

同時通訳のための述語動詞予測手法の検討

竹林佑斗[†], 荒瀬由紀[‡],

[†] 大阪大学工学部電子情報工学科, [‡] 大阪大学大学院情報科学研究科

{takebayashi.yuto, arase}@ist.osaka-u.ac.jp

1 はじめに

グローバル化に伴いビジネスや医療の現場, 国際会議など異言語間コミュニケーションの機会が増加し, 自分の母語とは異なる外国語でリアルタイムに対話することの重要性が高まっている. そこで利用者にとってコスト面で有利となる機械的な同時通訳を実現するシステムが研究されている [2, 5].

同時通訳とは, 入力の記事が完結する前に目的言語の部分的な翻訳結果を訳出するものである. 翻訳時のタイムラグは円滑なコミュニケーションの大きな障害であるため, 素早く正確な部分翻訳を生成することが必要とされている. 特に日本語のような SOV 型言語から英語のような SVO 型言語への同時通訳においては訳出開始までのタイムラグが大きな問題となる. これは図 1 に示すように, SOV 型言語では述語動詞が文の最後に現れるため, SVO 型言語への翻訳の途中で述語動詞が現れるまで待つ必要があるためである. そこで述語動詞を早期に予測できれば目的言語の訳出時間を早められタイムラグを改善できる. 図 1 の例では「私は毎朝テレビを見ながら朝食に」までが入力された時点で述語動詞「食べる」を予測できれば, 訳出までのタイムラグを短縮できる.

本稿では述語動詞予測を多クラス分類問題として定式化し, 予測性能に影響を与える要因を分析する. 文頭から n 単語を入力として用い, Bag-of-words により特徴ベクトルを作成し, 分類器を学習させる. 実験の結果, 重文・複文では動詞予測が困難となること, また予測においては動詞の語義や階層関係を考慮すべきことが明らかとなった.

2 関連研究

SOV 型言語から SVO 型言語への同時通訳における翻訳遅れを解消する手法として, 強化学習を用いた翻訳手法が Grissom II ら [2] によって提案されている.

私は 毎朝 テレビを見ながら 朝食に フルーツとヨーグルトを 食べます。	
I every morning TV watching breakfast fruits and yogurt	eat
I (. 待ち時間)	eat fruits and yogurt ...
私は 毎朝 テレビを見ながら 朝食に フルーツとヨーグルトを 食べます。	
I every morning TV watching breakfast fruits and yogurt	eat 予測
I (. 待ち時間)	eat fruits and yogurt for breakfast while watching ...

図 1: 日英同時通訳における述語動詞予測

ドイツ語から英語への翻訳において単語が断続的に入力される中で, いつ入力言語の述語動詞を予測し, いつ予測された述語動詞を用いて部分翻訳を訳出するか, タイミングを学習させることで, 翻訳精度を保ったまま訳出時間を早めている. 実験結果から SOV 型言語から SVO 型言語の同時通訳においては正確な述語動詞予測が重要であることが示されている.

また同時通訳における入力のように不完全な文において翻訳単位に隣接する文法要素を予測し, 得られた結果を文の一部として扱い翻訳する手法が, 小田ら [5] によって提案されている. さらに予測した文法要素を用いて後続の入力を待つべきかどうかを判断することで翻訳の質を向上できることが示されている.

これらの研究では最終的な翻訳性能のみ評価しており, 述語動詞予測の精度自体は検証されておらず, 述語動詞予測に影響を与える要因については詳細に分析されていなかった.

3 多クラス分類による動詞予測

提案手法では入力文の初めから n 単語を Bag-of-words を用いてベクトル化したものを特徴量とし, 予測する述語動詞のクラスに分類することで述語動詞を予測する. 分類器としては多様な分類問題において高い性能を示している Support Vector Machine (SVM) [1] を用いる.

ここで述語動詞とは, 日本語係り受け解析器の

表 1: コーパスの例

述語	文章
褒める	お子さんが少しでも上手に出来るた大袈裟だくらい褒めてあげてください。
公開	小淵内閣時代のホームページがいまも公開するれるています。

CaboCha[4]を用いて係り受け解析を行った結果、ルートの文節に出てくる最初の「動詞」、または最初の「サ変接続+する(できる)」のサ変接続の名詞と定義する。文章を日本語形態素解析エンジンのMeCab[3]を用いて形態素に分割した後、各形態素の原形を用いる。述語動詞とその文章の例を表1に示す。表1の2行目“公開”がサ変接続の名詞を述語動詞としたものである。本来日本語では名詞扱いであるが、翻訳すると目的語での動詞に相当するため、サ変接続も述語動詞と考えることとした。

Bag-of-wordsの辞書には、学習コーパス中の頻出20,000単語を用いる。ただしトップ10件の頻出単語については、あらゆる文章で現れる可能性が高く特徴量としては有効でないと考えられるため、ストップワードとして除去する。辞書に含まれない単語は全て未知語(unlk)とする。その結果コーパスにおけるストップワード、辞書内の単語、及び未知語が占める割合はそれぞれ30.4%、67.1%及び2.5%となった。

生成した特徴ベクトルは非常に疎であり、分類精度を下げる要因となるため、主成分分析を用いて次元削減をし低次元のベクトルを生成する。

4 実験

4.1 実験設定

分類にはpythonのモジュールであるsklearnに実装されているSVMを使用し多クラス分類を行なった。カーネル関数はRBFカーネル、線形カーネル、多項式カーネル及びシグモイドカーネルに対応するrbf, linear, poly及びsigmoidの4つを比較した。コストパラメータ C は1.0に、rbf, poly及びsigmoidをカーネル関数とした時のカーネル係数である γ は、 $\gamma = 1/N$ (N は特徴ベクトルの次元数)となるようにした。特徴ベクトルの次元削減にはpythonのモジュールであるsklearnに実装されているものを用いた。

表 2: 各カーネル関数と予測動詞数の影響

予測動詞数	カーネル関数			
	rbf	linear	poly	sigmoid
2	66.37	67.79	49.34	49.33
10	19.85	22.40	8.978	8.966
50	2.356	5.872	1.436	1.436
100	0.578	2.756	0.530	0.532

本稿の実験ではYahoo!知恵袋データ(第2版)*1の回答約20,000,000文から述語動詞を抽出し、その文章と述語動詞の対を正解データとして用いる。Yahoo!知恵袋データ(第2版)はWikipediaほど文語的ではなく、Twitterほど崩れた口語でもないため、使用目的である同時通訳に近いと考えた。また様々なドメインの文章を使用できるという点もYahoo!知恵袋データ(第2版)の利点である。

実験では予測(分類)対象とする述語動詞は頻出100動詞の中からランダムに選択し、学習データの数が合計10,000文となるようにそれぞれの述語動詞を持つ文を抽出した。

予測対象の述語動詞を決定した後、それぞれを含む文が均等になるようにコーパスからサンプルした。例えば10種類の述語動詞を予測する場合には、各述語動詞を含む文をそれぞれ1,000文ずつサンプルして用いる。そのため述語動詞の選び方によって正答率が大きく変動すると考えられる。そこですべての条件において5回ずつ試行しその平均をとった値を用いる。

実験では予測する述語動詞数および入力する単語数 n を変化させ、それらの動詞予測に対する影響を調査する。評価には5分割交差検定を用いた。評価指標として、以下の式で計算される正答率を用いる。

$$\text{正答率} = \frac{\text{予測に成功した文}}{\text{入力文数}} \quad (1)$$

4.2 実験結果

4.2.1 ベクトルの次元数、カーネル関数の影響

次元削減後の特徴ベクトルの次元数を100に固定し、予測動詞数を2から100まで変化させ、4つのカーネル関数を用いた際の分類結果を表2に示す。各予測動詞数で正答率が最も高いものを太字で示した。表2より予測動詞数が多くなるほどlinearカーネルでの分類精度が相対的に高くなっていることがわかる。これは主

*1http://www.nii.ac.jp/dsc/idr/yahoo/chiebkr2/Y_chiebukuro.html

表 3: linear カーネルにおける削減次元数と予測動詞数

削減次元	予測動詞数			
	2	10	50	100
2	54.34	11.28	2.094	0.952
50	64.28	19.11	5.100	2.434
100	67.79	22.40	5.872	2.756
500	72.06	25.81	6.950	3.982
2000	69.84	26.97	7.782	4.360

表 4: linear カーネルにおける入力単語数と予測動詞数

入力単語数	予測動詞数			
	2	10	50	100
10	70.41	26.93	6.618	3.444
15	71.87	27.32	7.028	3.596
20	72.19	27.44	7.648	3.916
all	78.15	44.24	14.10	7.858

成分分析による特徴ベクトルの次元削減が効果的に働いたためと考えられる。

次に分類性能が高かった linear カーネルを用い、予測動詞数を 2 から 100 とした際、特徴ベクトルの次元数を 2 から 2,000 まで変化させた結果を表 3 に示す。各予測動詞数で正答率が最も高いものを太字で示した。予測動詞数が多くなるにつれ次元数が大きい方が予測性能が向上しており、多数の述語動詞を予測するには特徴量を保持する必要があることが分かる。

4.2.2 入力単語数の影響

表 4 に linear カーネルを用い、特徴ベクトルの次元数を 200 に固定した*2時の、入力単語数 n と予測動詞数の関係を示す。4 行目の“all”は文頭から述語動詞が現れるまでの単語を全て入力した時の結果であり、入力単語数の平均は 22.6 であった。どの予測動詞数においても入力単語数が多いほど正答率が高くなっている。これは入力単語数が多いと、述語動詞を予測する上で特徴的な単語や単語の組み合わせが出現する可能性が高く、入力単語数が少ない時に比べ精度が向上するためと考えられる。文頭から述語動詞の直前までの単語全てを用いた場合、10 単語のみ入力する場合に比べて予測精度が 1.1~2.3 倍となっている。直前までの

*2表 3 の結果では 2000 次元のベクトルを用いた場合が最も高い分類精度となったが、学習時間短縮のため本実験では 200 次元のベクトルを用いた。

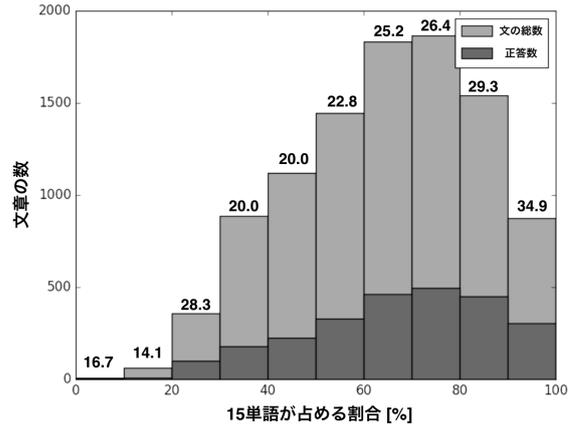


図 2: 前単語に対する入力単語の割合

全ての単語を特徴として考慮することを考えると、1 文のみを入力とし、Bag-of-words を特徴量として用いる場合の予測性能の上限と考えられる。しかし 50 種類の動詞の予測では正答率は 14.1% に留まっており、動詞予測問題が単純なアプローチでは解決するのが難しいことを示唆している。

次に、文長に対する入力単語の占める割合が正答率に与える影響を調査する。Linear カーネルを用い、入力単語数 15、予測動詞数 10、特徴ベクトルの次元数 200 という条件で動詞予測を行なった結果について、文頭から述語動詞の直前の単語までの単語数に対し、入力する 15 単語が占める割合ごとに、文の総数と予測に成功した文数のヒストグラムを図 2 に示す。グラフ中の数字は各割合別での正答率 [%] を表している。グラフから入力単語が占める割合が 0~20% までの文章は述語動詞予測の正答率が非常に低いことがわかる。これらの文章は述語動詞までの単語数が 75 単語以上の長い文であり、重文・複文である可能性が高い。実際に 67 文をサンプルして調査したところ、75 単語未満の文では重文・複文の割合が約 76% なのに対し、75 単語以上の文では 100% が重文・複文であった。重文・複文では各節で独立して述語動詞を取るため、前半の節内の単語から後半の節の動詞を予測するのは困難である。一方で入力単語数の占める割合が 80% 以上では正答率が高くなっている。これは述語動詞までの単語数が 19 個未満の短い文である。これは短文では述語動詞に影響する分類に有効な単語の割合が、長文に比べ相対的に高くなるためと考えられる。表 5 は実験データからサンプルした例である。括弧内が入力単語、予測すべき述語動詞を太字で示す。短文では“不要な”や“メール”など予測に有効な単語が多く含まれているが、長文では“削除”には関係のない単語が多く、予

表 5: サンプルデータの例

文章
[この時もし不要なメールを発見したら、その場でメールボックス]から削除してください。
[また、システムの復元は Windows 内に記憶させる容量がととも]多いので、HDD の残り容量が少なくなってくれば、当然復元ポイントも削除されます。

		分類されたクラス									
		とる	答える	つける	表示	待つ	送る	削除	消える	すぎる	言える
本語のクラス	とる	240	56	110	59	71	84	14	91	156	119
	答える	93	368	64	44	52	36	66	54	110	113
	つける	202	65	182	61	67	72	53	69	111	118
	表示	96	31	51	431	32	54	91	93	61	60
	待つ	142	55	100	64	163	114	60	98	101	103
	送る	142	47	80	66	125	267	52	72	72	77
	削除	51	72	53	104	67	86	368	87	46	66
	消える	159	51	69	121	65	52	92	172	127	92
	すぎる	187	79	73	44	55	34	35	70	263	160
	言える	138	81	96	43	57	40	23	62	170	290

図 3: 10 クラス分類での混同行列

測が難しいことが分かる。

4.2.3 動詞の特性による影響

図 2 と同様の条件で分類した結果について、混同行列を図 3 に示す (数値は実数である)。対角線上に色が濃い部分については予測精度が高かった動詞であり、“答える”や“表示”、“削除”などの述語動詞が正確に分類されていることがわかる。一方“つける”や“待つ”、“消える”などは他の述語動詞として予測されることが多く、精度が低い。また多くの述語動詞が“とる”に分類されている。この理由の一つに、“とる”が多語義であるということが挙げられる。日本語ワードネット^{*3}で“とる”を検索すると語義が 16 個あるが、誤分類されにくい“削除”については 9 個のみである。“とる”の語義には「取り除く。除去する。」という語義があり、“消える”と意味が近い。そのため“消える”が“とる”に誤分類されたと考えられる。一方で“つける”と“とる”は反意関係にあるが、反意語は出現するコンテキストが類似する傾向にあるため、誤分類の要因となったと考えられる。したがって、すべての動詞を均一に予測するのではなく、動詞の語義やその

^{*3}<http://nlpwww.nict.go.jp/wn-ja/index.en.html>

階層関係を用い、段階的に分類するアプローチが有効と考えられる。

5 まとめ

本稿では同時通訳のための動詞予測問題を文頭からの数単語を用いた動詞の分類問題として定式化し、予測性能に影響を与える要因を分析した。

実験結果より、文頭から述語動詞までの単語数に入力する単語数が占める割合が大きいほど分類性能が高いことを示し、長文となる傾向のある重文・複文では予測が困難になることが明らかになった。さらにすべての動詞を均一に扱うのではなく、語義や階層関係を考慮し段階的な予測を行うアプローチが有効と考えられることを示した。

今後は、本稿の実験で判明した上記課題への取り組みと、リカレントニューラルネットワークなどを用いた単語の順序を考慮するモデルを利用した予測に取り組む。また、実際に動詞予測機構を組み込んだ日英翻訳システムを構築し、翻訳の性能についても調査する予定である。

謝辞

本研究は大阪大学と NTT コミュニケーション科学基礎研究所の共同研究の一環で行われた。また本稿執筆にあたり有益なご助言をいただいた大阪大学大学院情報科学研究科の高田祥平氏に感謝する。

参考文献

- [1] Corinna Cortes and Vladimir Vapnik. Support-Vector Networks. In *Machine Learning*, pp. 273–297, 1995.
- [2] Alvin Grissom II, He He, Jordan Boyd-Graber, John Morgan, and Hal Daumé III. Don't Until the Final Verb Wait: Reinforcement Learning for Simultaneous Machine Translation. In *In Proc. of EMNLP 2014*, pp. 1342–1352, (Oct. 2014).
- [3] Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. Applying Conditional Random Fields to Japanese Morphological Analysis. In *In Proc. of EMNLP 2014*, pp. 230–237, (Oct. 2014).
- [4] 工藤拓, 松本裕治. チャンキングの段階適用による日本語係り受け解析. Vol. 43, No. 6, pp. 1834–1842, 2002.
- [5] 小田悠介, Graham Neubig, Sakriani Sakti, 戸田智基, 中村哲. 不完全な文の構文解析に基づく同時音声翻訳. 言語処理学会第 21 回年次大会 (NLP2015).