実時間質問応答のための探索制御付き命題照合

志賀 正裕‡　太田 知宏†　藤畑 勝之†　公文 隆太郎††　森 辰則†
‡横浜国立大学 大学院 工学研究科　†横浜国立大学 大学院 環境情報学府　††横浜国立大学 工学部
E-mail: {shig,tomo_o,fujihata,kumon,mori}@forest.eis.ynu.ac.jp

1 はじめに

情報検索と情報抽出を融合した技術として、近年、質問応答システム (QA システム) が注目されている。QA システムとは、例えば「現在の日本の首相は誰ですか?」という自然言語による問に対して「小泉純一郎」と解答するというように、望みの情報を含む文書ではなく、文書から取り出したその情報自身を提示するものである。現在提案されている最も一般的な QA システムは，4W1H 型の質問文を受け付け、情報検索技術に基づくパッセージ検索と、固有表現抽出などの情報抽出を組み合わせて解を発見し、出力するものである [TRE99, TRE00]。

ここで，4W1H 型の質問における，解を分類すると次の 2 種類となる。一つめは,「誰 (who)」「何処 (where)」「何 (what)」「何時 (when)」に対する解で、人名、地名、企業名、製品名、日時などいわゆる固有名が主な解として期待される。二つめは,「どれくらい (how)」に対する解で、距離、時間などの数量表現である。

後者については、構文解析により、その数量が「どのような観点」による「何について」の数量であるのかを明らかにし、質問文と比較することで解を得る根拠とする方法が提案されている [山下 01]。しかし、前者については検索された文書内にある固有名について、その種類だけで解の根拠とした場合、解の推定の精度に問題がある。

そこで、固有表現の種類による制約の他に品詞情報や構文情報を用いた文照合により精度を向上する手法が提案されている [村田 00]。同手法では，解候補を含む文に対して質問文との類似度を計算し、固有表現の種類以外の情報も用いて解としての信頼性を求めている。しかし、同提案では解候補となる文書断片の全てに対し類似度を計算しており、コスト[1]が非常に高い。

そこで、本稿では命題照合の中でもコストの高いいくつかの処理に注目し、それらの処理を全ての解候補に対して実行するのではなく、解として可能性の高い候補から先に処理する最良優先探索を導入することを提案する。これにより、計算機資源と解の精度のトレードオフをバランスさせつつ、与えられた環境での実時間応答を目指す。

2 実時間質問応答システム

2.1 システムの概略

我々の提案する実時間応答システムの概略を図 1 に示す。基本には従来提案されている質問応答システムの構造を踏襲しているが、命題照合モジュールにおいて本論文で提案する機構が組み込まれる。本システムは主に 4 つのモジュールから構成されており、質問文解析モジュール、文書検索モジュール、パッセージ検索モジュール、そして命題照合モジュールに分類される。以下では、これらモジュールの働きを述べる。

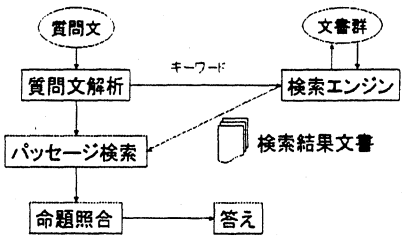

図 1: システムの全体図

2.2 質問文の解析

正解を得るための手がかりとして質問文から得られる情報は次の 2 種類である。一つめは、正解に関連の高い語句 (キーワード) である。質問文内に出現する語は文書中で正解の語とともに出現する可能性が高いと考えられるため、これをキーワードとして利用する。具体的には、文書検索では質問文中の名詞、パッセージ検索、命題照合では名詞に加えて動詞、形容詞の語幹などをキーワードとする。ただし、「こと」や「もの」など検索に不要と思われる語はあらかじめ除いておく。

二つめは、質問文が正解として求めている表現の種類 (質問文タイプ) である。これが人名であるのか、日付であるのか、あるいは金額であるのかといったことがわかれば、情報抽出の技術と組み合わせて正解を導くための大きな手がかりとなる。そこで、質問文の表層表現から質問文タイプを判定する。例えば、

勤労感謝の日は何月何日ですか。

という質問では、「何月何日ですか」から質問文タイプは date(日付) であると判定する。

2.3 文書検索

質問文解析により得られたキーワードを元に、文書検索を行う。文書検索は正解を含む可能性のある文書を網羅的に取り出すために行う。検索エンジンには、TFIDF による語の重みづけとベクトル空間法による類似度尺度を用いて、与えられたキーワード集合と各文書の類似度を求めるエンジンを採用した。

[1]以下、コストとは時間計算量の観点からのコストを指す。

2.4 関連パッセージ抽出

文書検索で得られた関連文書の中でも、質問の解となる情報が書かれているのはその一部だけである。これをそのまま命題照合 (2.5 節で後述) することはコストの面で非効率的なので、正解に関わる文脈を小さなコストで先に切り出しておいたほうがよい。これを行うのがパッセージ抽出である。パッセージとは、文章における連続した一部分のことであり、パッセージ抽出は文書集合から正解の語を含む可能性の高いパッセージを取り出すために行う。

パッセージ抽出で行なう処理は次のとおりである。数文の長さのウィンドウを設け、文書の文頭から 1 文ごとに走査する。ウィンドウ内の文章をパッセージとして取り出して質問文のキーワードの含有数を計算し、そのパッセージのスコアとする。また、質問文タイプに対応する表現、例えば質問文タイプが money(金額) であれば A 円といった表現がパッセージ内に現れた場合には、それが正解の語である可能性が高いと考えられるので、スコアを加算する。スコアを元にパッセージを順位付けし、上位のパッセージ、すなわち正解を含む可能性が高いと考えられるパッセージを出力する。

2.5 命題照合

このモジュールはパッセージ抽出で得られた検索結果 (複数の文) を入力とする。各解候補を仮に解として質問文の疑問詞部分と照合し、文中の他の部分も考慮した類似度から最終解を導き出す。ここで、各解候補とは助詞や記号等を除いた各形態素を指す。

一般に、精細な照合を行なう場合は構文解析した質問文と検索文のそれぞれの係り受け関係を比較して類似度を求める。さらに、固有表現抽出により固有名を同定することで、解の信頼度を高める。最終的に類似度の一番高い解候補が最終解として選ばれる。構文情報の照合においては、さらに表現の置き換え等を行ない、照合の可能性を広げたりする手法も提案されている。

しかし、詳細な照合を行ない、照合スコアの分解能や精度を上げるためには、コストの大きい解析 (形態素解析、構文解析、固有表現抽出) が必要となる。すなわち、照合スコアの分解能と処理コストはトレードオフの関係にある。

我々が現在採用している命題照合の手法は以下に述べるように基本的なものである。まず、質問文の疑問詞および各検索結果文の各解候補について、以下のような構文情報を得る。

1. 疑問詞 (解候補) があるキーワードに係るという係受け情報
2. あるキーワードが疑問詞 (解候補) に係るという係受け情報
3. 同じ述部に係る疑問詞 (解候補) とキーワードの情報とそれぞれの格の情報

次に、検索結果文中の各解候補を解と仮定した時の上記構文情報をそれぞれ得て、各解候補について質問文の構文情報との一致および近さを調べ、それをスコアとしている。

3 探索制御付き命題照合

3.1 命題照合のための最良優先探索

節 1 で述べているように、既存の命題照合にあるように各解候補に対してコストの高い計算を行なうという方法は非常に効率が悪い。そこで、コストの大きい解析を全ての解候補に対して行なうのではなく、最終解となる可能性の高い解候補から順番に処理を進めるような、最良優先探索 [太原 88] の導入を提案する。この探索を導入することで、最終解となる可能性の高い解候補ほど先に全ての処理を終えることになり、唯一の最終解を求めたいときは元より、複数の解を得たい場合にも、n-best 検索として有効に機能する。

ここで、命題照合における各処理を探索木にあてはめたものを図 2 に示す。

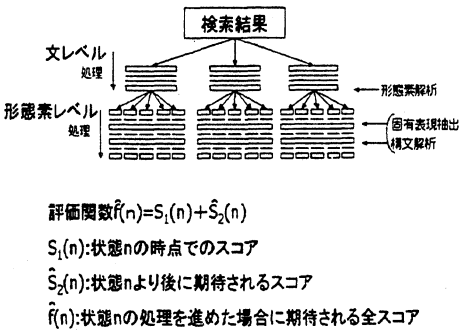

図 2: 命題照合における探索木

最良優先探索においては、ある状態 n における評価関数 $\hat{f}(n)$ はそこまでの処理で得られたスコア $S_1(n)$ と残り全ての処理に対するスコア $S_2(n)$ の推定値 $\hat{S}_2(n)$ との和で定義され、全ての状態のうち、評価関数 $\hat{f}(n)$ の値が大きいものから順番に処理を先に進める。ここで、推定値 $\hat{S}_2$ の求め方が重要な要素となる。

この探索におけるスコアとは、ある解候補を解と仮定した時、命題照合における質問文との類似度を数値化したものである。また、探索により最良解を必ず見つけるためには、式 (1) を満たす必要がある。

$$\hat{S}_2(n) \geq S_2(n) \tag{1}$$

3.2 照合スコアの推定

推定値 $\hat{S}_2(n)$ は、コストの大きい正確な処理をする前に、コストの小さい処理によって求めることに意味がある。一つの方法は、その時点より先にある全ての処理について、それぞれスコアの最大値を与えることである。この方法は、(1) 式を満たし、かつ、余分な処理が不要であるという利点があるが、スコアの推定精度が低いので見込みのない解の探索を余分に行ないがちである。そこで我々は、その時点で得られている情報、もしくは少ないコストで得られる情報を用いて、次の探索の展開については、より精度の高い近似解を

求めることを考える。以下では、各処理をする前に行なうスコアの推定について述べる。

3.2.1 形態素解析後の照合スコアを形態素解析の前に推定する手法

パッセージ抽出された文中の各形態素に対するスコアとしてまず以下の観点によるものがある。

1. 文中の解候補以外の箇所における質問文のキーワードの含有数
2. 文中の解候補以外の箇所における質問文のキーワード+助詞の含有数

上記の含有数が高い程、質問文と検索結果の文が類似した内容であると言える。これらは形態素解析を必要とする。そこで、推定スコアとして形態素解析によらない文字列照合のみで上記スコアを推定することを考える。推定スコアを求める観点は以下のようになり、スコア付与の対象となる解候補は各文字である。

1. 文中の解候補以外における質問文のキーワード (相当) の含有数
2. 文中の解候補以外における質問文のキーワード (相当)+助詞 (相当) の含有数

相当とは、文字列照合のみを用いているため、キーワードと同じ文字列が見つかったとしても、キーワードではない可能性があることを意味する。なお、この推定スコアは (1) 式を満たす。

3.2.2 構文解析後の照合スコアを構文解析の前に推定する手法

文を構文解析することで、文節の区切りと構文情報が得られる。この構文情報を元に、解候補周辺の係り受け関係の一致の度合を調べ構文レベルの照合スコアを求める。例として以下のようなものが挙げられる。

1. 解候補とこれに係っているキーワードとの距離
2. 解候補とこれを受けているキーワードとの距離
3. 解候補と同じ述語に係るキーワードの一致数

構文解析は形態素解析の情報を元に行なわれるので、形態素解析の次の段階として位置付けることができる。よって、構文解析の直前において最も詳細な解析が形態素解析の結果なので、推定スコアにこれを用いる。推定スコアは以下の観点で計算され、スコア付与の対象となる解候補は各形態素である。

1. 解候補とこれに係っているであろうキーワードとの距離[2]
2. 解候補とこれを受けているであろうキーワードとの距離
3. 解候補と同じ述語に係るであろうキーワードの一致数

係受け関係の推定は、それぞれの形態素の後で一番近くに現われる動詞を係り先としている。ただし、係受け関係の推定が誤っている場合があるので、この推定スコアは (1) 式を満たすとは限らない。

[2]形態素単位で測定

3.2.3 固有表現抽出後の照合スコアを固有表現抽出前に推定する手法

文から固有表現抽出をすることで、存在する固有表現の種類が得られる。固有表現抽出は形態素解析の情報を元に行なわれるので、形態素解析の次の段階として位置付けることができる。固有表現の種類の情報を得て、質問文のタイプと照合することで、解候補の適格性を判断できる。よってここでのスコアの導出は以下の観点で行なわれる。

1. 解候補に付与された固有表現の種類が質問文タイプと一致

スコアの推定はこの段階において最も詳細なレベルの解析が形態素解析の結果なので、これを用いる。推定スコアを求める観点は以下のようになり、スコア付与の対象となる解候補は各形態素である。

1. 解候補に付与された品詞細分類が質問文タイプと一致

品詞細分類で固有表現の種類の同定ができていない場合があるので、この推定スコア (1) 式を満たすとは限らない。

3.3 解の決定

上記全ての処理を終えた解候補のスコア $f(n)$ が他の全ての解候補の評価関数 $\hat{f}(n)$ よりも高い場合、最終解として取り出される。解候補は形態素単位であるため、出力する解を決定するにあたり、最終解を含む句の主辞を探し、これを主辞とする名詞句 (名詞連接) を解とする。第 2 位以降の解を求める時には引続き探索を行なう。

4 実験と考察

4.1 実験内容

本節では前節までに説明した我々の実時間質問応答システムに対する評価実験について述べる。

タスク定義は NTCIR Workshop 3 質問応答タスク [NTC01] のタスク 1 に準ずる。本実験では以下の値を評価した。

1. 各処理の段階における推定スコア $\hat{S}_2(n)$ および実スコア $S_2(n)$ の導出時間
2. 探索制御無しシステム…(a)

   探索を用いずにパッセージ検索結果の文全て (30 文) に対して全ての処理を実行した場合のスコア上位 5 件を解とした時の正解点数と実行時間。
3. 探索制御付きシステム…(b)

   探索を用いて上位 5 件の解が出たら終了とした時の正解点数と実行時間。

正解点数は最も高い優先順位を与えられた正解を基に順位の逆数として計算される。例えば、1 位なら 1 点、2 位なら 1/2 点、3 位なら 1/3 点となる。

実験に以下のものを用いた。

- 形態素解析器:JUMAN 3.61 [黒橋 98b]

- 構文解析器:KNP 2.0b6 [黒橋 98a]
- 固有表現抽出器:SVM を用いた固有表現抽出 [山田 01] を元に作成されたシステム
- 質問文:QAC Dryrun で用意された 50 個の質問文
- 対象テキスト (情報源):1998 年と 1999 年の毎日新聞の新聞記事

また、パッセージ抽出では文書検索の結果上位 100 件を採用し、命題照合ではパッセージ抽出の結果上位 10 パッセージ (30 文) を採用した。3.2.2 節と 3.2.3 節のスコア付与の例を表 1 に示す。なお、本システムでは

表 1: スコア付与の例

観点	スコア
解候補とこれに係るキーワードの距離が一致	2point
解候補と同じ述語に係るキーワードが一致	3point
解候補の固有表現の種類が質問文タイプと一致	4point

表 2: スコアと推定スコアの導出時間 (一回あたりの処理の平均)

	実スコア	推定スコア	比率
形態素解析	0.021 秒	0.0023 秒	0.11
構文解析	0.15 秒	0.0029 秒	0.019
固有表現抽出	11.93 秒	0.0003 秒	0.000025

表 3: システムの正解点数と実行時間 (1 問平均)

	正解点数	実行時間
探索制御有	0.30 点 (14.9/50)	303.1 秒 (15155/50)
探索制御無	0.29 点 (14.3/50)	34.3 秒 (1715/50)
比率	0.97	0.11

スクリプト語 Perl により大部分を実装しているので、絶対的な計算時間は長い。

4.2 実験結果

実スコアと推定スコアの導出時間を表 2 に、探索制御無しのシステム (a) と有りのシステム (b) の正解点数と実行時間を表 3 に示す。

4.3 考察

表 2 により、コストの大きな各処理よりも十分に短い時間でスコアの推定が行なわれていることが確かめられた。

表 3 を見ると、(b) では (a) に対して 1/9 以上の処理時間の短縮を実現している上に、システムの精度といえる正解点数はほぼ同じ値である。これにより我々の提案した探索制御付き命題照合が効率良く機能し、大幅なコスト削減が図れることを確認できた。

一部の問題では (a) の正解点数と (b) の正解点数とが異なる値を示した。これは、(b) の探索において (1) 式が満たされていない場合があるので、正解に至る経路の推定スコアが低く推定されてしまい、別の解候補が先に最終解として選ばれてしまったと言える。

5 関連研究

村田ら [村田 00] の質問応答システムの命題照合では、抽出したパッセージの全ての文節について、質問文のいずれかの文節に対応づけて、類似度を逐一計算するため、抽出された文が長くなればなるほどコストの大きな照合となる。

また、黒橋ら [黒橋 01] は、類義表現を結び付けるルール群を用意し、それらを共有メモリを通してボトムアップ的に適用することにより、強力な類義性判定システムを提案しているが、辞書の定義文の展開や上記ルールの適用に大きなコストがかかる。これら命題照合においては、全ての解候補に適用するにはコストが高いということが共通して言える。このような詳細な命題照合を行なう前に、我々の手法による探索制御を組み合わせれば、コストの削減が期待できる。

6 まとめ

本稿では、質問応答における命題照合に探索制御を用いることで、コストが大きな処理を全ての解候補に対して実行するのではなく、最終解となる可能性の高い候補から処理する方法を提案し、評価実験をもって全体のコスト削減につながることを確認した。これにより、計算機資源と解の精度のトレードオフをバランスさせつつ、与えられた環境での実時間応答を目指すことが可能となる。

今後の課題としては、QA システムとしての精度向上、構文解析後の命題照合の改善、数値情報抽出の導入、および、各スコア付与の最適化が考えられる。

参考文献

[NTC01] NTCIR Project. NTCIR Workshop3 質問応答タスク (QAC Homepage). http://www.nlp.cs.ritsumei.ac.jp/qac/, 2001.

[TRE99] TREC Project. *Proceedings of The Eighth Text Retreival Conference TREC 8*. http://trec.nist.gov/pubs/trec8/t8_proceedings.html, 1999.

[TRE00] TREC Project. *Proceedings of The Eighth Text Retreival Conference TREC 9*. http://trec.nist.gov/pubs/trec9/t9_proceedings.html, 2000.

[黒橋 01] 黒橋禎夫, 酒井康行. 日本語表現の柔軟な照合. 言語処理学会第 7 回年次大会発表論文集, pp. 343–346, 3 月 2001.

[村田 00] 村田真樹, 内山将夫, 井佐原均. 類似度に基づく推論を用いた質問応答システム. No. 2000-NL-135, 1 月 2000.

[山下 01] 山下竜之, 藤畑勝之, 角田久直, 志賀正裕, 森辰則. 質問応答システムにおける数量表現の取り扱い. 言語処理学会第 7 回年次大会発表論文集, 3 月 2001.

[山田 01] 山田寛康, 工藤拓, 松本裕治. Support vector machines を用いた日本語固有表現抽出. 情報処理学会研究報告 01-NL-142-17, 情報処理学会, 3 月 2001.

[黒橋 98a] 黒橋禎夫. 日本語構文解析システム KNP version 2.0b6 使用説明書. 京都大学大学院 情報学研究科, 1998.

[黒橋 98b] 黒橋禎夫, 長尾真. 日本語形態素解析システム JUMAN version 3.6 使用説明書. 京都大学大学院情報学研究科, 1998.

[太原 88] 太原育夫. 人口知能の基礎知識. 近代科学社, 1988.