Web文書中のユーザが知らない語を予測する読解支援システム

江原 遥
東京大学情報理工学系研究科
{ehara,ninomi,shimizu}@r.dl.itc.u-tokyo.ac.jp

二宮 崇　清水 伸幸　中川 裕志
東京大学情報基盤センター
nakagawa@dl.itc.u-tokyo.ac.jp

1　はじめに

近年，英文Web文書を読むニーズが増えている．第二言語で書かれたWeb文書を読む際には，ユーザが知らない語（ユーザ未知語）が読解を妨げる原因の一つとなる．この問題に対応するため，**語義注釈システム**が提案されてきた．語義注釈システムを用いると，ユーザは，知らない語（ユーザ未知語）に遭遇した場合，クリックまたはマウスオーバーなどの語の選択操作により，語義を表示させ，語の意味を知ることができる．図1に挙げるpop辞書[1]では，マウスオーバーしたユーザ未知語の語義をポップアップで表示している．また，ドラッグ操作で選択したユーザ未知語の語義をWeb文書中に埋め込むシステムも提案されている[2]．

語義注釈システムでは，ユーザがクリックした単語を記録することにより，ユーザのユーザ未知語のログが蓄積される．このログを，本稿では**単語クリックログ**と呼ぶ．単語クリックログは，読解の障害となるユーザ未知語のリストであるので，読解支援にとって有用な情報であると考えられる．既存の語義注釈システムでは，単語クリックログは活用されてこなかったが，単語クリックログを解析することにより，読解の障害となるユーザ未知語を予測し，予め語義を付与して読解を容易にすることが可能となると考えられる．

本稿では，単語クリックログに機械学習による識別器を導入して既存の語義注釈システムにユーザ未知語を自動的に予測し，その語に語義の注釈をつける機能を付加することを提案する[3]．本システムにユーザがログインし，本システムを通してWeb文書を閲覧した図が図2である．赤く着色された部分がユーザ未知と判別された部分であり，語義注釈が付与されている．黄色く着色された部分が既知と判別された部分である．

ユーザの語彙を予測する際には，単に予測するのではなく，既存の被験者の語彙力を測定する手法と関連があるモデルを用いることが望ましい．また，多数のユーザからリアルタイムに収集される単語クリックログは逐次的に増加するので，予測には逐次アルゴリズムを用いることが望ましい．

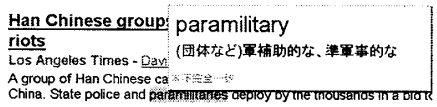

図1: pop辞書での注釈の例．

2　システム

IRT (item response theory，項目反応理論)[4] は，TOEFL (Test of English as a Foreign Language) をはじめとする既存の言語テストの設計に使用されているモデルであるので，本稿では，その最も単純なRaschモデルを改良して用いた．Raschモデルはロジスティック回帰に帰着できる．

有限集合 X の要素数を $|X|$ と書き，ユーザ $u_n \in U$ の，文書中の個々の単語 $t_n \in T$ に対する反応を $y_n \in 0,1$ とする．$y_n = 1$ のとき，ユーザ u_n は単語 t_n を知っている（既知）とし，$y_n = 0$ のとき，ユーザ u_n は単語 t_n を知らない（ユーザ未知）とする．すると，単語クリックログは $\{(u_n, t_n, y_n)|n \in \{1,\ldots,N\}\}$ と書ける．Raschモデルでは，$P(y_n = 1|u_n, t_n) = \sigma(\theta_{u_n} - d_{t_n})$ を最尤推定する．ただし，σ はロジスティックシグモイド関数で，θ_{u_n} は被験者 u_n の能力パラメータで θ_{u_n} が高いほど，被験者 u_n の正答率が増加する．また，d_{t_n} は項目 t_n の難易度パラメータで d_{t_n} が高いほど，被験者 u_n の項目 t_n に対する正答率が低下する．

単語の難しさに関する素性を導入するため，Raschモデルを次のように拡張した．重みベクトル $\boldsymbol{w}_{rasch} = (\boldsymbol{\theta}\ \boldsymbol{d})^{\mathrm{T}}$ と特徴量ベクトル $\boldsymbol{\phi}_{rasch}(u,t) = (\boldsymbol{e}_u\ \boldsymbol{e}_t)^{\mathrm{T}}$ を用いて，$P(y_n = 1|u_n, t_n) = \sigma(\theta_{u_n} - d_{t_n})$ を，$\sigma(\mathbf{w}_{rasch}^{\mathrm{T}} \boldsymbol{\phi}_{rasch}(u_n, t_n))$ と表すことができる．ただし，$\boldsymbol{\theta} = (\theta_1,\ldots,\theta_u,\ldots,\theta_{|U|})$，$\mathbf{d} = (-d_1,\ldots,-d_t,\ldots,-d_{|T|})$ であり，$\boldsymbol{e}_u$ を u 番目の要素のみ1で他は0のサイズ $|U|$ のユニットベクトル，$\boldsymbol{e}_t$ を t 番目の要素のみ1で他は0のサイズ $|T|$ のユニットベクトルである．ここで，重みベクトル $\boldsymbol{w}_{rasch}$ を $\boldsymbol{w}_{LR} = (\boldsymbol{\theta}\ \boldsymbol{d}\ \boldsymbol{w}_a)^{\mathrm{T}}$ に，特徴量ベクトル $\boldsymbol{\phi}_{rasch}$ を $\boldsymbol{\phi}_{LR}(u,t) = (\boldsymbol{e}_u\ \boldsymbol{e}_t\ \boldsymbol{\phi}_a)^{\mathrm{T}}$ と拡張することにより，$\boldsymbol{\phi}_a$ に素性を追加することが可能である．追加した素性は，Google 1-gramとSVL12000である．Google 1-gramは，約1兆ページのWeb文書中の英単語の頻度である

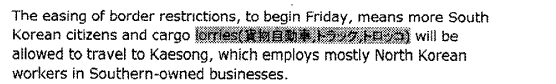

図2: 本システムでの注釈の例

[1] http://www.popjisyo.com/
[2] http://www.popin.cc/
[3] システムは，http://socialdict.appspot.com/ にて運用されている．また，英語版 Wikipedia に対しては，最初の *en.* を *enn* に置き換えることでも利用出来るように設計した．（例：http://en.wikipedia.org/wiki/Main_page に対して，http://ennwikipedia.org/wiki/Main_page）

[2]. SVL12000 は，基本的な語彙 12,000 語に対し，人手で 12 段階の難易度をつけた語彙リストである [3].

パラメータ更新の際に，データセット全体（この場合は，$\{(u_n, t_n, y_n)|n \in \{1, \ldots, N\}\}$）に対して最適化を行うパラメータ推定手法をバッチ学習法という．バッチ学習法を用いると，ユーザがクリックするたびにデータセット全体を参照し最適化を行う必要が生じるので，逐次的に増加する単語クリックログを扱う本システムでは逐次学習法を用いることが望ましい．ロジスティック回帰の最尤推定に関しては，Stochastic Gradient Descent (SGD) という逐次アルゴリズムが提案されており，本稿ではこれを用いた [1].

3 実験

実験では，辞書引きログが N_0 個蓄積されたところに，ユーザが 1 人新規にシステムを使い始め，N_1 個の単語の既知/ユーザ未知が得られたと想定し，そのユーザの**テストセット**に含まれる語のうち正しく予測できた異なり語数での語の割合を 1 人の精度として測定した．

精度評価のために，1 人 12,000 語について 5 段階の自己申告形式で回答させる方法で，被験者たち（東京大学を中心とする大学院生 16 人）の語彙力を測定し，二値化した．[4]．単語クリックログを，同じデータ構造を持つ **smart.fm**（`http://smart.fm/`）というシステムのログで代用した[5]10,526 人分のデータを **smart.fm** から取得し，継続的にシステムを利用していると思われる 675 人分のデータを実際に用いた．11,999 語を $N_1 = 600$ 語までの訓練データセット，1400 語のデベロップメントセット，9999 語のテストセットに分けた．

まず，素性追加の効果について実験を行ったところ，全ての訓練数において約 5%精度が向上した．次に，逐次アルゴリズムの効果を測定するために，他手法と比較する実験を行った．図 3，表 4 中の *LR* と *SGD* は，それぞれ，今回使用している Rasch モデルのバッチ学習法（LIBLINEAR, `http://www.csie.ntu.edu.tw/~cjlin/liblinear/` を使用），逐次学習法（SGD）である．また，*SVM (Linear)* は線形カーネルの SVM (Support Vector Machine)，*SVM (RBF)* は RBF カーネルの SVM であり，実装には LIBSVM（`http://www.csie.ntu.edu.tw/~cjlin/libsvm/`）を用いた．

LR が $N_1 = 300,600$ の時に，SVM よりも良い精度を達成していることから本システムに Rasch モデルを使用することが，予測精度の点からも妥当であることが示唆される．また，逐次学習法である SGD を用いても，バッチ学習法である LR に対して 1%〜2% ほどの精度の減少に抑えられることが示された．

[4]見たこともない，見たことがある気がする，確実に見たことはあるが意味は知らない／覚えたことがあるが意味を忘れている，意味を知っている気がする／意味が推測できる，がユーザ未知，意味を確実に知っている，が既知．

[5]**smart.fm** は，Web 上で単語を学習するためのシステムであり，学習済みの単語を学習項目から除外するために既知/ユーザ未知をユーザに問う際に辞書引きログと同じ構造を持つデータが蓄積される．

4 結論と今後の課題

語義注釈システムにおいては，クリックされた単語のログをとることにより，単語クリックログが蓄積される．既存の語義注釈システムでは，単語クリックログが活用されてこなかったが，単語クリックログを解析することにより，読解の障害となるユーザ未知語を予測し，その語に語義の注釈を付与する予測機能を持った語義注釈システムを提案した．また，英語版 Wikipedia に対しては，最初の *en.* を *enn* に置き換えることでも利用出来るように設計した．

今後の課題としては，閲覧中の Web 文書に関する特徴量を加えることなどが挙げられる．

表 1: 各手法の精度

	$N_1 = 5$	30	100	300	600
SVM (Linear)	**74.78**	**78.08**	78.88	79.20	79.27
SVM (RBF)	67.61	77.27	**79.16**	79.55	79.91
SGD	73,84	73.19	78.50	77.93	78.80
LR	73.25	77.89	79.09	**80.03**	**80.01**

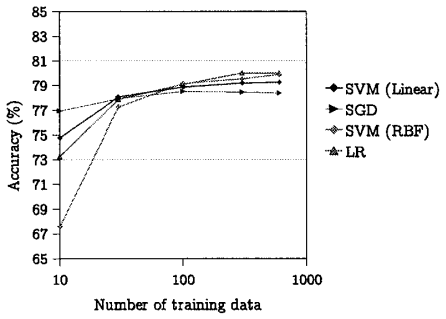

図 3: 各手法の精度

参考文献

[1] C. M. Bishop. *Pattern recognition and machine learning.* Springer, 2006.

[2] T. Brants and A. Franz. *Web 1T 5-gram Version 1.* Linguistic Data Consortium, Philadelphia, 2006.

[3] SPACE ALC Inc. Standard vocabulary list 12,000, 1998. `http://www.alc.co.jp/goi/`.

[4] 豊田秀樹. 項目反応理論 [理論編]-テストの数理-. 朝倉書店, 2005.