係り受け関係と言い換え関係を用いた柔軟な日本語検索

竹内 淳平 † 辻井 潤一 §‡
† 東京大学理学部情報科学科 ‡CREST, 科学技術振興事業団
§ 東京大学大学院情報理工学系研究科コンピュータ科学専攻
{tj_jug, tsujii}@is.s.u-tokyo.ac.jp

1 はじめに

日本語の検索システムに，キーワード検索だけでなく単語間の係り受け関係を取り入れたシステムというのが情報検索 [1, 2]，質問応答 [3, 4] の分野ですでに提案されており，検索精度の向上が報告されている．しかし単純な係り受け構造のマッチングでは,「システムを利用する」というクエリに対し,「システムのサービスを利用する」や「私の利用したこの新しいシステムは…」という文は「システム」という文節の係り先が「利用する」ではないためにマッチングは失敗する．このような 係り受け構造が変化する言い換え(以下，**構造的言い換え**) を解消すれば検索精度がさらに上がることが期待される．これまで，より一般的な言い換えを解消するために，あらかじめ言い換え表現のペアを同義辞書に登録しておき質問拡張に用いる手法が研究されてきた [3, 4]．しかし一つの表現に対する言い換えの総数は，語彙の言い換えの数や構造的言い換えの数に比例して大きくなり，言い換え表現のペアが多いほど質問拡張にかかるコストは大きくなる．

本稿では，係り受けのマッチングを行う検索システムに対して，質問拡張のコストを減らすために，構造的言い換え情報を予め文書側に含ませインデクスを作成する手法を提案する．具体的には，係り受けグラフ構造に構造的言い換えを表す枝を予め付与しておき，それを用いた検索を行う．情報検索テストコレクション NTCIR-1[5] を対象とした実験を行い係り受け構造と提案構造との比較を行った．

2 構造的言い換えを含むグラフ構造

文に与えられる係り受け構造は，文節を点，係り受け関係を有向枝と見なすと，枝が葉から根に向かう木構造になる．枝 $u \to v$ は “u v” という係り受け表現がその文中に含まれていることを表している．点は文節内の内容語と対応し，付属語や活用は取り除く．また複合語も構成要素に分ける [4]．この係り受けグラフ構造に構造的言い換え表現を表す枝を追加することで，それらへのマッチングを可能にすることを考える．図 1 は元の文構造 (実線) に「システムを利用する」という枝 (点線) が加わった図である．

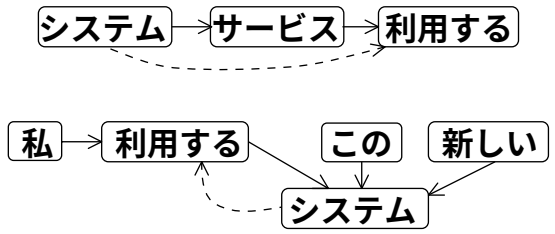

図 1: 言い換え表現に枝を追加する

我々は枝を追加するために，言い換えに関する研究 [6, 7] にあげられている構造的言い換えのうち，単純なパターンマッチングにより発見できるような

- 換喩，省略
  - システムのサービスを利用する
    → システムを利用する
  - 辞書による検索に対応 → 辞書に対応
- 複合名詞の換喩，複合名詞内の省略,
  - 行動パターンを調べる → 行動を調べる
  - 国際自然開発会議 → 国際会議
- 名詞化
  - システムを利用する → 利用したシステム
  - 犬が走る → 走っている犬

など 5 つのパターンに着目した．“換喩” の場合は，A, B, C を内容語，P, Q を格助詞として

$$A/P/B/Q/C \to A/Q/C$$

(例) システム/の/サービス/を/利用する
→ システム/を/利用する

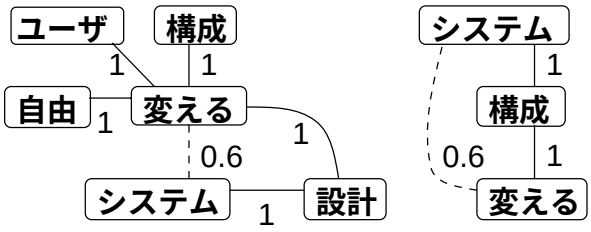

図 2: 類似度の計算

というパターンを見つけ枝を足していく．“名詞化” の場合では，全ての枝に反対方向の枝を追加する (グラフの無向化)．

しかしこの単純なパターンマッチングでは明らかに構造的言い換えにならない表現にも枝を足してしまう．

- 私の父を殴る → 私を殴る
- 犬についての研究を始める → 犬を始める

これを解消するため，枝には「言い換えとして適切である可能性」を値として重み付けする．具体的には

- 元の係り受け関係には最大値を与える．
- 新しく追加された枝には，規則に応じて一定割合で重みを減らす．

ということをする．この重み付けはスコア計算の時に反映され，言い換えにより追加された枝とのマッチングの方が係り受け関係の枝とのマッチングのときよりもスコアが低くなるようにする．なお一定割合というのはその規則から適切な言い換え表現が作られる可能性を表しており，本実験では前もって人手で設定した．

3 スコア計算

ランキングに用いられるスコアはクエリとテキスト文の構造の類似度とし，以下の式で定義した．

$$Sim = \frac{\text{文中にあるクエリと共通する枝の重みの総和}}{\text{クエリにある枝の重みの総和}}$$

例えば，テキスト文が「ユーザが構成を自由に変えられるシステム設計」でクエリが「システムの構成を変える」であるとする．図 2 はそれらを提案構造に変換させた図である．枝の重みの最大値は 1 とし，点線で表された追加枝は重みがそれよりも減っている．この場合，(構成–変える) と (システム–変える) という枝が一致しているのでスコアは

$$\frac{0.6 + 1.0}{1.0 + 1.0 + 0.6} = 0.62$$

となる．仮にテキスト文中にクエリと同じ文が現れたら，スコアは最大値の 1 となる．

4 システムの構成

このシステムの検索対象文書は生の日本語テキストであり，クエリは日本語の文またはその一部とする．システムは以下の手順で構成した．

1. 文書を解析して提案構造に変換しそれらの逆引きインデクスファイルを，キーワード情報のものと枝情報のものとで 2 種類作成する
2. 与えられたクエリを提案構造に変換する
3. クエリ内のキーワードを全て含む文書を検索する (この集合を結果集合として固定する)
4. 結果集合の文書毎にクエリとのスコアを求める

なお，形態素解析器として ChaSen[1]，係り受け解析器として CaboCha[2] を使用した．

5 実験と評価

5.1 概要

本提案システムの有効性を，情報検索用コーパスである NACSIS テストコレクション 1 (NTCIR 1) の J コレクション (日本語情報検索) を用いて評価した．NTCIR 1 は約 33 万件の論文抄録と 83 件の検索課題，各検索課題に対する正解文書からなる．論文抄録から，「論文タイトル」「抄録」の部分を抽出して作成した文書集合 (197MByte) を対象に検索を行った．検索課題はそれぞれ三通りの詳しさで日本語で書かれており，クエリの作成はそのうち正解判断に必要な最低限の記述力をもつ「DESCRIPTION」の項目に従って人手で行った．本検索システムは語彙レベルの言い換えには対応していないので，キーワード単位の言い換えは人手で行い OR 検索を行った．全 83 件から 45 件を無作為に選び抜きクエリを作ったところ，係り受け関係の精査が必要と思われるクエリ (Q_1) は 25 件，必要と思われない (単語のみで検索できてしまう) クエリ (Q_2) は 20 件であった．(Q_1) のクエリ例として「電波の人体への影響」，(Q_2) のクエリ例として「デジタル著作物」などがあった．各検索課題の正解集合には，適合 (A) または部分的適合 (B) の判定が付いており，A B 判定両方を正解集合とみなした．

[1] http://chasen.aist-nara.ac.jp/
[2] http://cl.aist-nara.ac.jp/ aku-ku/software/cabocha/

表 1: インデクスファイルサイズの比較 (KByte)

インデクス	キーワード	枝情報
係り受け構造のみ	39,366	94,256
提案構造	41,591	172,108

5.2 インデクスファイル

実験に先立ち，提案構造と係り受けのみのグラフ構造それぞれを逆引きインデクスファイルに変換した．それらのサイズを表 1 に示す．これらは圧縮されたサイズだが，提案構造は係り受け構造よりも，枝を追加したために枝情報のファイルサイズが約 2 倍大きくなってしまった．さらに提案構造では総インデクスファイルサイズが元の文書サイズを超えてしまうという結果になった．これらは今後の課題である．

5.3 検索精度の評価

5.3.1 評価手法

検索の精度はキーワード検索で抜き出した文書に対するランキングの精度として測定した．ランキングの精度は

- 上位 5 位，10 位，20 位の時点の精度
- R-precision [8](正解集合と同数出力した時点の精度)

の二つ (どちらも平均を取る) で評価した．クエリ集合として，

- $(Q_1)(Q_2)$ を使った場合
- (Q_1) のみを使った場合

を試した．

5.3.2 精度結果

評価結果を表 2 に示した．これを見ると全体的に本手法の構造の方が少し精度が上がっている．しかし検索結果を見ると単純な枝マッチングでは同スコアに並ぶものが多く，それにより高い精度が出ないものもあった．また，わずかな差しか出なかった原因として，提案したグラフ構造によって新たにマッチした文書が NTCIR 1 の判断基準によって大半が不正解に判定されていたことが考えられる．これらは構造的言い換えの出現程度では判定できないような深い意味解析のレベルでの判定を必要としていた．

表 2: 係り受けと本手法の精度比較

	Q_1 と Q_2		Q_1 のみ	
評価法 \	係り受け	提案構造	係り受け	提案構造
5 docs	0.5739	0.5652	0.4583	0.4833
10 docs	0.4478	0.4739	0.4167	0.4542
20 docs	0.3826	0.4174	0.3338	0.3458
R-pre	0.4363	0.4479	0.3437	0.3605

5.4 個々の検索結果の考察

そこで我々は次に本システムがどれだけ構造的言い換えの解消に成功し，同時にどれだけ間違いを起こしているのかを具体的に観察するために，人手で 7 クエリに対する検索結果 431 件を観察し，以下の表 3 の 6 パターンに分類した．

表 3: 検索結果のパターン分類 (件数)

本手法の構造に	マッチした	マッチせず
クエリと同じ係り受け構造がある	98 $(\star 1)$	2 $(\star 2)$
同じ係り受け構造はなく構造的言い換えがある	53 $(\star 3)$	14 $(\star 4)$
どちらもない	10 $(\star 5)$	254 $(\star 6)$

$(\star 1)$ $(\star 6)$ は係り受けマッチと同じ結果を示した部分である．検索の失敗と言える $(\star 2)$ $(\star 4)$ $(\star 5)$ に比べて，適切に構造的言い換えを検索できた $(\star 3)$ の方が多いという結果になった．

5.4.1 適切な構造的言い換えが検索された例

$(\star 3)$ の部分は，係り受け構造はないが構造的言い換えがあり，本手法で検索できたものである．具体例として

- 「シソーラスの自動構築」というクエリに
  - 自動的にシソーラスを構築
  - $\cdots$ を 自動的に構築したシソーラス
- 「電波の人体への影響」というクエリに
  - 電波の人体影響
  - 電波 曝露による局所 SAR の 人体影響
- 「連結グラフを求める」というクエリに
  - 有向 グラフのk-連結 成分を 求める
  - 最大共通 連結 部分 グラフを求める

などがあった．

5.4.2 係り受け解析の失敗

(⋆2) の部分は，係り受け解析の失敗により，クエリと同じ係り受け構造が文書に本来存在するはずだったのにマッチしなかったものである．

5.4.3 検索されなかった構造的言い換え

(⋆4) は今回の規則では対応できなかった構文的言い換え規則の存在によるものであった．

- 「シソーラスの作成」というクエリに
  - ··· を表す シソーラスについては，上位, 下位関係を中心に 作成された **もの**が多い
- 「連結グラフを求める」というクエリに
  - 部分 グラフが連結と **なる**ものを 求める

上の例は「シソーラス」が「もの」にかかり，下の例は「連結」が「なる」にかかってしまったものである．また，

- 「連結グラフを求める」というクエリに
  - 2連結グラフに対する **根を**中心とする生成木を 求める 線形時間アルゴリズム

上の例は,「グラフ」が「根」に係ると解析された.「生成木」と解析されれば,「グラフ」→「求める」が枝の追加されていたはずのところである．

5.4.4 不適切な検索例

構造的言い換えがないのに高いスコアを出した例 (⋆5) として，主に二つの理由があった．

不適切な枝の追加

- 「高速なアプリケーション」というクエリに
  - 高速 ネットワーク上の有効な アプリケーション
  - ··· の アプリケーションの 一つである 高速 データ通信

これらは単純なパターンマッチングの結果間違った言い換えの枝が追加されたためである．

格関係の除去

- 「シソーラスの作成」というクエリに
  - ···作成のためのシソーラス

今回のシステムでは格助詞を全て除去したため，係り受け関係の種類は特定できなかった．

6 終わりに

本稿では，係り受け関係が変化する構造的言い換えを解消するために，質問拡張としてではなく，文書側に言い換え情報を持たせる手法を提案した．その結果検索精度が上昇し，実際に多くの構造的言い換えを検索できていることが確認された．今回構造的言い換えに対する枝は単純なパターンマッチングによるものであり，それによる間違った枝の追加による不適切な表現への検索もいくつか確認された．これに対し，既に存在する言い換えデータベースや同義表現辞書を用いれば適切な枝の追加が可能であり，これによりインデクスファイルサイズの増加も抑えられるほか，実行時の質問拡張の数を減らすことができるはずである．

また，今回では構造的言い換え表現のみを扱っており，他の表現の揺れの解消技術が本手法の構造でも再現できるか，またどのように組み入れるかは重要な課題である．今後，これらの課題に取り組んでいく予定である．

参考文献

[1] 新美和彦, 兵藤安昭, 池田尚志. 係り受け情報を用いた全文検索とその評価. ディジタル図書館, No. 11.

[2] 立石健二, 大庭直行, 峯恒憲, 雨宮真人. 係り受け情報を利用した web 上の日本語テキスト検索システム. 研究報告「デジタル・ドキュメント」, No. 59, 1998.

[3] 高橋哲朗, 縄田浩三, 乾健太郎, 松本裕治. 質問応答における構文的照合と言い換えの効果. 言語処理学会第 9 回年次大会, 2003.

[4] Yoji Kiyota. *Dialog Navigator: A Navigation System from Vague Questions to Specific Answers based on Real-World Text Collections*. PhD thesis, Univ. Kyoto, 2004.

[5] National Center for Science Information Systems, editor. *Proceedings of the 1st NTCIR Workshop on Research in Japanese Text Retrieval and Term Recognition*, 1999.

[6] 乾健太郎. 言語表現を言い換える技術. 言語処理学会第 8 回年次大会チュートリアル, pp. 1–21, 2002.

[7] 乾健太郎, 藤田篤. 言い換え技術に関する研究動向. 自然言語処理, Vol. 11, No. 5, pp. 151–198, 10 2004.

[8] R. Baeza-Yates, editor. *Modern Information Retrieval*. Addison-Wesley, 1999.