名詞の主要語義の推定と語義識別への応用

江口晃 ○
茨城大学工学部情報工学科

新納浩幸
茨城大学工学部情報工学科

佐々木稔
茨城大学工学部情報工学科

1　はじめに

本論文では多義語の名詞に対して、その主要語義を推定することで、文脈の適切なベクトル化を目指した。語義識別問題でその効果を調べた。

語義識別問題に見られるように、自然言語処理では文脈をベクトル化することが頻繁に行われる。その際、利用する素性は、文脈中の単語だけではなく、単語の語義コードも利用するのが一般的である。ただしその単語が複数の語義コードを持つ、つまり多義語であった場合、その複数の語義コードを全て利用することが通常行われている。この方法では、明らかに文脈を正確にベクトル化していない。

ここでは名詞の語義のコードとして日本語 WordNet[1] を利用する。日本語ワードネットから語義を検索すると、通常複数の語義が提示されるが、それら語義から主要な上位 3 つを推定する。文脈をベクトル化する際に、多義語の名詞に対しては、推定した主要語義のコードだけを素性に利用することにする。

実験では Senseval2 の辞書タスク [3] の動詞 50 単語に対する語義識別を行った。以下の 3 つのケースについて正解率を調べた。(1) 語義コードを利用しない、(2) すべての語義コードを利用する、(3) 主要語義のコードだけを利用する。正解率は (2)、(3)、(1) の順に高かった。主要語義の推定の精度が悪い、主要語義の推定を名詞だけしていることが問題だと考えられる。今後は辞書を併用して推定の精度を高めたい。

2　文脈のベクトル化の問題点

名詞「記録」の語義識別問題を例に文脈のベクトル化の問題点を述べる。「記録」には以下の 2 つの語義がある。

語義 1 のちのちに伝える必要から、事実を書きしるすこと

語義 2 競技などの成績・結果

そして以下の (a) の文の「記録」は (語義 1) の意味であり、(b) の文の「記録」は (語義 2) の意味である。

(a) これがこの地方の儀式の記録です。

(b) このタイムが公式記録です。

(a) の「記録」の文脈に「儀式」がある。「儀式」の日本語 WordNet での語義コードは以下の 6 つである。

07450842-n、01027379-n、07447261-n、01027859-n、01204055-n、04911420-n

また (b) の「記録」の文脈に「公式」がある。「公式」の日本語 WordNet での語義コードは以下の 5 つである。

01204055-n、04911420-n、06731802-n、05846932-n、00186491-r

(a) の「記録」の文脈と (b) の「記録」の文脈は異なるはずなので、共通の素性が現れないのが理想であるが、現実には “01204055-n” と “04911420-n” の 2 つが一致してしまう。結果として (a) の「記録」に対するベクトルと、(b) の「記録」に対するベクトルとに類似度が現れ、語義識別が誤ることがある。

文脈をベクトル化する際に語義コードを含めれば、異なる単語間に対しても意味が似ていれば類似度を与えることができるために、より適切なベクトル化が可能になる。しかしその反面、上記の例のように文脈内に多義語が存在すると、その多義語の利用されていない別の語義が、もう一方の文脈との類似度を誤って高める危険性もある。

上記問題を解決するには、文脈中に多義語が存在した場合、その多義語の語義を識別できることが理想である。ただし現実には多義語の語義を正確に識別することは困難であるし、本来、多義語の語義を識別するために文脈のベクトル化が必要なので、文脈のベクトル化の際に多義語の語義を識別することはできない。

本論文ではこの問題を解決するために名詞の主要語義を推定し、多義語の名詞に対しては、その主要語義だけを利用することを提案する。例えば上記の例では、「儀式」の主要語義は以下の 3 つであり、「儀式」と「公式」に類似度が生じることを避けることができる。

07450842-n、01027379-n、01027859-n

3 名詞主要語義の推定

ここでは名詞の主要語義を推定するためにコーパスを利用する。コーパスとしては BCCWJ コーパス [2] の「白書」を利用した。コーパス中の名詞 a の頻度を $f(a)$ とし、名詞 a の語義コードの集合を $W(a)$ とする。また関数 $in(x, S)$ は集合 S の中に要素 x が存在すれば 1 を、存在しなければ 0 を返す関数とする。

コーパスを形態素解析することで、各語義コード m_k に対して、以下の値を求める。

$$\sum_a in(m_k, W(a)) \cdot f(a)$$

この値の大きい順に全体の語義コードをソートした表を作成し、これを語義コード順位表と名付ける。

実際に多義語の名詞に対して主要語義を選出するには、その名詞の各語義コードが語義コード順位表で何番目の語義であるかを調べ、その順位の高いものから指定された個数だけ語義コードを選出することで行える。

4 語義識別問題への応用

語義識別問題に対して帰納学習手法の Naive Bayes を用いる。

対象単語の文脈 x が素性のリストとして、以下のように表現されたとする。

$$x = (f_1, f_2, \cdots, f_n)$$

対象単語の語義の集合を $C = \{c_1, c_2, \cdots, c_m\}$ とおく。語義識別問題はベイズの定理を利用して、以下を求めれば良い。

$$arg \max_{c \in C} P(c)P(x|c) \qquad (1)$$

問題は、$P(x|c)$ の推定だが、Naive Bayes のモデルは、この推定に以下の仮定を導入する。

$$P(x|c) = \prod_{i=1}^{n} P(f_i|c)$$

$P(c)$ や $P(f_i|c)$ を訓練データから推定することで、式 1 を求めることができる。

利用する素性としては、以下の 5 つを設定した。

e1	直前の単語
e2	直後の単語
e3	前方の内容語 2 つまで
e4	後方の内容語 2 つまで
e5	e3 と e4 の語義コード

例えば、語義識別の対象単語を「描く」として、以下の文を考える（形態素解析され各単語は原型に戻されているとする）。

過酷/な/世界/を/描く/ドラマ/。

この場合、「描く」の直前、直後の単語は「を」と「ドラマ」なので、‘e1=を’, ‘e2=ドラマ’ となる。次に、「描く」の前方の内容語は「過酷」、「世界」なので、ここから「描く」に近い順に 2 つ取り、‘e3=過酷’, ‘e3=世界’ が作られる．またここでは句読点も内容語に設定しているので、「描く」の後方の内容語は「ドラマ」と「。」となり、‘e4=ドラマ’、‘e4=。’ が作られる。次に「過酷」「世界」「ドラマ」の各語義コードを WordNet から調べると以下を得る。

- 過酷：01803583-a, 01041481-a,01507402-a
- 世界：09270894-n, 07966140-n, 14514805-n, 09480809-n, 05809878-n, 07965937-n
- ドラマ：07290278-n, 06376154-n, 07007945-n

結果として，上記の例文に対しては以下の 18 個の素性が得られる．

e1=を, e2=ドラマ, e3=過酷, e3=世界, e4=ドラマ, e4=。, e5=01803583-a, e5=01041481-a, e5=01507402-a, e5=09270894-n, e5=07966140-n, e5=14514805-n, e5=09480809-n, e5=05809878-n, e5=07965937-n, e5=07290278-n, e5=06376154-n, e5=07007945-n

本手法の名詞の主要語義の上位 3 つだけを利用する場合、

- 過酷：01803583-a, 01041481-a,01507402-a
- 世界：09270894-n, 14514805-n, 07965937-n
- ドラマ：07290278-n, 06376154-n, 07007945-n

となり、素性は以下の 15 個となる。

e1=を, e2=ドラマ, e3=過酷, e3=世界, e4=ドラマ, e4=。, e5=01803583-a, e5=01041481-a, e5=01507402-a, e5=09270894-n, e5=14514805-n, e5=07965937-n, e5=07290278-n, e5=06376154-n, e5=07007945-n

5 実験

推定した主要語義だけを用いる効果を調べるために、Naive Bayes を用いて、Senseval2 の辞書タスクの動詞 50 単語に対する語義識別を行った。このタスクは各動詞に対して 100 問のテストデータがあり、その正解率を調べた。結果を表 1 に示す。

表 1 の “語義なし” の列は素性に語義コードを利用しなかった結果である。前章で言えば e5 の素性を使わなかった結果である。“全語義” の列は多義語に対してすべての語義コードを利用した結果である。“主要語義” の列が本手法に対応し、多義語に対しては主要語義だけを利用した結果である。またここでは主要語義として語義コード順位表上の上位 3 つを使うことにした。

表 1 から、語義を利用することで識別の精度が向上するが、主要語義だけを使うよりも全語義を使った方が識別の精度が高いことがわかる。ただし以下の 11 単語に関しては、主要語義だけを使う方が全語義を使うよりも精度が高かった。本手法は効果がある面もあったが、総合的にみると全語義を使う方がよかったと言える。

ataeru	0.63	→	0.67
umareru	0.66	→	0.67
kau	0.84	→	0.85
kiku	0.62	→	0.64
kimeru	0.92	→	0.93
kuru	0.83	→	0.84
tsukau	0.96	→	0.97
tsutaeru	0.77	→	0.78
tou	0.63	→	0.65
nerau	0.98	→	0.99
motomeru	0.86	→	0.87

6 考察

実験では主要語義の個数を 3 つに設定した。この部分で最適な数を推定すれば、全語義を使ったものよりも良い精度が得られる可能性はあるが、この数がある程度大きいと、結果は全語義を使ったものとほぼ同等になるし、0 にすれば結果は語義を使わないものと同等になるので、最適な数を使っても前章の実験の結論に変化はないと予想する。

上記の実験結果で deru の結果は、語義を利用することで精度が下がっている。deru の 100 問のテストの中で “語義なし” が正解し “全語義” が誤ったものは 7 問、逆に “全語義” が正解し “語義ない” が誤ったものは 2 問である。総合して “全語義” の方が 0.05 精度が低い。“全語義” が誤った上記 7 問を概観すると e5 の素性、つまり語義の素性が非常に多く、本論文で示した文脈をベクトル化する際の問題が生じたと考えられる。一方この 7 問中 “主要語義 (本手法)” が誤っているものは 4 問であり、この部分で語義を主要語義に限定した効果が出ている。

実験結果で “全語義” よりも “主要語義 (本手法)” の方が劣っていた原因として、主要語義の推定が適切でなかったことが考えられる。本論文では名詞の主要語義を推定するのにコーパス中の語義の頻度を利用しているが、これはアドホックであり、これによって主要語義を推定できる保障はない。例えばある単語 a の頻度が非常に高く、また a の語義 m がややマイナーであるようなケースは容易に想像できる。この場合、ある多義語が語義 m を持っていれば、それは主要語義となってしまう。本論文では多義語の主要語義を 1 つに限定せずに複数使用することで、この問題に対処しているが、主要語義を推定する本手法は簡易すぎる。もっと精緻な手法を考案して、主要語義を推定する必要がある。この際、辞書の記述順が役に立つと予想している。辞書は主要な語義の順にその語義の定義が並んでいるからである。WordNet 等の利用するシソーラスの語義と辞書の語義の対応を自動的に行うことで主要語義を推定できる。また名詞に対してだけ主要語義を考慮したが、動詞などを含めた全単語を対象とすべきであった。今後はこの方向で研究を進めたい。

7 おわりに

本論文では文脈をベクトル化する際の多義語の問題を指摘し、多義名詞の主要語義だけを利用することを提案した。また実際にコーパスを利用して主要語義を推定した。実験では Senseval2 の辞書タスクの動詞 50 単語の語義識別問題を題材に、本手法の効果を調べた。語義識別問題に語義を使う効果は認められたが、その語義を主要語義に限定することに効果は見られなかっ

た。今後は主要語義の推定の部分を改良したい。

参考文献

[1] Francis Bond and Hitoshi Isahara and Sanae Fujita and Kiyotaka Uchimoto and Takayuki and Kuribayashi and Kyoko Kanzaki. Enhancing the Japanese WordNet. In *The 7th Workshop on Asian Language Resources, in conjunction with ACL-IJCNLP 2009*, 2009.

[2] Kikuo Maekawa. Design of a Balanced Corpus of Contemporary Written Japanese. In *Symposium on Large-Scale Knowledge Resources (LKR2007)*, pp. 55–58, 2007.

[3] 白井清昭. SENSEVAL-2 日本語辞書タスク. Vol. 10, No. 3, pp. 3–24, 2003.

表 1: 実験結果

見出し	訓練事例数	語義なし	全語義	主要語義（本手法）
ataeru	116	0.67	0.63	0.67
iu	366	0.94	0.94	0.94
ukeru	357	0.50	0.61	0.60
uttaeru	70	0.85	0.83	0.82
umareru	65	0.64	0.66	0.67
egaku	70	0.63	0.65	0.63
omou	465	0.90	0.89	0.89
kau	87	0.83	0.84	0.85
kakaru	115	0.51	0.62	0.55
kaku_v	135	0.68	0.68	0.67
kawaru	97	0.92	0.92	0.92
kangaeru	291	0.99	0.99	0.99
kiku	180	0.60	0.62	0.64
kimaru	117	0.96	0.96	0.96
kimeru	228	0.93	0.92	0.93
kuru	128	0.84	0.83	0.84
kuwaeru	109	0.89	0.90	0.89
koeru	119	0.77	0.79	0.78
shiru	246	0.97	0.97	0.97
susumu	112	0.46	0.46	0.44
susumeru	132	0.96	0.97	0.95
dasu	208	0.29	0.31	0.28
chigau	105	1.00	1.00	1.00
tsukau	270	0.97	0.96	0.97
tsukuru	183	0.59	0.63	0.61
tsutaeru	97	0.75	0.77	0.78
dekiru	75	0.81	0.81	0.81
deru	412	0.58	0.53	0.53
tou	71	0.68	0.63	0.65
toru	112	0.27	0.32	0.28
nerau	67	0.99	0.98	0.99
nokosu	98	0.79	0.79	0.79
noru	64	0.54	0.58	0.55
hairu	278	0.37	0.39	0.37
hakaru	84	0.92	0.92	0.92
hanasu	175	1.00	1.00	1.00
hiraku	224	0.80	0.90	0.81
fukumu	92	0.99	0.99	0.99
matsu	83	0.60	0.60	0.57
matomeru	93	0.80	0.81	0.80
mamoru	93	0.79	0.76	0.76
miseru	100	0.98	0.98	0.98
mitomeru	212	0.89	0.89	0.89
miru	408	0.72	0.73	0.72
mukaeru	93	0.89	0.89	0.89
motsu	267	0.50	0.55	0.51
motomeru	306	0.87	0.86	0.87
yomu	106	0.88	0.88	0.88
yoru	474	0.97	0.97	0.97
wakaru	178	0.90	0.90	0.90
平均		0.7714	0.7801	0.7734