コーパスを用いた言語習得度の推定

坂田浩亮　　新保仁　　松本裕治

奈良先端科学技術大学院大学 情報科学研究科 自然言語処理学講座
{kosuke-s, shimbo, matsu}@is.naist.jp

概 要

言語教育において，学習者の言語習得度を知ることは教師にとって重要なことである．また，言語学習者の言語習得度を客観的に量る手段があれば, 学習者は教師がいなくても自分の言語習得度を評価できるため，学習効率の向上につながると考えられる．本発表では, 言語学習者の作文と習得度別コーパスとの類似度に基づいて, 学習者の言語習得度を推定する手法を提案する. NICT JLE コーパスを用いて行った提案手法の評価実験と，その結果について紹介する．

1　はじめに

言語教育において，学習者の言語習得度を知ることは教師にとって重要なことである． また，言語学習者の言語習得度を客観的に量る手段があれば教師がいなくても自分の作文能力を評価できるため，学習効率の向上につながると考えられる．関連研究として，機械翻訳機 (MT) の性能を評価するためによく使われる BLEU [3] という尺度がある．この評価結果は，人間が下す評価に近いと言われている．しかしながら，BLEU は MT の翻訳結果と人間の翻訳結果に使われている単語の一致度を測定して性能を評価するため，人間によって翻訳された回答例をあらかじめ準備しておかなければならないとう問題がある．本研究では，MT の性能評価の指標に使われる BLEU を参考にして，MT の翻訳結果の代わりに言語学習者の作文を，人間の翻訳結果の代わりにコーパスを用いて類似度を測定し，学習者の言語習得度を推定する手法を提案し，NICT JLE コーパス [1] を用いた提案手法の評価実験, および実験結果について紹介する．

2　関連研究：BLEU

言語習得度の推定の関連研究として，機械翻訳機 (MT) の翻訳結果と，人間の翻訳結果に使われている単語の一致度を用いて MT の性能を評価する BLEU がある．この BLEU は MT の翻訳結果と人間による回答例との類似度を用いる方法であり，これによって MT の翻訳結果の妥当性と流暢性を評価している．また，人間が下す評価に近い性能評価をするという結果が示されている. BLEU は，評価したい MT の翻訳結果と人間の翻訳結果に関して n-gram の重なりに基づいて，次式で計算する．スコアは 0.0 から 1.0 の値をとり，スコアが大きいほどよい翻訳であるとされる．

$$S_{BLEU} = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log P_n\right)$$

ただし，

$$BP = \begin{cases} 1 & \text{if } c > r \\ \exp\left(\frac{1-r}{c}\right) & \text{if } c \leqq r \end{cases}$$

c は MT の翻訳結果の文長，r は人間による回答例の文長であり，P_n は *modified n-gram precisions* と呼ばれ，

$$P_n = \frac{\sum_{C \in \Gamma} \sum_{n\text{-gram} \in C} \mathrm{Count}_{\mathrm{clip}}\left(n\text{-gram}\right)}{\sum_{C' \in \Gamma} \sum_{n\text{-gram}' \in C'} \mathrm{Count}\left(n\text{-gram}'\right)}$$

と定義される．ここで，$\mathrm{Count}\left(s\right)$ は MT の翻訳結果に含まれる n-gram s の数であり，$\mathrm{Count}_{\mathrm{clip}}\left(s\right)$ は回答例に含まれる s の最大数を t として，$\min(\mathrm{Count}\left(s\right), t)$ である．Γ は MT の翻訳結果の集合である．

w_n は，正規化するための定数であり各 n-gram に対する重みと解釈できる．一般には，$1/N$ の均一重みが使われる [3].

3　提案手法

BLEU を言語学習者のレベル推定に応用することを考える．もっとも単純なのは，MT の翻訳結果の代わりに学習者の作文を，人間による回答例の代わりに教師によって訂正された学習者の作文を用いる方法である．

しかし，この方法で推定する場合，人間によって訂正された作文の回答例をあらかじめ用意しておかなければならない問題がある．レベルを推定したい学習者が多数の場合，各々の学習者の作文の回答例を用意することは非常に手間がかかり，さらに何を書くか予測できない作文の回答例をあらかじめ用意しておくことは困難である．現実的に考えると言語学習者の作文に対する回答例が不要な手法が望ましい．

われわれが以下で提案するレベル推定手法では，作文の回答例の代わりに, 多数のレベル付けされた作文 (以下, **参照コーパス**, あるいは単に**コーパス**と呼ぶ) を用いる．参照コーパス中の作文の内容はレベル推定の対象となる作文と必ずしも一致しない．したがって，BLEU で想定している状況 (同一内容の翻訳どうしの比較) とは異なり，別内容の作文どうしを比較することになる．

以下に 2 種類の提案手法について紹介する．提案手法 1 は，n-gram の類似度に重みを与えない手法である．一方，提案手法 2 では別途訓練データを用意し，n-gram の重みを回帰分析を用いて最適化する．

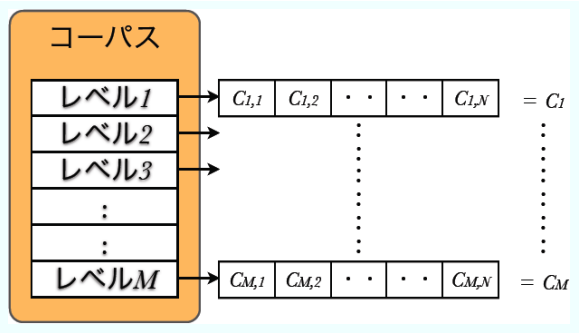

図 1: コーパスの素性ベクトル

3.1 提案手法 1

学習者による作文コーパスをレベル別に M 個に分割したもの[1]を用いて以下のステップを実行する．

1. 各レベルのコーパスから素性ベクトルを生成する
   - レベル 1,2,..,M の各コーパスから n-gram(n=1,2,..,N) を抽出し，頻度ベクトルを計算する．
   - レベル m のコーパスの素性ベクトルを C_m とすると，C_m=($C_{m,1}$,$C_{m,2}$,...,$C_{m,N}$) と書ける．
     ここで，$C_{m,n}$ は単語 n-gram の頻度ベクトルである．(図 1 を参照)
2. レベルを推定したい作文の素性ベクトル V を以下のように計算する
   - 作文の素性ベクトルを V とすると，V=(V_1,V_2,...,V_N) であり，ここで V_n は単語 n-gram ベクトルである．
3. 学習者の素性ベクトルに対して全てのコーパスとのコサインをとる

$$x_{m,n} = \cos\angle(V_n, C_{m,n})$$

   これらを並べて V のレベル別類似度ベクトル x とする．(Algorithm 1 を参照)
4. レベル推定を行う
   - 1〜N-gram のコサイン類似度の和をそのレベルのスコアとし，スコアが一番高いコーパスレベル y を推定値とする．すなわち，

$$y = \arg\max_m \sum_{n=1}^{N} x_{m,n}$$

3.2 提案手法 2

提案手法 1 では，コサイン類似度が最大のレベルをその作文レベルの推定値とした．提案手法 2 では，別途訓練データを用意し，回帰によって各 n-gram の重みを最適化する．この際，各レベルの n-gram のコサイン類似度に重み $w_{m,n}$ を与えて回帰式 $y = w \cdot x + b$ を求める．学習者レベルが既知の作文 K 個と，レベル別コーパスを用いて以下のステップを実行する．

[1]各レベルのコーパスには複数の作文が含まれるが，これらをまとめて一個として扱い n-gram を抽出する

Algorithm 1 *ComputeSimilarityVector:* レベル別類似度ベクトルを計算

入力: 作文の素性ベクトル $V = (V_1, \ldots, V_N)$)
出力: V のレベル別類似度ベクトル x

```
for m ← 1...M do { 各レベル m について }
  for n ← 1...N do { 各 n-gram について }
    x_{m,n} ← cos∠(V_n, C_{m,n})
  end for
end for
x ← (x_{1,1}, ..., x_{M,N})
return x
```

1. 各レベルのコーパスから素性ベクトルを生成する
   - レベル 1,2,..,M の各コーパスから n-gram(n=1,2,..,N) を抽出し，頻度ベクトルを計算する．
   - レベル m のコーパスの素性ベクトルを C_m とすると，C_m=($C_{m,1}$,$C_{m,2}$,...,$C_{m,N}$) であり，$C_{m,n}$ は単語 n-gram のベクトルである．
2. レベルが既知の訓練データから素性ベクトルを生成する
   - K 個の作文各々について n-gram(n=1,2,..,N) の統計をとる．
   - k 番目の作文の素性ベクトルを $V^{(k)}$ とすると，$V^{(k)}$=($V_1^{(k)}$,$V_2^{(k)}$,...,$V_N^{(k)}$) であり，ここで $V_n^{(k)}$ は k 番目の作文の単語 n-gram ベクトルである．図 2 を参照．
3. 訓練データの素性ベクトルに対して全てのコーパスとのコサイン類似度をとる

$$x_{m,n}^{(k)} = \cos\angle\left(V_n^{(k)}, C_{m,n}\right)$$

   これらを並べて各訓練データの類似度ベクトル $x^{(k)}$ を得る．(Algorithm2 を参照)
4. 回帰分析による重み付けを行う
   - 訓練データ $\{(x^{(k)}, y^{(k)})\}_{k=1}^{K}$ の近似式

$$y = w \cdot x + b \tag{1}$$

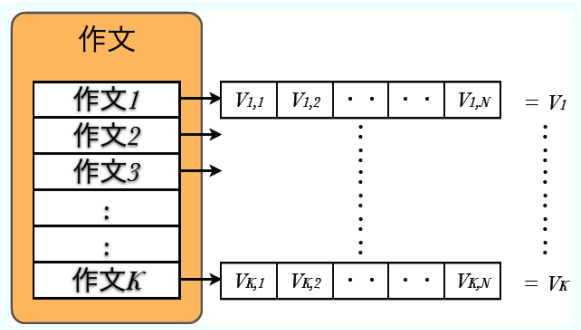

図 2: 作文の素性ベクトル

Algorithm 2 全訓練データに対してレベル別類似度ベクトルを計算

入力: 作文の集合 $\{V^{(k)}\}_{k=1}^{K}$
出力: レベル別類似度ベクトル集合 $\{x^{(k)}\}_{k=1}^{K}$

for $k \leftarrow 1 \ldots K$ **do** { 各作文 k について }
  $x^{(k)} \leftarrow ComputeSimilarityVector(v^{(k)})$
end for
return $\{x^{(k)}\}_{k=1}^{K}$

を回帰によって求める．ここで，

$$w = (w_{1,1}, w_{1,2}, ..., w_{M,N})$$

はレベル別類似度ベクトルの各要素に対する重みである．

5. レベル推定を行う
   - 求めた回帰式 (1) に作文の素性ベクトル x を代入し，y を計算する．y を四捨五入した値を推定レベルとする．

3.3 素性

BLEU では MT の翻訳結果に出てくる単語のみを用いて人間による回答例との類似度を測定しているが，使われる単語は書かれている文の話題に依存しやすいため偏りが生じる可能性がある．人間による回答例の代わりに使われるコーパスは BLEU のように評価したい文の内容と必ずしも一致しないため，単語のみの素性は適切ではないこともあり得る．その場合には，単語の代わりに品詞，または単語と品詞の両方を素性として使うことが可能である．

3.4 類似度尺度

今回, 上記の提案手法の説明 (および後述の実験) においてはコサイン類似度を用いたが, レベル別類似度ベクトルの要素として任意の類似度尺度を用いることが可能である．BLEU で用いられている modified n-gram precisions などをコサインのかわりに用いる, あるいは併用することも可能である．

4 実験

提案手法の有効性および素性ベクトルの有効性を検証する．

4.1 実験データ

本実験で使用するデータとして，NICT JLE コーパスを用いる．NICT JLE コーパスは，英語話者ではなく日本人英語学習者の会話を書き起こした言語データベースである．このコーパスの特徴として，英語話者とのインタンビュー形式で学習者が質問に答えており，話し言葉のため，フィラーや言い直し等もそのまま記載されている[2]．また，最大の特徴は SST (Standard Speaking Test) により発話能力を 9 レベルに分けてあり，このレベルは文法，語法，発音，流暢性を客観的に評価している点である．

4.2 実験設定

NICT JLE コーパスに対して 2 通りのレベル分けを行って実験をする．ひとつは，NICT JLE コーパスにもともと与えられた 9 段階のレベルに分ける．もうひとつは，NICT JLE コーパスの 9 レベルを 3 等分し，3 段階に分けて[3]実験に用いる．素性は，単語の n-gram はコーパスの会話によって使われる単語に偏りが生じてしまうため，単語の n-gram だけではなく，品詞の n-gram 及び単語の n-gram と品詞の n-gram を組み合わせたものを素性として用いる．なお，本実験では $N = 5$, すなわち 1〜5-gram を素性に使う．

提案手法 2 の回帰分析には，Support Vector Regression [4][4]を用いる．

4.3 テストデータと訓練データ

評価用に用いるテストデータは，NICT JLE コーパスからランダムに 30 個の作文を取り出したものを使う．この際，レベルに偏りが出来ないように 3 段階にコーパスを分けた後，各々のレベルから 10 人分ずつ取り出す．提案手法 2 では回帰分析で最適な重み付けを設定するために別途訓練データを用意する．この訓練データを用いて学習し，テストデータを用いて評価を行う．訓練データは，テストデータと同様に NICT JLE コーパスからランダムに 30 個の作文を取り出したものを使う．参照コーパスは NICT JLE コーパスからテストデータと訓練データを引いた残りを用いる．本実験ではこれらを 5 セット用意して評価を行う．

4.4 実験結果

提案手法および 2 つの提案手法の実験結果を図 3 および図 4 に示す．実験結果は 5 回の平均値である．

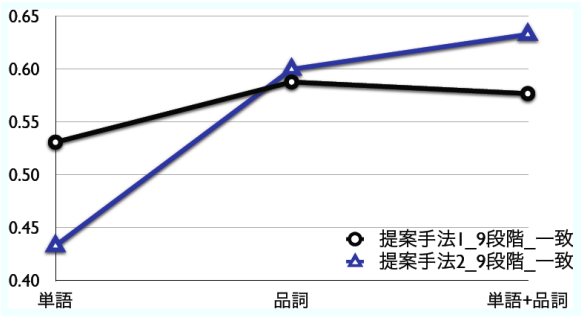

図 3: 実験結果 1 (9 段階)

図 3 は提案手法 1 と提案手法 2 による 9 段階に分割したコーパスレベルを推定したときの精度を示している．一方図 4 は提案手法 1 と提案手法 2 による 3 段階に分割したコーパ

[2]本実験では前処理によってフィラー等は取り除いた

[3]初級 (レベル 1〜3)，中級 (レベル 4〜6) および上級 (レベル 7〜9) の 3 段階

[4]実験には SVMlight[2] を利用した．

表 1: NICT JLE コーパスの語彙リスト ([1] より)

レベル	全体	1	2	3	4	5	6	7	8	9
被験者数	1,201	3	35	222	482	236	130	58	25	10
平均発話数	1,115	338	475	790	1,060	1,298	1,412	1,505	1,715	1,632
文の平均長	7.65	3.09	4.04	5.9	7.44	8.44	9	9.14	9.25	9.42

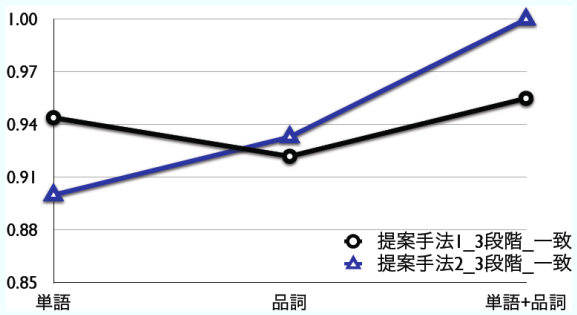

図 4: 実験結果 2 (3 段階)

スレベルを推定したときの精度を示している．2 つの実験結果より単語のみを素性ベクトルに用いた場合，提案手法 1 の方が良い精度であったが，品詞を素性ベクトルに加えることによって提案手法 2 が提案手法 1 よりも推定精度が上回った．このことより，素性ベクトルに品詞情報を用いるのは有効である事が分かる．また，図 4 より，提案手法 2 による 3 レベル推定ではほぼ確実に推定することができた．以上 2 つの実験結果より，単語と品詞を素性ベクトルにして回帰分析による重みの最適化は有効な手法であることが確認できた．

5 おわりに

学習者が書いた作文から言語習得度を推定する手法を提案した．提案手法では作文内容に依存した回答例を必要とせず，レベル付けされたコーパスを用いて最も近くなったコーパスのレベルを推定値とする．各レベルに対して 1〜5-gram のコサイン類似度を素性に使い，回帰分析により最適な重みを付けて推定を行う．実験データに NICT JLE コーパスを採用したところ，約 65%の精度で習得度を推定することができた．

NICT JLE コーパスのようにレベル付けされたコーパスはほとんどない上にサンプル数が少ないため，回帰分析を行うための訓練データにサンプルを多くとることが出来ない．さらに，中級レベルの学習者が極端に多く，初級および上級レベルの学習者が少ない問題もある．今後は，レベル付けされていないコーパスを使って学習者のレベルを推定する手法を考えたい．

参考文献

[1] 和泉絵美, 内元清貴, 井佐原均. 日本人 1200 人の英語スピーキングコーパス. アルク, 2004.

[2] Thorsten Joachims. Making large-scale SVM learning practical. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods: Support Vector Learning*. MIT Press, 1999.

[3] Kishore Papineni, Salim Roukos, Todd Ward, and Wei Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th ACL*, pp. 311–318, 2002.

[4] Vladimir Vapnik. *The Nature of Statistical Learning Theory*. Springer, New York, second edition, 1998.