青山学院大学　理工学部　情報テクノロジー学科

平成１５年度卒業論文要旨

質問応答のための質問文と知識文の間の意味ベースでの精密な照合方式

安部 建助(15800004)　竹原 一彰(15600048)　安田 智成(15800077)
原田研究室　　指導教員　原田 実 教授　　韓東力 助手

1. はじめに

日本語をベースとした質問応答の研究では、村田らが質問文と検索されたパッセージの係り受け木での対応語間に IDF や EDR の語彙による類似度を用いて、最も類似したパッセージから解を得ている[1]。この研究では意味解析を行っていないので語意の精度は非常に低く、また語間の関係も係り受け関係を用いているので深層的な意味に基づく重要度を把握できず、解答精度は 50%から 70%程度である。他の研究も同様で精度に問題がある。そこで、日本語文章の意味解析に基づき高精度の質問応答システムを構築するための基礎研究を行う。文章内容の形式表現方法を調査の結果、質問文と知識文（検索文）の精密な照合を行うために、Sowa の提案する概念グラフが最適だと判断した。質問文と知識文に対する概念グラフの類似度を計算することによって文章間の精密な照合を行うことができるようになる。

2. グラフの照合定理と質問応答の基本的考え方

照合の正しさは、「ある質問グラフ Q が真であるためには、真であることが確定している知識グラフ K(または K の部分グラフ)への特殊化の系列が存在することである」[2]という定理に基づく。この定理を言い換えれば、「Q が真であるためには Q を特殊化し(または特殊化しなくても)K の部分グラフと合同になればよい」となる。例えば、質問文「意味解析システムは何処で作られていますか。」と知識文「原田研究室で意味解析システム SAGE が開発されています。」を概念グラフに展開すると図 1のようになる。両グラフには点線で示したような特殊化が存在し、疑問詞に対応する箇所が解として抽出される。

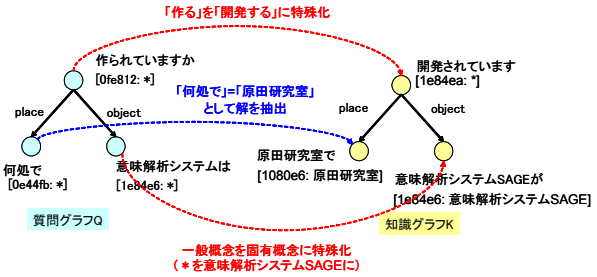

図 1 特殊化と解の抽出

しかし、一般には知識グラフに余分な修飾が含まれ合同を要求するには厳しすぎることがある。そこで、グラフ照合の際にノード飛び越しを許すことにし、この定理に反する緩和操作を数値的に反映するためにグラフ類似度を導入した。グラフ類似度はノード類似度とリレーション類似度の和で定義され（完全にグラフが合同になるとき最高値 200 となる）下式で計算する。

グラフ類似度 =ノード類似度＋リレーション類似度

$$\text{ノード類似度} = \frac{\sum \text{照合ノードペアの概念類似度}}{\text{質問グラフのノード数}} \times 100$$

$$\text{リレーション類似度} = \frac{\sum \text{照合ペア間のリレーション数}}{\text{質問グラフのリレーション数}} \times 100$$

3. グラフの照合アルゴリズム

アルゴリズムの主制御構造を図 2に、その中心となるグラフ照合アルゴリズムを図 3に示す。その他の各種関数の意味を表 1に示す。アルゴリズムの方針は質問グラフの主述語ノード qst からスタートして、質問グラフのノード qn を縦型に訪問しながら知識グラフのノード kn と照合していくというものである。また照合を行いながらグラフ類似度を計算する。

```
main(){
  kstList←getElm({kn | outdeg(qst)≦outdeg(kn)∧fill(qst,kn)});
  while(kst←pop(kstList)){
    bind(qst,kst);
    質問グラフをqstを根とする張る木に変換する;
    知識グラフをkstを根とする張る木に変換する;
    matching(qst,kst);
    if(リレーション類似度≧50)
      照合結果をファイルに書き出す;
  }
}
```

図 2 アルゴリズムの主制御構造

```
matching(qn, kn){
  nqnList←getElm(children(qn);
  dknList←getElm(descendants (kn));
  while(nqn←pop(nqnList)){
    while(dkn←pop(dknList)){
      if(rel(qn,nqn)=inrel(dkn)∧fill(nqn,dkn)){
        bind(nqn,dkn);
        matching(nqn,dkn);
        break;
      }
    }
    if(dknが偽){
      bind(nqn,φ);
      if(nqnが葉でない){
        dqnList←getElm(children(nqn))
        while(dqn←pop(dqnList)){
          rel(qn,nqn)をrel(qn,dqn)につなぎかえる;
          rel(nqn,dqn)をrel(qn,dqn)につなぎかえる;
        }
      }
      matching(qn,kn);
    }
  }
}
```

図 3 照合アルゴリズムの概要

表 1 アルゴリズムに用いた関数の説明

関数	意味
pop(List)	Listから次の要素を取り出す。なくなると偽
children(n)	nの未照合の子ノード順序集合を返す
descendants(n)	nの未照合の子孫の順序集合を返す
outdeg(n)	nの出次数を返す
bind(qn, kn)	qnとknを照合ペアとする(φとのペアは概念類似度0)
rel(n1, n2)	n1からn2へのリレーションを返す
inrel(n)	nへの入力辺のラベルを返す
getElm(S)	引数の順序集合の次の要素を返す。
fill(qn, kn)	ノード対qnとknが表2の条件を満たすなら真

表 2 ノード対の照合条件

	質問ノード	知識ノード
概念類似度	0.27以上(経験的に定めた値)	
リファレント	一般概念(*)	何でも良い
	固有概念	質問と同じリファレントを持つ
属性	同じ属性をもつ	

概念類似度 = $\max\left(\frac{2\times d_c}{d_q+d_k}\right)$ d_q,d_k : それぞれの概念の概念深さ

d_c : dq,dkの共通概念の概念深さ

3.1. アルゴリズムの開始点の決定

アルゴリズムの開始点となる質問グラフと知識グラフのペアを(qst,kst)とする。このとき qst は質問グラフの主述語ノードである。kst は qst より出次数が高く、表 2の条件を満たす知識ノードである。この処理は図 2中、2 行目に対応している。

3.2. 前処理

グラフの照合に先立ち、アルゴリズムの高速性や簡潔性のために前処理を行う。質問グラフの主述語ノード qst があり、qst と照合ペアとなる知識ノード kst を発見し、それぞれのグラフを qst、kst を根とする張る木に変換する。この処理は図 2の 5 行目から 7 行目に対応している。張る木に変換する際にアークが削除されるが、この削除されるアークは構文構造などを表すアークなので意味的に重要な情報は欠落しない。

3.3. ノード飛び越し

ノード飛び越しには質問ノードの飛び越しと知識ノードの飛び越しがある。知識ノードの飛び越しは既照合ペア(qn,kn)があるとき(qn は質問ノード、kn は知識ノード)、qn の次に訪問する質問ノード nqn を kn の直接の子ノード集合だけでなく、子孫全体を横型探索することによって実現する。質問ノードの飛び越しは、アークのつなぎ換えによって実現する。例えば、図 4に示すように、既照合ペア(qn,kn)があるとき qn の次に訪問する質問ノード nqn の照合ペアが kn の子孫集合を横型探索しても見出せなかった際、nqn への入力辺と出力辺を qn から nqn の子ノード集合のそれぞれへとつなぎかえる。この処理により nqn は飛び越される。

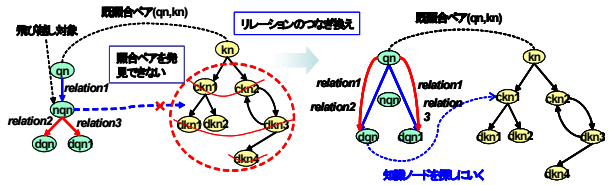

図 4 ノード飛び越し

4. Type Expansion

言語表現の多様性の差を吸収するために概念グラフの Type Expansion を行う。グラフ操作としては、深層関係を考慮しながら、ある概念を表現するノードを意味的に同値なグラフで置換する。簡単な例として、「メールを送る」を「メールを送信する」、「e メールを送る」などの同値な置き換えが可能である。このルールは現在人手で記述されているが国語辞典などから Type Expansion ルールの自動取得も視野にいれ、表 3 に示すフォーマットで、Type Expansion データベースに格納する。

表 3 Type Expansion の構成要素

条件グラフ	このグラフが部分グラフであるときType Expansionを適用できると判定する
定義グラフ	条件グラフに置き換わる同値なグラフ
コンストラクタ	置き換える際の、条件(接続関係など)を記述しておく

rules(G)を「グラフ G に適用できる Type Expansion のルール数」と定義すれば、このとき質問グラフ Q と知識グラフ群 K(j 個の知識グラフ K_1～K_j からなる)の照合はそれぞれのすべての展開グラフの組み合わせ $2^{rules(Q)}\times\sum_{i=1}^{j}2^{rules(K_i)}$ 通りのすべてについて行う(図 5)。

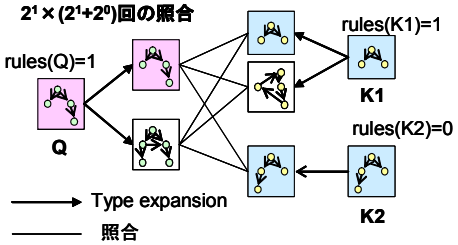

図 5 展開グラフとその照合

5. 解の抽出と表示

図 5のように照合が行われ、リレーション類似度が閾値 50 を超えた解がファイルに書き出される。そして抽出された解をグラフ類似度の順にソートし順位付で解を表示する。実際には図 6に示すように順位つき回答とその照合におけるグラフ類似度、解を抽出した知識文が表示される。

```
1st Answer = 黒質 : グラフ類似度 : 139.04

★ 根拠となる知識文 :
 --パーキンソン病は中脳の黒質にあるメラニン細胞が変性し
   黒質細胞内で作られる神経伝達物質のドーパミンがなくなり
   発病するとされている
-----------------------------------------
```

図 6 システム出力の解

6. 評価実験

英検 3 級のテキスト(日本語訳)などから what 型の質問だけを抽出し評価実験を行った。評価は質問文と知識文 1 つずつで 1 セットとし全部で 70 セット試した。その結果を表 4に示す。

表 4 実験結果

正解	誤回答	無回答
83% (58/70)	6% (4/70)	11% (8/70)

正解率 83%は現在報告されている他の質問応答システムの精度 50 から 70%を上回っている。この精度の高さは概念グラフにより従来の方法より文章内容を精密に照合できたため、また Type Expansion などのグラフの変形操作を意味解析レベルで行っているので正しい変形が行われたためと考えられる。また誤答率が少ないことも本手法は精密に照合が行えることを示している。

8. 総括と今後の課題

本研究において意味ベースの精密な照合方式を提案した。評価実験の結果から本照合方式の高い精密性が確認された。今後の課題として照合アルゴリズムの最適化や高速化があげられる。またすべての展開グラフを生成していた Type Expansion の適用方式の改善が挙げられる。各種閾値も詳細な実験を行って定める必要がある。実験結果において無回答率が高いことより、言語表現の差を吸収する Type Expansion ルールの整備や自動取得も重要なテーマになる。最後に、質問応答システム全体の作成も今後の大きな課題である。

参考文献

[1] 村田真樹、内山将夫、伊佐原均：“類似度に基づく推論を用いた質問応答システム”.

[2] John F. Sowa: “Conceptual Structures: Information Processing in Mind and Machine”, 1984.