SYNGRAPH: 国語辞典とコーパスから自動抽出した同義・上位下位関係に基づく柔軟マッチング

柴田 知秀　　小谷 通隆　　黒橋 禎夫

平成 22 年 4 月 26 日

1　概要

論文 [4, 3, 2] を参照。

SYNGRAPH のベースとなるのは、もとの文の依存構造木であり、そのノードは 1 つの自立語と 0 個以上の付属語からなるもので、以下基本ノードとよぶ。そして、基本ノードの自立語に同義グループがあれば、その SYNID を別のノードとして与える（これを SYN ノードとよぶ）。図 1 では色のついたノードが基本ノード、それ以外のノードが SYN ノードである基本ノードを同義表現の SYN ノードと区別するのは、完全マッチを優先するためである。

さらに、複数のノードに対応する表現に同義グループがあれば、その SYNID のノードも加える。図 1 の＜最寄り＞がこのような SYN ノードである。そしてさらに、各 SYN ノードに対して、類義表現データベースにおいて上位の同義グループがあれば、その SYN ノードを加える。

各ノードには、もとの文の表現からのずれに応じたスコア、NS(Node Score) を与える。

2　プログラム・データ

以下のコマンドにより、cvs でプログラム・データを取得することができる。

```
% cvs -d reed.kuee.kyoto-u.ac.jp:/share/service/cvs co SynGraph
```

主な辞書データ・スクリプトなどを図 2,?? に示す。

3　国語辞典の整形・マージ

4　国語辞典からの語の関係の抽出

国語辞典・コーパスから類義表現を抽出し、ゴミの削除、マージ、多義性解消などを行なう。

国語辞典から、以下の関係を抽出する。

- 同義表現
- 上位下位関係
- 反義関係
- 同義句

関係の抽出は人手で整備したパターンで行なう。パターンは数十あり、KNP のルールファイルで記述されている。以下がルールファイルである。

```
SynGraph/ExtractSynfromDic/tools/def_sentence.rule
```

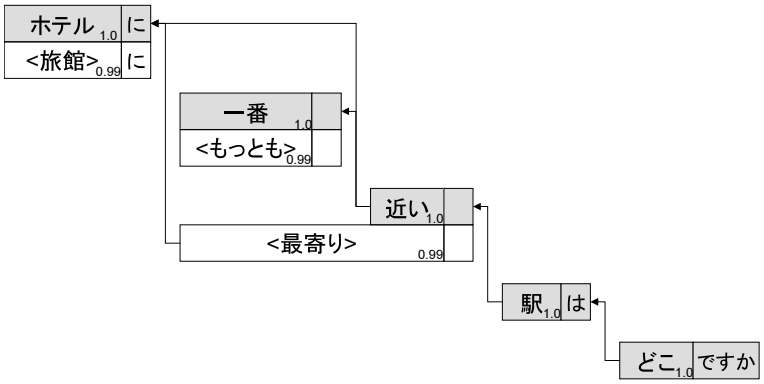

図 1: SYNGRAPH の例

```
VERSION : バージョン (例: 1.8-20080822)

ExtractSynfromDic/ : 辞書から同義表現抽出ツール
  orig/ : 元データ
    wikipedia.txt       Wikipedia から抽出した定義文
    aimai_list.txt      Wikipedia の曖昧さ回避ページから抽出した多義語リスト

  scripts/ : スクリプト
    x4syn.pl            同義・上位下位・反義関係抽出
    extract_text_from_wikipedia.pl          Wikipedia から定義文抽出

dic/ : 同義表現辞書
  rsk_iwanami/ : RSK、iwanami からの知識抽出結果
    synonym.txt         同義グループデータ
    definition.txt      定義文データ
    isa.txt             上位下位データ
    antonym.txt         反義データ

  web_news/ : WEB、新聞記事からの知識抽出結果
    nation.txt          wikipedia の国名ページがら抽出
    news.txt            毎日新聞・読売新聞・朝日新聞中の括弧表現から抽出
    www.txt             WWW 中の括弧表現から抽出
    all.txt             上記のファイルをマージしたもの

  wikipedia/ : Wikipedia からの知識抽出結果
    isa.txt             上位下位データ
    redirect_synonym_share_character_frequent.txt           リダイレクトから抽出した同義語
    aimai_synonym_isa.txt           曖昧さ回避ページから抽出した同義語・上位語

dic_change/ : 整形後の同義表現辞書
  synonym_dic.txt               辞書から抽出した同義グループデータ
  definition.txt                辞書から抽出した定義文データ
  isa.txt                       辞書から抽出した上位下位データ
  isa_wikipedia.txt             Wikipedia から抽出した上位下位データ
  antonym.txt                   辞書から抽出した反義データ
  synonym_web_news.txt          Web と新聞から抽出した同義グループデータ
```

図 2: 主な辞書データ

4.1 上位下位関係

4.2 同義表現

4.3 反義関係

辞書の記号「反対語」を利用し、反義関係を抽出する。以下の例では、「みぎ」の反対語として「左」が抽出される。

```
perl/ : Perl モジュール
  SynGraph.pm                   SYNGRAPH を扱うモジュール
  CalcSimWithSynGraph.pm        SYNGRAPH を用いてマッチングを行うモジュール

scripts/ : スクリプト
  merge_dic.sh                  辞書データ整形のスクリプト
  build.sh                      コンパイル (DB 作成) のスクリプト
  conv_syndb.pl                 コンパイルの前処理
  compile.pl                    コンパイル (類義表現 DB の SYNGRAPH 化)
  sort_synhead.pl               類義表現 DB のヘッドハッシュをソート
  knp_syn.pl                    SYNGRAPH 化のテストプログラム

syndb/ : 同義表現データベース
  i686/ :
    synhead.cdb                 類義表現 DB(head から同義グループを呼ぶ)
    syndata.mldbm               類義表現 DB(同義グループに属す SYNGRAPH を呼ぶ)
    synparent.cdb               上位データベース (上位グループを呼ぶ)
    synantonym.cdb              反義データベース (反義グループを呼ぶ)
    syndb.cdb                   同義グループの中身を保存

  x86_64/ :
    (i686/と同じ)

  cgi/ : CGI で使用する DB
    synhead.cdb                 類義表現 DB(head から同義グループを呼ぶ)
    synparent.cdb               上位データベース (上位グループを呼ぶ)
    synantonym.cdb              反義データベース (反義グループを呼ぶ)
    syndata.mldbm               SYNGRAPH 化の LOG 付 syndata
    syndb.cdb                   同義グループの中身を保存
    synnumber.cdb               番号から同義グループを呼ぶ
    synchild.cdb                下位データベース (下位グループを呼ぶ)
    log_dic.cdb                 辞書からの情報抽出の LOG
    log_isa.cdb                 DB の上位下位関係付与の LOG
    log_antonym.cdb             DB の反義関係付与の LOG

cgi : CGI ソース
  index.cgi             SynGraph の DEBUG 用 CGI のソース
```

図 3: 主なスクリプト・同義表現データベース

みぎ

北を向いたときに、東にあたるほう。

<用例>右がわ。

<反対語>左。</反対語>

4.4 同義句

5 コーパスからの同義表現抽出

5.1 括弧表現を利用したコーパスからの同義表現抽出

笹野らの手法 [1] を利用する。

6 類義表現の整理

以下のコマンドで行なうことができる。

```
% cd scripts
% ./merge_dic.sh
```

オプションを以下に示す。

- -m: 人手での修正を反映
- -w: Wikipedia 用

6.1 同義表現のマージ

複数の語が同一である類義表現のグループを、一つの類義表現のグループにマージする。副産物として、多義性が解消できる (場合がある)。

以下の 2 つの類義表現のグループを、

- 書籍/しょせき:1/1:1/1 本/ほん 書物/しょもつ 図書/としょ
- 書物/しょもつ:1/1:1/1 本/ほん 書籍/しょせき

以下のようにマージする。

- 書物/しょもつ:1/1:1/1 本/ほん 書籍/しょせき:1/1:1/1 図書/としょ

7 同義表現データベースのコンパイル

以下のコマンドで行なうことができる。

```
% cd scripts
% ./build.sh
```

主なオプションを以下に示す。

- -o: orchid
- -l: CGI 用
- -j: JUMAN コマンドを指定
- -r: JUMANRC を指定
- -d: knp を-dpnd で動かす (デフォルトは格解析)
- -w: Wikipedia 用

以下が順に行なわれる。

- 同義グループに SYNID 付与
- 構文解析
- コンパイル

以下では各手順について詳しく述べる。

7.1 同義グループに SYNID 付与

7.2 構文解析

7.3 コンパイル

8 SYNGRAPH フォーマット

8.1 フォーマット

KNP の解析結果に SYNGRAPH の情報をうめこむことができる。図 4 に「ホテルに一番近い駅」の解析結果を示す。

以下に notation の説明を示す。

- 「#」、「*」、「+」行は KNP の解析結果と同じ。
- 「!!」が付いている行は同じ基本句に対応している基本ノード、SYN ノードに共通する情報を出力する。 左から順に、対応している基本句番号、親のノードが対応している基本句番号（複数ある場合は「/」でつないでいる）、係り方、見出し、さらに存在すれば文法フラグ、格解析結果など。
- 「!」が付いている行は各ノードの情報を出力する。左から順に、対応する基本句番号、SYNID（基本 ID も SYNID として表記）、スコア、さらに存在すれば文法素性、上位語、反義語などが出力される。以下に文法素性を示す。
  - 否定
    - KNP の基本句に付与されている素性 <否定表現> をみている
  - 可能
  - 尊敬
  - 受身
  - 使役

  上位語・反義語はマッチングした元の表現に対するものであり、上記の文法素性とは意味合いが異なる。特に反義語は混乱するので、以下に例をあげる。

  1. 最悪だ<反義語> = 最善でない
  2. 最悪だ<否定> = 最悪でない
  3. 最悪だ<反義語><否定> = 最善だ

  1. は元の表現が「最善でない」であり、「最善」の反義語である「最悪」と否定表現がキャンセルされて作られたノードである。2. は「最悪」 + 否定表現のノードとなる。3. は「最善だ」と「最悪でない」が同義[1]であるため、このようになる。

  その他に、あれば下位語数が出力される。

以下のサンプルプログラムでテストすることができる。

```
% cd scripts
% perl knp_syn.pl -s ホテルに一番近い駅
```

[1] 厳密には反義+否定は元の表現と同義ではないが、これを扱うのは今後の課題である。

主なオプションを以下に示す。

- -relation: 上位語を付与
- -antonym: 反義語を付与
- -dbdir: 同義表現データベースを指定

図 4 に示した解析結果は以下のコマンドで動かしたものである。

```
% perl knp_syn.pl -s ホテルに一番近い駅 -relation
```

★複数ノードに関する仕様

8.2 KNP::Result で読みこむ

前節で説明したフォーマットを読み込んで処理するサンプルプログラム (SynGraph/scripts/read-knp-result.pl) を図 5 に示す。

KNP::SynNodes オブジェクトで提供されているメソッドを以下に示す。

synnode 全ての Syn ノードを返す。

tagid 対応する基本句 ID を返す。

tagids 対応する基本句 ID(配列) を返す。

parent 係り先の基本句 ID を返す。

parentids 係り先の基本句 ID(配列) を返す。

dpndtype 依存関係の種類 (D,P,I,A) を返す。

midasi 見出しを返す。

feature 文法素性を返す。

また、KNP::SynNode オブジェクトで提供されているメソッドを以下に示す。

tagid 対応する基本句 ID を返す。

tagids 対応する基本句 ID(配列) を返す。

synid SynID を返す。

synid スコアを返す。

feature 文法素性を返す。

9 課題

- 辞書を作る・インターフェース設計
  - 多義語で対応する ID がわからない場合に「1/1:?/2」のような ID をふる (例: 仕事)
- 同義グループ内をさらにグループに分ける (子供 = 童 問題)
  - 分布類似度が閾値以下の場合に別グループ ?
- 同義関係 (0.99), 上位下位関係 (0.7) のスコアを分布類似度に
  - ホーム, ハウス, 家というグループに対して、ホーム ↔ ハウス, ホーム ↔ 家, ハウス ↔ 家 の類似度をあらかじめ計算しておき、マッチした語間の類似度をかける
- 不要な同義句の削除
  - 例: 景気＝世の中の金回りの具合
  - Web 全体で「世の中の金回りの具合」がマッチする頻度を調べて閾値以下なら削除
- 一語のノードができるのを防ぐ
  - 「偶発」の上位ノードとしての、<発生する>
  - 「建設」の反義ノードとしての、<破壊>
- 品詞の利用
  - 「食事」と「食べる」,「買い物」と「買う」がマッチしてしまう
- 付属語の同義
  - 「使用済み」と「使い終える」
- 高速化
  - Syn ノードが付与できるかどうかをチェックする時に工夫
- 「高くなる = 高まる」、「高くする = 高める」をマッチさせる
  - ContentW.dic の「形容詞派生」をみてノードを作る

参考文献

[1] Ryohei Sasano, Daisuke Kawahara, and Sadao Kurohashi. Improving coreference resolution using bridging reference resolution and automatically acquired synonyms. In *Anaphora: Analysis, Alogorithms and Applications, 6th Discourse Anaphora and Anaphor Resolution Colloquium (DAARC2007)*, 2007.

[2] Tomohide Shibata, Michitaka Odani, Jun Harashima, Takashi Oonishi, and Sadao Kurohashi. SYNGRAPH: A flexible matching method based on synonymous expression extraction from an ordinary dictionary and a web corpus. In *Proceedings of IJCNLP2008*, 2008.

[3] 小谷通隆, 中澤敏明, 柴田知秀, 黒橋禎夫. SYNGRAPH データ構造における述語項構造の柔軟マッチング. 言語処理学会 第 13 回年次大会, pp. 43–46, 3 2007.

[4] 大西貴士, 黒橋禎夫. 国語辞典からの類義表現抽出と SYNGRAPH データ構造による柔軟マッチング. 言語処理学会 第 12 回年次大会, 3 2006.

```
# S-ID:1 KNP:3.0-20080214 DATE:2008/08/06 SCORE:-30.72806 SynGraph:1.7-20080805
* 2D <SM-主体><SM-場所><SM-組織><BGH:ホテル/ほてる><文頭><ニ><助詞><体言><係:ニ格><区切:0-0><RID:1180><格要素><連用要素><正規化代表表記:ホテル/ほてる><主辞代表表記:ホテル/ほてる>
+ 2D <SM-主体><SM-場所><SM-組織><BGH:ホテル/ほてる><文頭><ニ><助詞><体言><係:ニ格><区切:0-0><RID:1180><格要素><連用要素><名詞項候補><先行詞候補><正規化代表表記:ホテル/ほてる><解析格:ニ>
ホテル ほてる ホテル 名詞 6 普通名詞 1 * 0 * 0 "組織名末尾 カテゴリ:場所-施設 ドメイン:レクリエーション:ビジネス 代表表記:ホテル/ほてる" <組織名末尾><カテゴリ:場所-施設><ドメイン:レクリエーション:ビジネス><代表表記:ホテル/ほてる><正規化代表表記:ホテル/ほてる><文頭><記英数カ><カタカナ><名詞相当語><自立><内容語><タグ単位始><文節始><固有キー><文節主辞>
に に に 助詞 9 格助詞 1 * 0 * 0 NIL <品曖><ALT-に-に-に-9-3-0-0-NIL><品曖-格助詞><品曖-接続助詞><かな漢字><ひらがな><付属>
!! 0 1,2/2D <見出し:ホテルに><格解析結果:ニ格>
! 0 <SYNID:ホテル/ほてる><スコア:1>
! 0 <SYNID:s9192:宿泊施設><スコア:0.99>
! 0 <SYNID:s1866:ホテル/ほてる><スコア:0.99>
! 0 <SYNID:s249:宿/やど><スコア:0.693><上位語><下位語数:1>
* 2D <BGH:一番/いちばん><相対名詞修飾><用言弱修飾><副詞><修飾><係:連用><区切:0-4><RID:1398><連用要素><正規化代表表記:一番/いちばん><主辞代表表記:一番/いちばん>
+ 2D <BGH:一番/いちばん><相対名詞修飾><用言弱修飾><副詞><修飾><係:連用><区切:0-4><RID:1398><連用要素><正規化代表表記:一番/いちばん><解析格:修飾>
一番 いちばん 一番 副詞 8 * 0 * 0 * 0 "相対名詞修飾 用言弱修飾 代表表記:一番/いちばん" <相対名詞修飾><用言弱修飾><代表表記:一番/いちばん><正規化代表表記:一番/いちばん><漢字><かな漢字><自立><内容語><タグ単位始><文節始><文節主辞>
!! 1 2D <見出し:一番><格解析結果:修飾格>
! 1 <SYNID:一番/いちばん><スコア:1>
! 1 <SYNID:s523:トップ/とっぷ><スコア:0.99>
! 1 <SYNID:s2196:何より/なにより><スコア:0.99>
! 1 <SYNID:s1987:一番/いちばん><スコア:0.99>
! 1 <SYNID:s1988:最も/もっとも><スコア:0.99>
* 3D <BGH:近い/ちかい><連体修飾><用言:形><係:連格><レベル:B-><区切:0-5><ID:（形判連体）><RID:765><連体並列条件><正規化代表表記:近い/ちかい><主辞代表表記:近い/ちかい>
+ 3D <BGH:近い/ちかい><連体修飾><用言:形><係:連格><レベル:B-><区切:0-5><ID:（形判連体）><RID:765><連体並列条件><正規化代表表記:近い/ちかい><用言代表表記:近い/ちかい><格要素-ガ:駅><格要素-ニ:ホテル><格要素-ト:NIL><格要素-デ:NIL><格要素-カラ:NIL><格要素-ヨリ:NIL><格要素-マデ:NIL><格要素-ヘ:NIL><格要素-時間:NIL><格要素-外の関係:NIL><格要素-ノ:NIL><格要素-修飾:一番><格要素-トスル:NIL><格要素-ニクラベル:NIL><格要素-ガ２:NIL><格要素-ヲノゾク:NIL><格フレーム-ガ-主体><格フレーム-ニ-主体><格フレーム-ヨリ-主体><格フレーム-ノ-主体><格フレーム-ニクラベル-主体><格フレーム-ガ２-主体><格フレーム-ガ-主体ｏｒ主体準><格フレーム-ガ２-主体ｏｒ主体準><格フレーム-ニ-主体ｏｒ主体準><格関係 0:ニ:ホテル><格関係 1:修飾:一番><格関係 3:ガ:駅><格解析結果:近い/ちかい:形 6:ガ/N/駅/3/0/?; ニ/C/ホテル/0/0/?; ト/U/-/-/-/-; デ/U/-/-/-/-; カラ/U/-/-/-/-; ヨリ/U/-/-/-/-; マデ/U/-/-/-/-; ヘ/U/-/-/-/-; 時間/U/-/-/-/-; 外の関係/U/-/-/-/-; ノ/U/-/-/-/-; 修飾/C/一番/1/0/?; トスル/U/-/-/-/-; ニクラベル/U/-/-/-/-; ガ２/U/-/-/-/-; ヲノゾク/U/-/-/-/->
近い ちかい 近い 形容詞 3 * 0 イ形容詞アウオ段 18 基本形 2 "代表表記:近い/ちかい" <代表表記:近い/ちかい><正規化代表表記:近い/ちかい><かな漢字><活用語><自立><内容語><タグ単位始><文節始><文節主辞>
!! 2 3D <見出し:近い>
! 2 <SYNID:近い/ちかい><スコア:1>
! 2 <SYNID:s199:親しい/したしい><スコア:0.99>
! 2 <SYNID:s11:付近/ふきん><スコア:0.99>
! 2 <SYNID:s1201:所在/しょざい><スコア:0.693><上位語><下位語数:323>
! 2 <SYNID:s419:近い/ちかい><スコア:0.99>
!! 1,2 3D <見出し:近い>
! 1,2 <SYNID:s21291:最寄り/もより><スコア:0.99>
! 1,2 <SYNID:s1201:所在/しょざい><スコア:0.693><上位語><下位語数:323>
* -1D <SM-主体><SM-場所><SM-組織><BGH:駅/えき><文末><体言><用言:判><体言止><一文字漢字><レベル:C><区切:5-5><ID:（文末）><裸名詞><RID:1470><提題受:30><主節><定義文主辞><正規化代表表記:駅/えき><主辞代表表記:駅/えき>
+ -1D <SM-主体><SM-場所><SM-組織><BGH:駅/えき><文末><体言><用言:判><体言止><一文字漢字><レベル:C><区切:5-5><ID:（文末）><裸名詞><RID:1470><提題受:30><主節><定義文主辞><判定詞><名詞項候補><先行詞候補><正規化代表表記:駅/えき><用言代表表記:駅/えき><格要素-ガ:NIL><格要素-ヲ:NIL><格要素-ニ:NIL><格要素-ト:NIL><格要素-デ:NIL><格要素-カラ:NIL><格要素-ヨリ:NIL><格要素-マデ:NIL><格要素-ヘ:NIL><格要素-時間:NIL><格要素-外の関係:NIL><格要素-ノ:NIL><格要素-修飾:NIL><格要素-ガ２:NIL><格要素-トスル:NIL><格要素-ニトル:NIL><格要素-ニトモナウ:NIL><格要素-ニアワセル:NIL><格フレーム-ガ-主体><格フレーム-ヲ-主体><格フレーム-ニ-主体><格フレーム-デ-主体><格フレーム-カラ-主体><格フレーム-ヨリ-主体><格フレーム-マデ-主体><格フレーム-ノ-主体><格フレーム-修飾-主体><格フレーム-ガ２-主体><格フレーム-ニトル-主体><格フレーム-ガ-主体ｏｒ主体準><格フレーム-ガ２-主体ｏｒ主体準><格フレーム-ヲ-主体ｏｒ主体準><格フレーム-ニ-主体ｏｒ主体準><解析連格:ガ><格解析結果:駅/えき:判 0:ガ/U/-/-/-/-; ヲ/U/-/-/-/-; ニ/U/-/-/-/-; ト/U/-/-/-/-; デ/U/-/-/-/-; カラ/U/-/-/-/-; ヨリ/U/-/-/-/-; マデ/U/-/-/-/-; ヘ/U/-/-/-/-; 時間/U/-/-/-/-; 外の関係/U/-/-/-/-; ノ/U/-/-/-/-; 修飾/U/-/-/-/-; ガ２/U/-/-/-/-; トスル/U/-/-/-/-; ニトル/U/-/-/-/-; ニトモナウ/U/-/-/-/-; ニアワセル/U/-/-/-/->
駅 えき 駅 名詞 6 普通名詞 1 * 0 * 0 "漢字読み:音 地名末尾 カテゴリ:場所-施設 ドメイン:交通 代表表記:駅/えき" <漢字読み:音><地名末尾><カテゴリ:場所-施設><ドメイン:交通><代表表記:駅/えき><正規化代表表記:駅/えき><文末><表現文末><漢字><かな漢字><名詞相当語><自立><内容語><タグ単位始><文節始><文節主辞>
!! 3 -1D <見出し:駅>
! 3 <SYNID:駅/えき><スコア:1>
! 3 <SYNID:s2245:停車場/ていしゃば><スコア:0.99>
EOS
```

図 4: 「ホテルに一番近い駅」の解析結果

```
#!/usr/bin/env perl

# KNP::Result を使って SynGrpah 解析結果を読み込むサンプルプログラム

# usage: perl knp_syn.pl -s ホテルに一番近い駅 -relation | perl read-knp-result.pl

use strict;
use encoding 'euc-jp';
use KNP;

my $knp_buf;

while (<>) {
    $knp_buf .= $_;

    if (/^EOS$/) {
        my $knp_result = new KNP::Result($knp_buf);

        print $knp_result->all_dynamic, "\n";

        foreach my $tag ($knp_result->tag) {
            for my $synnodes ($tag->synnodes) {
                print $synnodes->tagid . ' ' . $synnodes->parent . $synnodes->dpndtype . ' ';
                print $synnodes->midasi . ' ' . $synnodes->feature . "\n";

                for my $synnode ($synnodes->synnode) {
                    print ' ' . $synnode->tagid . ' ' . $synnode->synid . ' ' . $synnode->score . "\n";
                }
            }
        }

        $knp_buf = '';
    }
}
```

図 5: read-knp-result.pl