複数の筆者の表記の違いを利用した同義語抽出の精度向上

村上 明子　　那須川 哲哉

日本アイ・ビー・エム（株） 東京基礎研究所
{akikom,nasukawa}@jp.ibm.com

1 はじめに

現在、大量の文章を分析して知識を発見するテキストマイニングが注目を集めている [2]。テキストマイニングでは言語処理を事前に行うことによって自然言語から知識を取り出すが、表現の揺らぎ等が多数あるため単語の同義性、多義性といった問題が精度を落とす原因となっている。今回、我々は同義性の解消のための処理に用いる辞書の作成について着目した。

専門用語を多数含む分野の場合、その文章が生成されたコミュニティによって使用する専門用語の表層表現が違う場合があり、マイニングの精度に直接影響を与えてしまう。また、筆者によって異なる省略語を用いるケースも見られる。したがって同義性解消の処理として一般的な同義語の辞書を用いる他に、分析の対象となる文章特有の同義語の辞書を用意することが必要となる。

本稿では、限定された分野における同義語の定義辞書を作成する支援を行なうため、複数の筆者のいる特定分野の文書コーパスから依存構造やその他の特性を用いて同義語の候補を自動的に抽出し、さらに筆者による表層表現の違いを利用することによって同義語候補取得の精度をあげることを示す。また、実際にテキストマイニングに用いられているデータを用いて実験を行い、その有効性を示す。

2 名詞の特徴量を用いた同義語候補の獲得

コーパスから同義語を抽出する研究は今までに多く行われている。Hindle[1] らは動詞と主語・目的語などの名詞の共起データを用いて名詞間の類似度を求める研究を行った。また Strzalkowski[3] らは動詞・形容詞の依存関係を用いて名詞の類似度を求めたあと、その名詞の抽象度の上下関係を見る研究を行っている。その他にコーパス中の文法情報を用いて単語の置換可能関係を抽出するものとして浦本 [4] の研究がある。

いずれの研究も、コーパスから名詞を抽出し、その類似性を求めるというものである。そのため、近年増加している技術的文書など特定分野に対しては未知語が頻出し、正しい類似度を計算することが出来ない。そこで、本手法では言語解析の段階で未知語をすべて名詞として扱うこととした。

そこで我々は、まず同義語定義辞書を得たい分野のコーパスに対して形態素解析と構文解析を行い、その結果から名詞（未知語で名詞として扱われたものを含む）に対して他の単語との係り受け関係を、頻度とともに情報ベクトルとして得、そのベクトルを内積によって比較し名詞間の類似度を得た [5]。これによって、限られた分野の文書に対して、入力名詞に対する類似性の高い名詞を同義語定義辞書のエントリーの候補として、類似度によって順位付けして得ることができる。しかし、この手法では入力名詞と同じ係り受け関係を持つ同義性のない名詞に対しても類似度が高くなり、ノイズとして候補の中に残ってしまう。特にノイズとして現れるものとしては、同じ上位概念を表す反意語などが挙げられる（「筆者」と「読者」など）。

この問題を解決するために、同義語が発生する原因の一つとなっている筆者ごとの表記の違いを用いて、ノイズとなっている同義語候補を除去する方法を考案した。次の章で詳細を示す。

	筆者 A	筆者 B	筆者 C	筆者 D	筆者 E
cust	6	335	2	3	2
customer	31	62	32	31	286
eu[1]	345	89	179	402	62
user	5	20	2	3	13

1 eu は End User の略語

表 1: 各々の筆者における"customer"の表記の頻度分布

3 筆者ごとの表記の違いを用いた精度向上

3.1 筆者ごとに分けられたデータの中での表記の特徴

同義語の生じる原因の一つに、一つの概念を示すのに複数の表層表現があることが挙げられる。そのため、特に複数の筆者によって書かれた文書が混在する場合、人それぞれが同じ意味の概念に対し異なる表現を用いていて文書間の語の統一が行なわれないという現象がよく見られる。したがって、業務上の報告書やアンケートなどの複数筆者の文書に対し、特に単語の同義性の理解が必要になるといえる。

しかし、単独の筆者によって書かれた文書に着目すると、文学作品など特別な場合を除き同じ意味の単語に対して同じ語で統一されている場合が多くみうけられる。Aという単語にa,bという表層表現がある場合、筆者は一度Aはaであると認識、あるいは確定するとめったに使う表現を変更することはない。

そこで、筆者ごとに分けられた実際のデータの中で一つの概念の表層表現がどのように分布しているかを調べた。調査した文書は、アメリカのIBMのコールセンターで電話を受けるコールテーカーが実際に報告書として記述したものである。表1に"customer"という単語について5人の筆者ごとの固有の表現と頻度を表す。"customer"の意味を表現する表層表現はデータ中に多く見受けられた。その中でも使用頻度の高い表現は"customer"、"cust"、"eu"、"user"など4種類であった。これらの単語の使用頻度を筆者別の文書で見てみると、表1に示したように筆者A、筆者Cと筆者Dはeu(End Userの略語)、筆者Bはcust、筆者Eはcustomerを主に用いていることが分かる。このことから、一人の筆者は同じ意味の単語に対して、主に一種類の表現をくりかえし利用する傾向にあることが分かる。この性質を利用すると、2章で求めた同義語候補の中から、反意語などのノイズとなっている単語を取り除くことができる。その方法を以下に述べる。

3.2 全体データからのノイズ消去法

まず、複数の筆者によって書かれた全体のデータから2章で述べた方法を用いて同義語を求めたい単語を入力として、各々の名詞の情報ベクトルの内積を用いてその単語の同義語の候補を複数得る。このとき、候補の中には複数のノイズとなる反意語等が同義語とともに混在している。

次に筆者ごとに分けられたデータを用意し、その部分データによる名詞の情報ベクトルを用いて同義語の候補を得る。このとき、類似度を比較する入力名詞の情報ベクトルは全体のデータによって作られたものである。つまり、全体のデータの中での名詞の特徴(どのような動詞と係り受けを持っているか等)と、筆者の中での名詞の特徴を比較していることになる。特徴が似ているものが入力単語と同義である可能性が高いと考えることができるが、筆者ごとのデータ中でその単語の表現として考えられるものは上位に現れており、候補として現れたもので類似度の下位の単語は同義ではないと考えることができる。

そこで、全体のデータから得られた同義語の候補の中で、筆者ごとのデータから得られた類似度の一番上位の単語と同じ単語が存在する場合、それは同義語である可能性が高いということで、"Absolute"というステータスを与える。反対に、筆者ごとのデータから得られた候補の中で類似度の順位が二位以下の単語とは同義語ではないと言う可能性が高いので、"Negative"というステータスを与え、同義語の候補の中からは除外する。これを複数の筆者のデータを用いて行なう。ただし、いずれかの筆者データから、一回でも"Absolute"と付けたられた単語については、他の筆者のデータ中で"Negative"と付与されてもステータスは変更されることはない。逆に、一旦"Negative"と付与された単語でも、他の筆者のデータ中で類似度が一位になっていれば、ステータスは"Absolute"に変化する。

3.3 具体例

ここでは具体例を挙げ、上記の方法を説明する。以下に実際のデータから得られた"battery"に関する同義語の結果を示す。 表2は全体のデータから得られた同義語の候補、表3は二人の筆者それぞれのデータから得られた同義語の候補である。全体データの結果を見ると、"battery"には"batt"や"BTY"など様々な表層表現が存在する。分かりやすさのために"battery"と同義の単語の前に○を表示してある。ところが、表3から分かるように、筆者ごとのデータを見ると、これらの筆者は主に"battery"という表層表現しか使っておらず、その

類似度の順位	同義語候補
1位	○ batt
2位	○ batterie
3位	ba
[redacted]	[redacted]
5位	○ BTY
6位	○ batterry
[redacted]	[redacted]
8位	bezel
[redacted]	[redacted]

表 2: "battery"の同義語候補 全体のデータからの結果

順位	同義語候補
1位	battery
2位	contoroller
[redacted]	[redacted]
4位	APM
[redacted]	[redacted]
6位	mark
[redacted]	[redacted]
8位	diskette
9位	checkmark
10位	boot

順位	同義語候補
1位	battery
2位	form
3位	protector
4位	DISKETTE
5位	Mwave
[redacted]	[redacted]
7位	mouse
[redacted]	[redacted]
9位	checkmark
10位	process

表 3: 筆者ごとのデータの結果

他の表現は類似度が二位以下のものには現れていない。

前の章で述べたように、筆者ごとのデータから得られた候補のうち、一位である"battery"以外のものは同義語ではないとして、全体のデータの中で"Negarive"ステータスとし候補からはずす。表 2、表 3 の中で、色がつけられているのが削除されたデータである。

このようにして、全体のデータの中から同義語を求めたい単語に対して類似度の高いものを候補として得、そのなかからノイズとして筆者ごとのデータから求めた結果を用いて消去し、候補の絞込みを行う。この例では、全体データの中に 4 つの正解が含まれ 6 個のノイズがあったが、操作後にはノイズは 2 個まで減っている。次の章では、実際のデータに対する実験とその結果を示す。

4 実験

4.1 実験データ

実験の対象としたテキストは、アメリカの IBM のコールセンターに電話で寄せられた問い合わせの内容を文書形式で記録したものであり、これらは顧客の問い合わせにおける質疑応答をオペレーターが手で入力したものである。実験に使用したデータは約一ヶ月分のデータであり、データはすべて英語で表記されている。これらのデータはオペレーターが電話を受けながら記述したものが多く、主語などの省略や略記の使用など、通常の文書に比べ文法あるいは表記の誤りが多く、通常の構文解析器ではエラーが多く発生する事がわかっている。そのため、今回の実験では品詞を統計 Tagger を用いて付与し、その結果をルールベースで範囲を決め、係り受けを判定する shallow parser を解析に用いた。shallow な構文解析器の方が話し言葉に近いコールセンターのデータに対しては有効だと考えられたため今回の実験に使用したが、parser の精度については今回は触れない事とする。また、係り受け関係を取り出す際に動詞に対する名詞の関係が主語であるか補語（あるいは目的語）であるかを区別した。

複数の筆者を含むデータから取り出された係り受けのペアは 3,350,200 個であり、名詞の異なり語は 29,961 種、動詞は 11,737 種である。この中を筆者ごとに分割し、データサイズの大きい筆者から順に最大 10 人のデータを、ノイズの消去のために使用することとした。10 人の筆者のデータから取り出された係り受けのペア数の平均は 37,453 個である。

実験には人手で作成した同義語の集合を用いその中で一番頻度の多い単語を検索語とした。一つの同義語の集合は平均 7.8 語の表層表現があった。今回、正解としてシステムが表示する順位を 20 位までとして実験を行なった。評価の方法として本稿では検索システムなどの評価の方法として一般的に用いられている適合率 (Precision) と再現率 (Recall) を用いた。適合率、再現率の定義は次のようである。

$$\text{適合率 } (P) = \frac{\text{システムで得られた正解数}}{\text{システムが正解だと回答した数}}$$

$$\text{再現率 } (R) = \frac{\text{システムで得られた正解数}}{\text{得られるべき正解数}}$$

これらから適合率と再現率を同時に評価するため F 値を求めた。F 値についての定義は以下のようである。

$$F - measure(F\ 値) = \frac{2 \times R \times P}{R + P}$$

4.2 実験結果

上記のデータを用いて、全体のデータおよび筆者ごとのデータを用いて実験を行なった。実験において、全体の結果から"Negative"ステータスのものを除いたものを正解としている。表4に結果を示す。 筆者ごとのデータ

	全体	筆者3人	筆者5人	筆者10人
適合率	0.210	0.288	0.329	0.339
再現率	0.624	0.562	0.595	0.595
F値	0.314	0.380	0.424	0.432

表 4: 筆者ごとのデータを用いてノイズを消去した結果

を用いてノイズを消去したものの方が、全体データだけからの結果より、適合率は上昇しているが、再現率は少し下がる傾向にある。これから、筆者が一つ以上の表層表現を用いてしまっている際に正解データもノイズとして削除されてしまっている可能性があると考えられる。しかし、総合の性能をあらわすF値の値は上昇をしている。

また、使う筆者ごとのデータの数を増やしていくことによって、F値が上昇していることが分かる。今回は筆者ごとのデータは１０人しか使用しなかったが、さらに増やしてF値つまりこのシステムの精度を見ることが必要であろう。

もし筆者が一つの表層表現しか用いていないのなら、筆者ごとのデータで一位のものだけを選ぶと同義語の候補として得る、という考えもできる。そこで、筆者ごとのデータで一位のもの、つまり前述の実験で"Absolute"ステータスが付与されたものだけを正解としたときの再現率を表5に示す。 この結果より、筆者ごとのデータ

	筆者3人	筆者5人	筆者10人
再現率	0.114	0.114	0.143

表 5: 筆者ごとデータ一位を正解とした時の再現率

で類似度が一位のものは必ずしも入力単語と同義語ではないことが分かる。それは、筆者によっては入力単語を同義であるものを一つも入力していない可能性も否定できないからである。このことから、筆者ごとのデータを用いて同義語を求める際に上位のものを同義語にするのではなく、上位以外のものをノイズとして扱い全体データの精度を上げるという方針が正しいことが分かった。

5 おわりに

本稿では、複数の筆者のいる特定分野の文書コーパスから依存構造その他の特性を用いて同義語の候補を自動的に抽出し、さらに書き手の情報を用いて精度をあげる手法を示した。

同義語の候補をデータから取得する際にはノイズが残ってしまうのですべて自動で行なうことはできず、最終的には人手で同義語かどうかの判定をすることが必要である。今回提案した手法では、適合率を上げることによって人手で同義語の集合を求める際の支援として適当だと言える。また、この手法によって低下してしまう再現率について、同義語の辞書の網羅性が必要な際には筆者ごとのデータより削除するノイズを完全に削除はせず、別枠でユーザーに提示することによって利便性を高めるといったことも考えられる。

言語処理を用いたテキストマイニングでは、同義性の解消は避けられない問題であるから、これから同義語辞書を作成する支援がますます必要となってくると考えられる。今回は純粋な同義を考えたが、今後マイニングの目的に合った抽象度の高い単語の集合も扱う辞書も考慮に入れなければいけないだろう。

参考文献

[1] Donald Hindle. Noun Classification From Predicate-Argument Structures. Proc.28th Annual Meeting of ACL, pp.268-275,(1990)

[2] Nasukawa, T. and Nagano, T. Text analysis and knowledge mining system. *IBM Systems Journal*, Vol. 40, No. 4, pp. 967–984,(2001)

[3] Tomek Strzalkowski and Barbara Vauthey. Information Retrieval Using Robust Natural Language Processing. Proc.30th Annual Meeting of ACL, pp.104-111,(1992)

[4] 浦本直彦. 文の多義性解消における置換可能関係を用いた事例の適用率向上. 人工知能学会誌、Vol.10 No.2 pp.242-249,(1995)

[5] 村上明子. 情報処理学会 第61回全国大会 2T-02 (2000)