名詞関連語知識に基づく文章のグラフ表現とその応用

進 義治　　黒橋 禎夫
京都大学大学院情報学研究科
shin@nlp.ist.i.kyoto-u.ac.jp, kuro@i.kyoto-u.ac.jp

1　はじめに

計算機による文章理解を実現するには，語と語の常識的な関係の知識，すなわち，関連語知識が重要であり，それを表現するモデルが必要となる．図 1 に，本研究で提案するモデルによって以下の文章の理解の状態をグラフ表現した例を示す．

1. 東京ドームが WBC の球場として選ばれ，多くの集客により大きな売上を得た．
2. そのおかげで，２０１４年の連結決算で増収増益となった．

図 1 の上半分は，1 文目まで解析した状態を表す．「WBC」は「世界ボクシング評議会」,「ワールド・ベースボール・クラシック」などの多義性があるが，周辺文脈との共通の関連語「野球」があることから，「ワールド・ベースボール・クラシック」であるとわかり，さらに文章中に明示的に出現しない「野球」が重要な概念であるとわかる．

図 1 の下半分は，2 文目まで解析した状態であり，話題は収益の話に移っており,「増収」,「増益」や，関連語の「業績」が重要であるとわかる．

本論文では，このように，関連語知識により重要な語をとらえ，また文が進むごとに重要語が変化していく様子を捉える文章のグラフ表現を提案し，その応用として語義曖昧性解消の実験結果を報告する．

2　関連研究

奥村ら [2] は概念ベースとよばれる関連語知識のデータベースを構築している．まず，国語辞書の見出し語に対して，語義定義文から自立語を抽出し，語に関する様々な知識を用いて不要な語をフィルタリングすることで，基本概念ベースを構築している．その後，新聞コーパスにおける共起を用いて基本概念ベースを拡張している．しかし，奥村らの概念ベースでは多義性の問題を扱っていない．

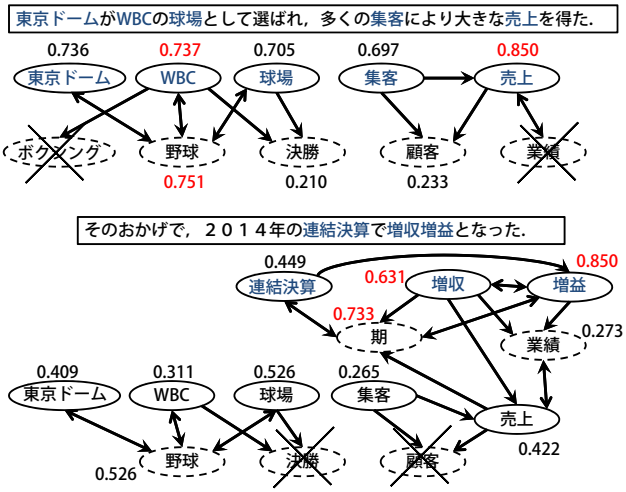

図 1: グラフによる関連語の重要度計算．点線のノードは文章中に現れない名詞を，実線のノードは文章中に現れる名詞を表す．関連語の関係を有向エッジで表す．

Okamoto ら [1] は，基本語彙 1,100 語と連想語 64,000 語からなる連想概念辞書を構築し，それ用いて文章のネットワークを構築するモデルを提案している．ネットワークのノードは単語タイプであり，活性値という値を持つ．単語が入力されるごとに，ネットワークの拡張と，各ノードの活性値の再計算を行う．「額 (ひたい，がく)」のように読みに曖昧性のある語に関して曖昧性解消の報告を行っている．これに対して本研究では，そのような限定をせず一般的な多義性を対象とし，さらに，100 万語以上の名詞を用いて，実テキストにおける頑健な関連語処理を実現している．

3　名詞関連語知識の獲得

3.1　名詞関連語知識

本研究では，関連語知識を考える単位として 1 単語の名詞 (単名詞) も複合名詞も扱う．ただし，あらゆる名詞を扱うことはできないため，あらかじめ扱う名詞の集合を定義しておく．以下では，その集合の要素を**名詞表現**とよぶことにする．名詞表現として，我々が構築した語彙データベース [3] に含まれる語彙から，

表 1: 名詞表現の分類

知識源	語義数	個数	例
JUMAN 基本語辞書	単語義	22.5k	野球
	多義	88	トラック
Wikipedia	単語義	1.19M	爽健美茶
			日本+語
	多義	67.0k	ＷＢＣ
			河原+町
Web テキスト	単語義	25.3k	アドセンス
			模擬+戦

	野球	WBC	選手	投手	ボクシング	…
野球	< 1.00	0.05	0.18	0.08	0	... >
WBC 1 (World Boxing Council)	< 0	1.00	0.11	0	0.28	... >
WBC 2 (World Baseball Classic)	< 0.90	1.00	0.69	0.28	0	... >

図 2: 関連語ベクトルの例

形態素解析器 JUMAN の基本語辞書に含まれる単名詞，Wikipedia の見出し語から獲得した単名詞と複合名詞，Web テキストから獲得した単名詞と複合名詞を用いる (表 1).

本研究では，名詞関連語知識をベクトルモデルによって表す．各名詞表現は関連語ベクトルを持っており，関連語ベクトルの 1 つの次元は名詞表現 1 個に対応する．また，多義の名詞表現は語義ごとに異なる関連語ベクトルを持つ．図 2 の関連語ベクトルでは,「野球」は単語義,「WBC」は多義である.「野球」からみた「選手」のように，関連語ベクトルにおける次元が正の値をもつ名詞表現を関連語とみなす．その値 0.18 は関連の強さを表す値で，**関連度**とよぶことにする．自分自身との関連度は 1.00 に，他の名詞表現との関連度最大値は 0.90 に固定している．語義ベクトルの次元において，多義の名詞表現，例えば「WBC」は 1 つの次元であり，語義を区別しない．

3.2 単語義名詞表現の関連語

ある見出し語の定義文中に現れる語は，見出し語と関係が強いと考えられる．そこで，単語義名詞表現を見出し語とする国語辞典と Wikipedia 記事の定義文において，自分自身を除く名詞表現の出現頻度 (TF) を求める[1]．この際，複合名詞内部の名詞もカウントするが,「広島+県」の「県」のように複合名詞内部の 1 文字の名詞は経験的に副作用の危険があるので無視する．

Web テキストと定義文を組み合わせて，名詞表現の共起頻度データベースを構築する．まず，Web テキスト 4.2 億文から，1 文ごとに最長マッチで名詞表現を抽出して共起頻度をカウントする．Web テキストは規模が大きく，十分に共起頻度が得られるため，複合名詞内部の名詞はカウントしない．次に，定義文も Web テキストの一種とみなし，定義文における TF を見出し語との共起頻度として Web での共起頻度に加える．低頻度の名詞表現においては，Web テキストから十分な共起頻度が得られないため，定義文が特に重要である．

[1]Wikipedia 記事による定義文は，見出し語の定義が書かれていることが多い，記事の第一段落にある文章を用いる

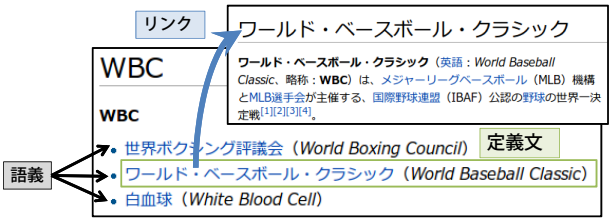

図 3: Wikipedia 曖昧さ回避ページ

関連語を獲得する対象の名詞表現 w_1 に対して，それ以外の任意の名詞表現 w_2 と CoScore を求める．

$$\mathrm{CoScore}(w_1, w_2) = f(w_1, w_2) * \mathrm{PMI}(w_1, w_2)$$

ただし，$f(w_1, w_2)$ は共起頻度データベースにおける w_1 と w_2 の共起頻度であり，$\mathrm{PMI}(w_1, w_2)$ は w_1 と w_2 の自己相互情報量である．w_1 と共起しやすく，かつ低頻度でない w_2 の CoScore が高くなりやすい．

CoScore の高いものから 25 個の名詞表現を関連語として獲得する．自分以外の名詞表現との関連度最大値が 0.90 になるように，CoScore を正規化した NormalizeCoScore を関連度として用いる．

$$\mathrm{NormalizeCoScore}(w_1, w_2) = 0.9 * \frac{\mathrm{CoScore}(w_1, w_2)}{\mathrm{CoScoreMax}(w_1)}$$

定義文中に出現する名詞表現は関係が強いと考えられるので，CoScore の制約を緩め，NormalizeCoScore $\geq$ 0.01 なら関連語として獲得する．このときも関連度は NormalizeCoScore の値を用いる．

3.3 多義名詞表現の関連語

多義名詞表現の各語義を 1 つの単語義の名詞表現とみなして，3.2 節で述べた単語義の関連語獲得手法を適用することで，語義の関連語を獲得する．

本研究では，多義名詞表現とその語義は Wikipedia の曖昧さ回避ページから獲得している．曖昧さ回避ページは図 3 のように，見出し語に対して複数の語義とその定義文が箇条書きされており，一部の語義には，その語義と関係する Wikipedia 記事へのリンクがある．関連語獲得のときに用いる定義文には，曖昧さ回

避ページの定義文を利用する．リンクが存在する語義の場合，リンク先の Wikipedia 記事の第一段落も定義文として加える．

Web テキストにおける多義名詞表現と他の名詞表現の共起頻度は，語義ごとにカウントする必要がある．表層的には出現の語義を区別できないため，Wikipedia のリンクをもとに教師あり学習によって Web テキストにおける出現語義を推定し，共起頻度をカウントする．

まず，Wikipedia のリンクから語義の正解データを作成する．Wikipedia の「松坂大輔」の記事には，表記が「WBC」でリンク先が「ワールド・ベースボール・クラシック」となっているリンクがある．このようなリンクで，リンク先（ワールド・ベースボール・クラシック）と，表記の多義名詞表現 (WBC) のある語義のリンク先が一致するとき，語義の正解データとして利用できる．このように獲得した正解データを学習データとして用いて，教師あり学習により Web テキストにおける多義名詞表現の語義を推定する．

このように語義推定を行うのは，次のような多義名詞表現とする．まず，十分な量の出現を確保するため，4.2 億文の Web テキストにおける出現頻度 10,000 回以上の名詞表現だけを対象にする．さらに，正しく分類するには十分な量の学習データが必要と考えられるため，学習データが 60 以上存在する多義名詞表現のみ対象とする．以上の条件により，対象にするのは約 67,000 個の多義名詞表現のうち 924 個である．

リンクの前後 100 文字ずつで形態素解析を行い，単語ユニグラムを素性として SVM(Support Vector Machine) を学習する．SVM のパッケージとして TinySVM[2] を用いた．カーネルは線形カーネルとして，それ以外のパラメータはデフォルトの値を用いた．ただし，1 つの語義の学習データのみ得られた多義名詞表現は，SVM を利用せず全てその語義であると推定する．対象とする 924 個の多義名詞表現に対して得られた正解データ 108 万件中，多義名詞表現ごとに 4/5 を学習データとして利用し，残りの 1/5 は多義性解消実験 (5 章) において評価データとして利用する．

教師あり学習の対象でない多義名詞表現の語義は，Web テキストにおける共起頻度が全く得られないため，定義文のみから関連語を獲得することになる．

4 グラフによる文章中の関連語の重要度計算

1 章の図 1 のように，文章中の名詞表現とその関連語をグラフのノードとし，関連語の関係をエッジにしたグラフを構築する．各ノードは重要度の値を持っている．1 文入力するごとにグラフは以下のステップにより更新され，その文までの時点における重要度を計算することができる．

1. ノード生成　読み込んだ文から，最長マッチにより名詞表現を抽出し，グラフのノードとする．重要度初期値は 1 にする．抽出した名詞表現の関連語もノードにして，重要度初期値を 0 にする．すでにグラフにノードが存在する名詞表現の場合，文中に出現するなら重要度を 1 にして，関連語として出現するならなにもしない．多義名詞表現も 1 つのノードとみなし，語義は区別せず全ての語義の全ての関連語のノードを加える．

2. エッジ探索　任意の 2 ノード間で名詞表現とその関連語の関係があれば，名詞表現→関連語の有向エッジ (リンク) を張る．ただし，文章中に出現しない (非明示的な) 関連語のノード同士で関連語の関係があっても，文章の背景知識から離れていることが多いため，リンクを張らない．非明示的で 1 つのノードのみと隣接するノードは，次ステップである重要度の伝搬においてノイズとなるので，グラフから除去する．

3. 重要度の伝搬　リンク元からリンク先のノードに重要度を伝搬させることで，文章中の名詞表現の関連語であるノードの重要度を高くする．各ノードの各リンクに対して，(リンク元重要度) * (関連度) の値をリンク先の重要度に 1 回だけ加える．多義名詞表現の関連語における関連度は，全ての語義から最大値をとる．さらに，文章に特有の名詞表現の重要度を高くするため，各ノードに Web 4.2 億文における名詞表現の IDF をかける．

4. 重要度の正規化　次の入力文に含まれる名詞表現の重要度初期値よりも重要度が大きくならないよう，全てのノードの重要度に (0.85/重要度最大値) をかけて，最大 0.85 になるよう正規化する．一旦重要でないとみなされた名詞表現は以後の文章で出現しない限り重要になることはないと考え，重要度が 0.2 未満になったノードを削除する．

5 重要度計算の語義曖昧性解消への応用

文章中の関連語の重要度計算を利用して，多義名詞表現の語義曖昧性解消を行う．文脈中での適切な語義の関連語は文章の背景知識に沿っており，重要度が高いと考えられる．また，多義語がどの語義で出現しや

[2] http://chasen.org/~taku/software/TinySVM/

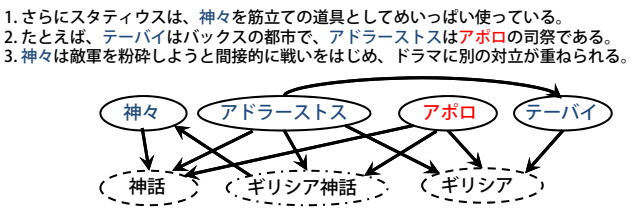

図 4: 語義曖昧性解消の改善例

すいかを表した語義確率も語義曖昧性解消に有用である．そこで，文脈における語義スコアを (多義語自身の重要度と語義関連語の重要度の総和 * 語義確率) で求める．語義確率は，3.3 節で述べた Wikipedia のリンクから得た正解データを用いて計算する．

1 文解析するごとに，曖昧性解消がされていない多義名詞表現の各語義で語義スコアを求める．語義スコア最大と 2 番目の差がしきい値 θ 未満の場合，その文の時点ではどちらの語義が正しいか判別できないと考え，次の文まで曖昧性を引き継ぐ．差が θ 以上の場合，語義スコア最大の語義を出力する．文章の最後の文まで解析しても語義を判別できない場合，最後の文を解析した時点で語義スコアが最大の語義を出力する．

提案手法の語義曖昧性解消に対する有効性を調べるために実験を行った．正解データが 5 例以上ある多義名詞表現を対象にして，Wikipedia のリンクから獲得した語義の正解データを多義名詞表現ごとに 4/5 の学習データと 1/5 の評価データに分割し，学習データから語義確率を計算した．Web での出現を分類する対象の 924 個の名詞表現では，学習データを用いて SVM も学習した．評価データ中から 1,806 例をランダムに選択して，評価対象とした．ベースラインとして，SVM を学習した多義名詞表現に関しては SVM を使い，それ以外では語義確率が最大の語義 (MFS) を選択する SVM+MFS を用いた．文脈の長さと精度の関係を調べるため，多義名詞表現を含む文とその前後 n 文 ($n = 0 \sim 5$) を文脈としたときの精度を求めた．

結果を表 2 に示す．提案手法は全体ではベースラインよりも精度が良いが，SVM が使える多義名詞表現に関しては SVM の方がよい．提案手法は，学習事例があまり多くない多義名詞表現に対しても，語義曖昧性解消できるという特徴がある．ベースラインからの改善例を図 4 に示す．この例では，「アポロ」がギリシア神話の神の語義で用いられている．SVM は学習されておらず，MFS は楽曲のタイトルの語義である．この文脈では，「アドラーストス (ギリシア神話の神)」が「ギリシア神話」を関連語にもつため，ギリシア神話の神の語義が正しく選べている．

表 2: 語義曖昧性解消結果

全体 (1,806 例/1,806 例)

Context	0 文	±1 文	±2 文	±3 文	±4 文	±5 文
Proposed	0.864	0.865	**0.870**	0.865	0.868	0.867
SVM+MFS	0.832	0.836	0.840	0.843	0.843	0.845

SVM を学習した多義名詞表現 (820 例/1,806 例)

Context	0 文	±1 文	±2 文	±3 文	±4 文	±5 文
Proposed	0.878	0.884	0.891	0.889	0.893	0.894
SVM	0.900	0.909	0.918	0.924	0.926	**0.929**

SVM を学習していない多義名詞表現 (986 例/1,806 例)

Context	0 文	±1 文	±2 文	±3 文	±4 文	±5 文
Proposed	0.838	0.850	**0.853**	0.846	0.847	0.845
MFS	0.777					

6 おわりに

本研究では，名詞関連語知識のデータベースを構築し，関連語知識を用いて文章中の名詞表現とその関連語の重要度計算を行う手法を提案し，その重要度計算を応用して語義曖昧性解消の実験を行った．

今回は明示的な語義の曖昧性を扱ったが語義が 1 つと思われる「牛」のような名詞でも，「牛肉，牛革」など関連語ごとに異なる側面があると考えられる．語義曖昧性解消と同様に，文脈に応じて重要な側面を重要度計算によりとらえられると考えている．

また，紙面の都合上本稿では報告できないが関連語の重要度計算の別の応用として，「プレゼン+スライド」，「プレゼンス+ライド」のように分割に曖昧性のある複合名詞の曖昧性解消を行うタスクを行っており，既存の形態素解析器と比べて良好な結果を得ている．

参考文献

[1] Jun Okamoto and Shun Ishizaki. Homographic ideogram understanding using contextual dynamic network. In *Proceedings of LREC 2010*, pp. 1180–1186, May. 2010.

[2] 奥村紀之, 土屋誠司, 渡部広一, 河岡司. 概念間の関連度計算のための大規模概念ベースの構築. 自然言語処理, Vol. 14, No. 5, pp. 41–64, Oct. 2007.

[3] 黒橋禎夫, 進義治, 柴田知秀, 村脇有吾, 河原大輔. 日本語語彙知識の統一的・整合的管理のデザイン. 言語処理学会 第 19 回年次大会, pp. 26–29, Mar. 2013.