注意すると、$w_{t,n+1}$ は $(0, B]$ を動き、かつ各 round において変化しないか、小さくなる。逆に ε は 0 を初期値として各 round ごとに変化しない (損失が 0 のとき) か、大きくなる。

[1]線形結合のうち、結合係数が全て非負であり、かつその和が 1 となるもの

7 Multiclass Problems

ある事例 $\mathbf{x}_t$ に対し、あらかじめ決められたラベルセット $\mathcal{Y} = \{1, ..., k\}$ のそれぞれに対するスコアを出力する (ラベルをランキングする) ことにより、その事例のラベルを推定するという問題を考える。つまり $\mathbf{x}_t$ を入力として k 次元のベクトルを出力する。これは text categorization などに用いられる。なお正解のラベルは複数あってもよいものとし、この場合、推定が正しいとは、全ての正解ラベルが全ての不正解ラベルよりも上位にランクされることを指す。

いま d 個の素性 $\phi_1, ..., \phi_d$ を考え、これらを入力事例 $\mathbf{x}$ とラベル $y \in \mathcal{Y}$ とのペア $(\mathbf{x}, y)$ に適用したものを、ベクトルとして $\Phi(\mathbf{x}, y) = (\phi_1(\mathbf{x}, y), ..., \phi_d(\mathbf{x}, y))$ と表記する。round t におけるシステムの出力は、d 次元の重みベクトル $\mathbf{w} \in \mathbb{R}^d$ を用いて、以下のような k 次元ベクトルとする:

$$(\mathbf{w}_t \cdot \Phi(\mathbf{x}_t, 1), ..., \mathbf{w}_t \cdot \Phi(\mathbf{x}_t, k)) \tag{29}$$

例えば素性として各単語 μ_j の TF-IDF を用いるとする (素性の数は単語の異なり数となる)。S をドキュメントの集合、$\mathbf{x} \in S$ を一つのドキュメント、y を一つのトピックとすると、素性 $\phi_j(\mathbf{x}, y)$(単語 μ_j に関する素性) は以下のように書ける:

$$\phi_j(\mathbf{x}, y) = \text{TF}(\mu_j, \mathbf{x}) \cdot \log\left(\frac{|S|}{\text{DF}(\mu_j, y)}\right) \tag{30}$$

$\text{TF}(\mu_j, \mathbf{x})$ はドキュメント $\mathbf{x}$ 中での単語 μ_j の出現回数であり、$\text{DF}(\mu_j, y)$ は y **以外**のラベルが付与された文書のうち、μ_j が出現する文書の数である。つまり、各素性はラベル (= トピック) に依存することになる。

round t における事例 $\mathbf{x}_t$ に対する正解ラベルセットを Y_t とし、推定されたラベルと正解との *margin* を以下のように定義する:

$$\gamma(\mathbf{w}_t; (\mathbf{x}_t, Y_t)) = \min_{r \in Y_t} \mathbf{w}_t \cdot \Phi(\mathbf{x}_t, r) - \max_{s \notin Y_t} \mathbf{w}_t \cdot \Phi(\mathbf{x}_t, s) \tag{31}$$

つまり「正解のラベルに対して推定されたスコアの最小値 − 不正解のラベルに対して推定されたスコアの最大値」を margin とする。この定義により、最小の正解ラベルと最大の不正解ラベル以外は無視されることになる。この定義では全ての正解ラベルが不正解のラベルよりも上位にランクされたときのみ、margin が正となる。さらに、margin は少なくとも 1 以上あって欲しいと考えて、以下のような hinge-loss 関数を考える:

$$\ell_{\text{MC}}(\mathbf{w}; (\mathbf{x}, Y)) = \begin{cases} 0 & \gamma(\mathbf{w}; (\mathbf{x}, Y)) \geq 1 \\ 1 - \gamma(\mathbf{w}; (\mathbf{x}, Y)) & \text{otherwise} \end{cases} \tag{32}$$

PA と同様に、重みベクトルの更新を以下のように定義する:

$$\mathbf{w}_{t+1} = \underset{\mathbf{w} \in \mathbb{R}^n}{\text{argmin}} \frac{1}{2}||\mathbf{w} - \mathbf{w}_t||^2 \quad \text{s.t.} \quad \ell_{\text{MC}}(\mathbf{w}; (\mathbf{x}_t, Y_t)) = 0 \tag{33}$$

この条件部は、以下の複数の不等式をすべて満たすことと同値である:

$$\forall r \in Y_t \quad \forall s \notin Y_t \quad \mathbf{w} \cdot \Phi(\mathbf{x}_t, r) - \mathbf{w} \cdot \Phi(\mathbf{x}_t, s) \geq 1 \tag{34}$$

しかしここではこのうちの一つの条件 (margin が最大となる条件) のみに注目すればよい。

$$r_t = \underset{r \in Y_t}{\text{argmin}}\, \mathbf{w}_t \cdot \Phi(\mathbf{x}_t, r), \qquad s_t = \underset{s \notin Y_t}{\text{argmax}}\, \mathbf{w}_t \cdot \Phi(\mathbf{x}_t, s) \tag{35}$$

とおくと、上式は以下のように書き換えられる:

$$\mathbf{w}_{t+1} = \underset{\mathbf{w} \in \mathbb{R}^n}{\text{argmin}} \frac{1}{2}||\mathbf{w} - \mathbf{w}_t||^2 \quad \text{s.t.} \quad \mathbf{w} \cdot \Phi(\mathbf{x}_t, r_t) - \mathbf{w} \cdot \Phi(\mathbf{x}_t, s_t) \geq 1 \tag{36}$$

これを解くと、$\ell_t = \ell_{\text{MC}}(\mathbf{w}_t; (\mathbf{x}_t, Y_t))$ とおいて:

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \tau_t(\Phi(\mathbf{x}_t, r_t) - \Phi(\mathbf{x}_t, s_t)) \quad \text{where} \quad \tau_t = \frac{\ell_t}{||\Phi(\mathbf{x}_t, r_t) - \Phi(\mathbf{x}_t, s_t)||^2} \tag{37}$$

これは Binary Classification(式 2) において $\mathbf{x}_t = \Phi(\mathbf{x}_t, r_t) - \Phi(\mathbf{x}_t, s_t)$、$y_t = +1$ とおけば容易に想像がつく。

同様に PA-I、PA-II については、

$$\tau_t = \min\left\{C, \frac{\ell_t}{||\Phi(\mathbf{x}_t, r_t) - \Phi(\mathbf{x}_t, s_t)||^2}\right\} \text{ (PA-I)} \quad \tau_t = \frac{\ell_t}{||\Phi(\mathbf{x}_t, r_t) - \Phi(\mathbf{x}_t, s_t)||^2 + \frac{1}{2C}} \text{ (PA-II)} \tag{38}$$

Multi-prototype Classification　ここまでの問題設定では、ラベルに依存する素性関数を入力事例 $\mathbf{x}_t$ に対して適用した結果をベクトルとし ($\Phi(\mathbf{x}_t, r)$)、このベクトルをただ一つの重みベクトル $\mathbf{w}$ で重みづけしたものを全てのラベルについて書きならべたベクトル

$$(\mathbf{w}_t \cdot \Phi(\mathbf{x}_t, 1), ..., \mathbf{w}_t \cdot \Phi(\mathbf{x}_t, k)) \tag{39}$$

を出力としていた (*single-prototype multiclass setting*) が、これはいささか不自然である。そこでラベルごとに異なる k 個の重みベクトル $\mathbf{w}_t^r \in \mathbb{R}^n$ を用意し、これを $\mathbf{x}_t$ に対して適用したものを書きならべたベクトル

$$(\mathbf{w}_t^1 \cdot \mathbf{x}_t, ..., \mathbf{w}_t^k \cdot \mathbf{x}_t) \tag{40}$$

を出力するという設定 (*multi-prototype multiclass setting*) を考え、これを single-prototype における技術を使って解決する。

まず $\Phi(\mathbf{x}, y)$ を $k \cdot n$ 次元のベクトル (サイズ n のベクトルが k 個集まったもの) とし、y 番目のベクトルは $\mathbf{x}$ であり、それ以外は 0 であるとする。同様に重みベクトル $\mathbf{w}_t$ も $k \cdot n$ 次元のベクトルであるとし、r 番目のベクトルを $\mathbf{w}_t^r$ で表すものとする。これらを用いると $\mathbf{w}_t \cdot \Phi(\mathbf{x}_t, r) = \mathbf{w}_t^r \cdot \mathbf{x}_t$ となり、

$$\begin{aligned} r_t &= \operatorname*{argmin}_{r \in Y_t} \mathbf{w}_t \cdot \Phi(\mathbf{x}_t, r) = \operatorname*{argmin}_{r \in Y_t} \mathbf{w}_t^r \cdot \mathbf{x}_t \\ s_t &= \operatorname*{argmax}_{s \notin Y_t} \mathbf{w}_t \cdot \Phi(\mathbf{x}_t, s) = \operatorname*{argmax}_{s \notin Y_t} \mathbf{w}_t^s \cdot \mathbf{x}_t \end{aligned} \tag{41}$$

$$\ell(\mathbf{w}_t^1, ..., \mathbf{w}_t^k; (\mathbf{x}_t, Y_t)) = \begin{cases} 0 & \mathbf{w}_t^{r_t} \cdot \mathbf{x}_t - \mathbf{w}_t^{s_t} \cdot \mathbf{x}_t \geq 1 \\ 1 - \mathbf{w}_t^{r_t} \cdot \mathbf{x}_t + \mathbf{w}_t^{s_t} \cdot \mathbf{x}_t & \text{otherwise} \end{cases} \tag{42}$$

$\Phi(\mathbf{x}_t, r_t) - \Phi(\mathbf{x}_t, s_t)$ が r_t 番目が $\mathbf{x}$、s_t 番目が $-\mathbf{x}$ である $k \cdot n$ 次元ベクトルであることに注意すると、式 37 による更新は、

$$\mathbf{w}_{t+1}^{r_t} = \mathbf{w}_t^{r_t} + \tau_t \mathbf{x}_t \quad \text{and} \quad \mathbf{w}_{t+1}^{s_t} = \mathbf{w}_t^{s_t} + \tau_t \mathbf{x}_t \tag{43}$$

$$\text{where} \quad \tau_t = \frac{\ell_t}{||\Phi(\mathbf{x}_t, r_t) - \Phi(\mathbf{x}_t, s_t)||^2} = \frac{\ell_t}{2||\mathbf{x}_t||^2} \tag{44}$$

8 Cost-Sensitive Multiclass Classification

誤りの種類によって cost(損失) が異なる場合を考える。multiclass single-label classification では、推定されたラベル

$$\hat{y}_t = \operatorname*{argmax}_{y \in \mathcal{Y}} (\mathbf{w}_t \cdot \Phi(\mathbf{x}_t, y)) \tag{45}$$

を出力し、これが正解ラベル y と異なる場合に誤りとなる。ラベルのペア (y, y') に対してコスト $\rho(y, y')$ を定義する (y が正解で y' が出力)。全ての $y \in \mathcal{Y}$ について $\rho(y, y) = 0$ とし、$y \neq y'$ ならば $\rho(y, y') \geq 0$ とする。目標は、$\sum_t \rho(y_t, \hat{y}_t)$ を最小化することである。

パラメータ更新にこのコスト関数を導入することにより PA を用いてこの問題を解く。更新の制約条件である式 34 は、この問題では以下のようになる[2]:

$$\forall r \in \{\mathcal{Y} \setminus y_t\} \quad \mathbf{w}_t \cdot \Phi(\mathbf{x}_t, y_t) - \mathbf{w}_t \cdot \Phi(\mathbf{x}_t, r) \geq \sqrt{\rho(y_t, r)} \tag{46}$$

前章では margin が最大となる条件のみに注目したが、ここでもこのうちの一つの条件のみに注目する。この一つの条件の選びかたは二通りある。

prediction-based update　推定されたラベル $\hat{y}_t$ による条件を採用する。重みベクトルの更新は以下の式により行われる:

$$\mathbf{w}_{t+1} = \underset{\mathbf{w} \in \mathbb{R}^n}{\operatorname{argmin}} \frac{1}{2}||\mathbf{w} - \mathbf{w}_t||^2 \quad \text{s.t.} \quad \mathbf{w} \cdot \Phi(\mathbf{x}_t, y_t) - \mathbf{w} \cdot \Phi(\mathbf{x}_t, \hat{y}_t) \geq \sqrt{\rho(y_t, \hat{y}_t)} \tag{47}$$

また損失を以下のように定義する:

$$\ell_{\text{PB}}(\mathbf{w}; (\mathbf{x}, y)) = \mathbf{w} \cdot \Phi(\mathbf{x}_t, \hat{y}_t)) - \mathbf{w} \cdot (\Phi(\mathbf{x}_t, y_t) + \sqrt{\rho(y_t, \hat{y}_t)} \tag{48}$$

これは $\hat{y}_t = y_t$ のときのみ 0 となる。最適化問題の解は

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \tau_t(\Phi(\mathbf{x}_t, y_t) - \Phi(\mathbf{x}_t, \hat{y}_t)) \quad \text{where} \quad \tau_t = \frac{\ell_{\text{PB}}(\mathbf{w}_t; (\mathbf{x}_t, y_t))}{||\Phi(\mathbf{x}_t, y_t) - \Phi(\mathbf{x}_t, \hat{y}_t)||^2} \tag{49}$$

となり、PA-I、PA-II については

$$\tau_t = \min\left\{C, \frac{\ell_{\text{PB}}(\mathbf{w}_t; (\mathbf{x}_t, y_t))}{||\Phi(\mathbf{x}_t, y_t) - \Phi(\mathbf{x}_t, \hat{y}_t)||^2}\right\} \text{ (PA-I)} \quad \tau_t = \frac{\ell_{\text{PB}}(\mathbf{w}_t; (\mathbf{x}_t, y_t))}{||\Phi(\mathbf{x}_t, y_t) - \Phi(\mathbf{x}_t, \hat{y}_t)||^2 + \frac{1}{2C}} \text{ (PA-II)} \tag{50}$$

max-loss update　以下の式により求められる $\tilde{y}_t$ による条件を採用する:

$$\tilde{y}_t = \underset{r \in \mathcal{Y}}{\operatorname{argmax}} \left(\mathbf{w}_t \cdot \Phi(\mathbf{x}_t, r) - \mathbf{w}_t \cdot \Phi(\mathbf{x}_t, y_t) + \sqrt{\rho(y_t, r)}\right) \tag{51}$$

つまり $\tilde{y}_t$ は損失が最も大きくなるようなラベルである。注意すべきは、推定されたラベル (出力) はあくまでも $\hat{y}_t$ であり、$\tilde{y}_t$ はパラメータの更新にのみ用いられるということである。

最適化問題とその解は、prediction-based update の式を $\hat{y}_t \to \tilde{y}_t$、$\ell_{\text{PB}}(\mathbf{w}; (\mathbf{x}, y)) \to \ell_{\text{ML}}(\mathbf{w}; (\mathbf{x}, y)) = \Phi(\mathbf{x}_t, \tilde{y}_t)) - \mathbf{w} \cdot (\Phi(\mathbf{x}_t, y_t) + \sqrt{\rho(y_t, \tilde{y}_t)}$ に置き換えたものとなる。

なお max-loss update は $\tilde{y}$ を求めるための計算量がかなり多いため、prediction-based update よりもスピードが遅くなる。

9 Learning with Structured Output

前章の cost-sensitive アルゴリズムを使って、出力および正解が構造的 (ex. 文字列、グラフ構造、木構造) である場合について考える。基本的には前章と全く同様に問題が解けるのだが、コスト関数 ρ や素性ベクトル Φ の定義を工夫する必要がある。

例えば出力と正解が k 種類のアルファベット m 文字からなるとする。d 個の素性を入力 $\mathbf{x}$ とその正解 $\mathbf{y}$ にそれぞれ適用したもの並べたベクトルを $\Phi(\mathbf{x}, \mathbf{y})$ とすると、各 round において出力は

$$\hat{\mathbf{y}}_t = \underset{\mathbf{y} \in \mathcal{Y}^m}{\operatorname{argmax}}(\mathbf{w}_t \cdot \Phi(\mathbf{x}_t, \mathbf{y})) \tag{52}$$

となる。しかし $\hat{\mathbf{y}}_t$ を得るためには単純には k^m 回の計算が必要であり、m が大きくなると扱えなくなる。これを解決するためには Φ に何らかの制約をつける必要がある。例えば各素性 ϕ_j を以下のような一次マルコフ過程のような定義にすることが考えられる:

$$\phi_j(\mathbf{x}_t, \mathbf{y}) = \Psi_j^0(y_1, \mathbf{x}_t) + \sum_{i=2}^{m} \Psi_j(y_i, y_{i-1}, \mathbf{x}_t) \tag{53}$$

ここで Ψ_j^0 と Ψ_j は任意の計算可能な関数とする。

ρ については、$\rho(\mathbf{y}, \mathbf{y}') = \sum_{i=1}^{m} \hat{\rho}(y_i, y_i')$ などとすればよい。

[2]ここで ρ のルートを用いている理由は、章のように累積二乗損失の上限の議論を行うためのものであり、ルートを用いなければならないわけではない

10 MIRA を使った研究

既存の研究で MIRA を利用したオンライン学習を用いているものをいくつか紹介する。なお詳しい内容は各論文をあたってもらうことにして、ここではトレーニングに関する部分だけを抜粋する。

10.1 構文解析 (McDonald et al., ACL 2005)

McDonald ら [7] は依存構造解析器のトレーニングに MIRA を用いている。入力文 $\mathbf{x}$ の一つの構文木 $\mathbf{y}$ のスコアを、構文木内の全ての枝のスコアの和として以下のように計算する:

$$s(\mathbf{x}, \mathbf{y}) = \sum_{(i,j)\in\mathbf{y}} s(i,j) = \sum_{(i,j)\in\mathbf{y}} \mathbf{w}\cdot\mathbf{f}(i,j) \tag{54}$$

パラメータの更新は

$$\begin{aligned} &\min||\mathbf{w}^{i+1} - \mathbf{w}^{i}|| \\ \text{s.t.} \quad &\forall \mathbf{y}' \in \text{best}_k(\mathbf{x}_t; \mathbf{w}^i): \quad s(\mathbf{x}_t, \mathbf{y}_t) - s(\mathbf{x}_t, \mathbf{y}'; \mathbf{w}) \geq L(\mathbf{y}_t, \mathbf{y}') \end{aligned} \tag{55}$$

により行う。$\text{best}_k(\mathbf{x}_t; \mathbf{w}^i)$ は $s(\mathbf{x}_t, \mathbf{y})$ が高いものから k 個取る関数であり、$L(\mathbf{y}_t, \mathbf{y}')$ は正解の構文木 $\mathbf{y}_t$ から見た構文木 $\mathbf{y}'$ の損失であり、“誤った親を持つ単語の数” としている。このため、損失の最大値は文の単語数となる。

10.2 形態素解析 (Canasai et al., ACL 2009)

Canasai ら [6] は中国語の形態素解析において MIRA を利用している。Canasai らは形態素解析を、“単語ノードと文字ノードとを同時に利用するラティスから正解のパスを探すタスク” と定義している。パラメータの更新は McDonald ら [7] と同様の式で行う。明示的には書かれていないが、$s(\mathbf{x}, \mathbf{y}) = \mathbf{w}\cdot\mathbf{f}(\mathbf{x}, \mathbf{y})$ としている。

損失関数 $L(\mathbf{y}_t, \mathbf{y}')$ は false positive=*FP*(出力されたノードのうち、正解パス内に含まれないものの個数) と false negative=*FN*(正解パスにあるノードのうち、出力に含まれていないものの個数) の和として $L(\mathbf{y}_t, \mathbf{y}') = FP + FN$ と定義している。

10.3 翻訳でのパラメータチューニング (Watanabe et al., EMNLP 2007, Chiang et al., NAACL 2009)

Watanabe ら [9] や Chiang ら [1] は翻訳で用いるパラメータのチューニングに MIRA を利用している。入力文 f に対する出力 (翻訳) 文 $\hat{e}$ は $\hat{e} = \text{argmax}_e\, \mathbf{w}\cdot\mathbf{h}(f, e)$ により得られる。ここで $\mathbf{h}(f, e)$ は素性ベクトルである。Watanabe ら [9] は重みベクトル $\mathbf{w}$ の更新を、非負のスラック変数 $\xi(\hat{e}, e')$ を用いて以下のように表している:

$$\begin{aligned} &\hat{\mathbf{w}}^{i+1} = \underset{\mathbf{w}^{i+1}}{\text{argmin}}\, ||\mathbf{w}^{i+1} - \mathbf{w}^{i}|| + C\sum_{\hat{e},e'} \xi(\hat{e}, e') \\ \text{s.t.} \quad &s^{i+1}(f^t, \hat{e}) - s^{i+1}(f^t, e') + \xi(\hat{e}, e') \geq L(\hat{e}, e'; \mathbf{e}^t), \quad \xi(\hat{e}, e') \geq 0, \quad \forall \hat{e} \in \mathcal{O}^t, \quad \forall e' \in \mathcal{C}^t \end{aligned} \tag{56}$$

ここで $s^i(f^t, e) = \mathbf{w}^i\cdot\mathbf{h}(f^t, e)$ であり、$C \geq 0$ は aggressiveness parameter である。L は損失関数であり、$\mathbf{e}^t$ をリファレンスとした $\hat{e}$ と e' との BLEU 値の差である。また $\hat{e}$ は正解の翻訳 (だと思っているもの) である。

Chiang ら [1] は入力文を $\mathbf{f}_1, ..., \mathbf{f}_m$、各 $\mathbf{f}_i$ に対する翻訳候補を $\mathbf{e}_{i1}, ..., \mathbf{e}_{in}$ とし、パラメータの更新を以下の式で行う:

$$\mathbf{w}' = \underset{\mathbf{w}'}{\text{argmin}}\, \frac{1}{2}||\mathbf{w}' - \mathbf{w}||^2 + C\sum_{i=1}^{m} \max_{1\leq j\leq n} (\ell_{ij} - \Delta\mathbf{h}_{ij}\cdot\mathbf{w}') \tag{57}$$

ここで ℓ_{ij} は損失関数であり、正解の翻訳 $\mathbf{e}_i^*$ と $\mathbf{e}_{ij}$ との BLEU 値の差であり、$\Delta\mathbf{h}_{ij} = \mathbf{h}(\mathbf{e}_i^*) - \mathbf{h}(\mathbf{e}_{ij})$ である。

Appendix A. ラグランジュの未定乗数法

Wikipedia によると、

> ラグランジュの未定乗数法（みていじょうすうほう）とは、束縛条件のもとで最適化を行うための数学（解析学）的な方法である。いくつかの変数に対して、いくつかの関数の値を固定するという束縛条件のもとで、別のある 1 つの関数の極値を求めるという問題を考える。各束縛条件に対して定数（未定乗数）を用意し、これらを係数とする線形結合を新しい関数（未定乗数も新たな変数とする）として考える。これで束縛問題を普通の極値問題として解くことができる。

メモ帳ブログ@ wiki[3]を引用すると、

D 次元の変数 $\mathbf{x} = x_1, x_2, ..., x_D$ に対し、q 個の束縛条件 $g_i(\mathbf{x}) = 0 (i = 1, 2, ...q)$ の下で関数 $f(\mathbf{x})$ を最大 (小) 化する。ラグランジュ乗数 $\lambda_i \neq 0 (i = 1, 2, ..., q)$ を導入し、ラグランジュ関数を

$$L(\mathbf{x}, \lambda) = f(x) + \sum_{i=1}^{q} \lambda_i g_i(\mathbf{x}) \tag{58}$$

とする。このとき、$\frac{\partial}{\partial \mathbf{x}} L = 0$、$\frac{\partial}{\partial \lambda} L = 0$ の連立方程式を解くことにより解の候補が得られる。

例題 1　周囲の長さが 1 の長方形のうち、面積が最大となるのはどのような長方形か?

例題 2　円柱と三角錐を組み合わせた鉛筆形の立体があり、体積が決まると、円柱の半径と円柱・円錐の高さは鉛筆形の表面積が最小になるように決まるとする。円錐部分の稜線 (母線) の長さが 3 のとき、円柱部分の半径はいくらか? ただし、底面の半径が x、母線の長さが z の円錐の表面積は πxz となり、体積は $\frac{\pi x^2}{3}\sqrt{z^2 - x^2}$ となる。[4]

Appendix B. KKT 条件

不等式制約が含まれる最適化問題を考える。q 個の等式制約 $g_1(\mathbf{x}) = 0, ..., g_q(\mathbf{x}) = 0$ と r 個の不等式制約 $h_1(\mathbf{x}) \leq 0, ..., h_r(\mathbf{x}) \leq 0$ の下で $f(\mathbf{x})$ を**最小化**する。ラグランジュ関数を

$$L(\mathbf{x}, \lambda, \mu) = f(x) + \sum_{i=1}^{q} \lambda_i g_i(\mathbf{x}) + \sum_{j=1}^{r} \mu_j h_j(\mathbf{x}) \tag{59}$$

とすると、L を $\mathbf{x}$ について最小化し、λ と μ について最大化すればよい。

さらに、最適解を $\mathbf{x}^*$ とすると、$\mathbf{x}^*$ は以下の条件を満たす:

- $\nabla_{\mathbf{x}} L(\mathbf{x}^*, \lambda, \mu) = 0$ (Stationarity)
- $g_i(\mathbf{x}^*) = 0 \; \forall i \quad \text{and} \quad h_j(\mathbf{x}^*) \leq 0 \; \forall j$ (Primal feasibility)
- $\mu_j \geq 0 \; \forall j$ (Dual feasibility)
- $\mu_j h_j(\mathbf{x}^*) = 0 \; \forall j$ (Complementary slackness: 相補条件)

[3] http://www27.atwiki.jp/nina_a/pages/31.html
[4] http://ynomura.dip.jp/archives/2009/02/lagrange.html

これらの条件を **KKT 条件** (*Karush-Kuhn-Tucker conditions*) という。

また、

$$d(\lambda, \mu) = \min_{\mathbf{x}} L(\mathbf{x}, \lambda, \mu) \tag{60}$$

とするとき、以下の問題を双対問題と呼ぶ:

$$\max_{\lambda,\mu} d(\lambda, \mu) \quad \text{s.t.} \quad \mu \geq 0 \tag{61}$$

ちなみに $f(\mathbf{x})$ を**最大化**する場合は、ラグランジュ関数を作る際に目的関数から制約条件を**引き算**するので注意

$$L(\mathbf{x}, \lambda, \mu) = f(x) + \sum_{i=1}^{q} \lambda_i g_i(\mathbf{x}) - \sum_{i=1}^{r} \mu_i h_i(\mathbf{x}) \tag{62}$$

例題 3 $f(x) = -(x_1 - 4)^2 - (x_2 - 4)^2$ を $x_1 + x_2 \leq 4$ かつ $x_1 + 3x_2 \leq 9$ の下で最大化せよ。[5]

参考文献

[1] David Chiang, Kevin Knight, and Wei Wang. 11,001 new features for statistical machine translation. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 218–226, Boulder, Colorado, June 2009. Association for Computational Linguistics.

[2] Koby Crammer, Ofer Dekel, Joseph Keshet, Shai Shalev-Shwartz, and Yoram Singer. Online passive-aggressive algorithms. *Journal of Machine Learning Research*, 2006.

[3] Koby Crammer and Yoram Singer. Ultraconservative online algorithms for multiclass problems. *Journal of Machine Learning Research*, 2003.

[4] Y. Freund and R. E. Schapire. Large margin classification using the perceptron algorithm. *Machine Learning*, 37(3):277–296, 1999.

[5] C. Gentile. The robustness of the p-norm algorithms. *Machine Learning*, 53(3), 2002.

[6] Canasai Kruengkrai, Kiyotaka Uchimoto, Jun'ichi Kazama, Yiou Wang, Kentaro Torisawa, and Hitoshi Isahara. An error-driven word-character hybrid model for joint chinese word segmentation and pos tagging. In *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics (ACL'09)*, pages ??–??, 2009.

[7] Ryan McDonald, Fernando Pereira, Seth Kulick, Scott Winters, Yang Jin, and Pete White. Simple algorithms for complex relation extraction with applications to biomedical IE. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 491–498, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics.

[8] A. B. J. Novikoff. On convergence proofs on perceptrons. In *In Proceedings of the Symposium on the Mathematical Theory of Automata*, volume XII, pages 615–622, 1962.

[9] Taro Watanabe, Jun Suzuki, Hajime Tsukada, and Hideki Isozaki. Online large-margin training for statistical machine translation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 764–773, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[5] http://www.isigas.com/KKTexample1.doc