F-5

災害時情報への質問応答システムの適用

Application of Question-Answering System to Disaster-related Information

風間 淳一†　Stijn De Saeger†　鳥澤 健太郎†　後藤 淳†　István Varga†
Jun'ichi Kazama, Stijn De Saeger, Kentaro Torisawa, Jun Goto and István Varga

1. はじめに

東日本大震災では，Twitter をはじめとした情報システムの情報交換手段としての可能性が強く認識された一方で，大量に飛び交う情報から迅速，正確に状況を把握をすることは大変難しいという教訓を得た．救援活動や復興支援においても，組織や個人間での情報共有が十分に行われず，多くの無駄や問題が生じている．我々は，こうした問題を解決するため，これまで培ってきた情報分析技術を用いて，より適切な状況把握・判断を行うための情報を提供する情報配信基盤技術の開発を始めている．我々は，この一環として，我々が開発する質問応答システム「一休」を災害時の情報に適用することを試みている．このシステムの目標は，例えば「宮城県で孤立しているのはどこ？」といった質問に回答することである． 本稿では現在開発中のシステムについて述べる．

2. 背景と質問応答システムの可能性

今回の震災において，正確な状況把握を阻んだ本質的な原因は，大量の情報を整理し，必要な人に必要な情報を届ける手段が十分に準備されていなかったということである．例えば，Twitter で発信される安否確認，救援要請のツイートは，明確な宛先が無く，救援を行う行政等に有効に届かなかった．一方で，大量に飛び交うデマも含めた未整理の情報に振り回された人も多く出た．

そうした中，Twitter では，ハッシュタグの使用に見られるように，ユーザが自ら適応して情報整理の手段を編み出そうとする行動も見られた．また，「sinsai.info」「Google パーソンファインダー」等のように，安否情報等の整理を人海戦術で行うという試みも行われた．これらの活動をサポートするため，ボランティアの研究者らにより，自然言語処理を用いて Twitter 上の安否情報を整理することを目指した「ANPI_NLP」の取り組みが行われたが，開発の速度や多数のボランティアの組織化には課題があったことが報告されている [1].

本稿では，我々の取り組みの第一歩として行っている，質問応答システムを災害時情報に適用する試みについて述べる．

質問応答システムは，「質問」で表されたユーザの情報ニーズに対し，通常の検索のように単なるドキュメントの列挙ではなく，「答え」を端的に列挙する．これは，迅速で正確な状況把握のための有力な手段の一つであると考えられる．また， 今回の震災で明らかになったことは，刻々と変わる災害の状況の中で，その都度，安否情報，支援物資情報等々と新たにシステムを構築するのでは，状況変化に十分に対応できないということである．さらに，我々が備えるべき次の災害が，今回の震災と同じである保証はない．その点で，トピックを限ることなく質問に回答をすることができる質問応答システムは，災害時にも有効であると考えられる．

以上をふまえると，「今回の震災と同様の状況に対応できる」ことを条件としつつ，質問応答システムの実用性を改善していくことが重要となる．現在我々は，質問応答システムとして「一休」を使用し，これを，東日本大震災に関連するツイートに対して適用して，災害時に重要となる点について改善を行うという方針で研究開発を行っている．

3. 質問応答システム「一休」

一休は，スマートフォンからの音声入力を備えた質問応答システムで，ユーザからの多様なトピックの質問に意外な回答まで含めて回答することができる [1].

一休では，入力の質問文は，2 つの名詞スロットを持つ肯定文の係り受けパターン（入力パターン）に変換される．また，入力パターンと言い換えの関係にあるパターン（言い換えパターン）が，De Saeger et al. [2]を元にした方法であらかじめ列挙されてデータベースとして格納されており（**言い換え DB**），さらに，元となる文書群（通常の一休では，Web 6 億ページの文書．今回の研究では東日本大震災関連のツイート）からは，2 つの名詞が共起する係り受けパターンがあらかじめ列挙されている（**係り受け DB**）．

質問への回答は，入力パターン，および言い換え DB から得られる言い換えパターンを，質問中の名詞でスロットを埋めた状態で，上記の係り受け DB にマッチさせ，空いているスロットを埋める名詞を回答として抽出することで行う．回答のランキングは，上記の言い換え時のスコアに基づく．

例えば，「宮城県で孤立しているのはどこ」という質問文は，「X で孤立しているのは Y」という入力パターンに変換される（正確には，質問文からはいくつかのルールにより複数の入力パターンが生成される．例えば，これに加えて，「Y は X で孤立している」なども生成される）．この入力パターンの言い換えパターンには，例えば「X で Y が孤立する」「X で Y から動けない」などがあり，係り受け DB に「宮城県で A 小学校が孤立する」「宮城県で B 小学校から動けない」といったデータがあるとすると，「A 小学校」「B 小学校」が回答として抽出される．

この例から分かる通り，一休では言い換え認識を用いて様々な表現の差を吸収して回答を抽出することができる．

†情報通信研究機構，National Institute of Information and Communications Technology

1. http://www2.nict.go.jp/univ-com/info_analysis/index.htm に「一休」の紹介ビデオがある．

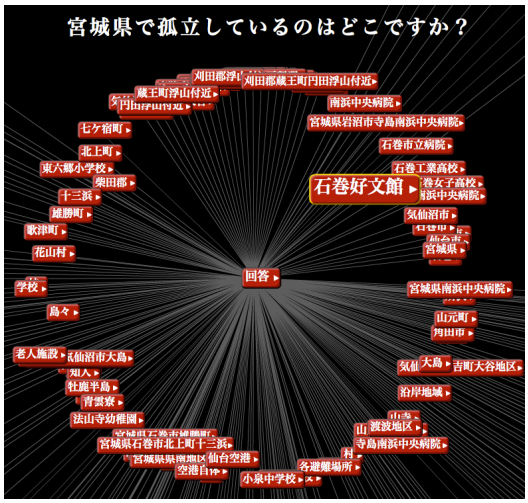


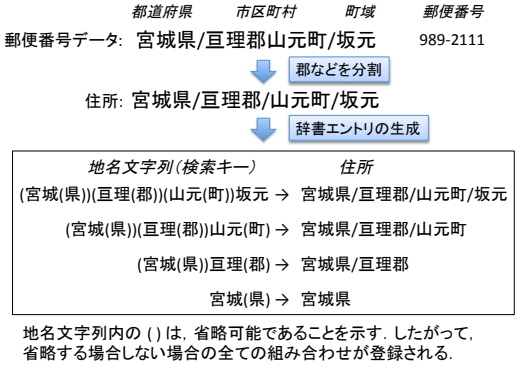
図 1：開発中の耐災害質問応答システムの動作の様子．「宮城県で孤立しているのはどこですか？」という質問に対して、「石巻好文館」や「東六郷小学校」などの回答が列挙されている．

4. 災害時情報への適用における問題点

災害時には，情報の取りこぼしが無いように再現率を保ちつつ適合率を向上させる必要がある．しかし，一休を適用する過程で，場所に関する質問に関して再現率を著しく低下させる次のような問題があることが明らかとなった．

- **場所の非明示性:** 一休をそのまま適用したのでは，「宮城県で孤立しているのはどこ」といった，震災時もまた通常時にも重要であると考えられる場所に関する質問にほとんど回答できない．前述したように一休は係り受けパターンを元に回答を抽出するが，イベントが起きた場所がイベントを表す動詞等に明示的には係らないことがツイートに限らず一般的に多いためである．
- **場所の包含性:** 場所には，暗黙の知識として包含性があり，それに対応した処理をしない場合再現率が低下する．例えば，ツイート中に「仙台市で…」と記述されていても，仙台市が宮城県の中にあることを正しく認識し，それを処理する手だてがなければ，「宮城県で…？」と問う質問には回答できないということが起きる．
- **場所の曖昧性:** 一部の地名は非常に大きな曖昧性を持ち，上記の包含性を扱おうとする場合に特に問題となる．例えば，「福島」という地名は日本全国に 50 以上もあり，そこから正しい一つを選ぶ必要がある．

これらの問題に対して，現在我々は簡単な対処方法を実装して解決をはかり，回答の量の増加と精度の向上を予備的な実験で確認している．図 1 は，本稿執筆時点でのシステムの動作の様子である．



地名文字列内の ( ) は，省略可能であることを示す．したがって，省略する場合しない場合の全ての組み合わせが登録される．

図 2：郵便番号データからの地名辞書の生成

5. 場所に関する問題への対処

5.1.地名・場所辞書の作成

まず，日本郵便が公開している郵便番号データを利用して，地名辞書を作成した．郵便番号データからは「都道府県/市区町村/町域」 で表される住所の情報を取り出すことができる．そこから，可能な地名文字列から住所へのマッピングを取り出す．ここで，可能な地名文字列とは，その住所を表し得る文字列であり，住所から簡単な方法により生成する（図２）．多少過剰生成となるが，場所の認識のカバレッジを上げるためには有用である．また，「都道府県/市区町村/町域」という住所の階層性は，先に挙げた場所の包含性に対処するための情報源となる．このようにして，2,486,545 個のエントリ持つ辞書（**地名辞書**）を得た（地名文字列-住所の対の数は 5,129,162）．そのうち 84,633 エントリが曖昧性のある地名だった．曖昧性がある場合の平均の曖昧性は 32.2，曖昧性の最大は 366 であった．例えば「福島」の曖昧性は 57 であった．

次に，我々が開発した上位下位関係抽出ツール（高度言語情報融合フォーラム ALAGIN から入手可能）を用いて Wikipedia から抽出した上位下位関係から，下位語が場所らしいものを取り出して利用した．これは，「学校」などの，郵便番号データには載っていないような場所にも住所へのマッピングを生成するためである．上位語が「(自治体名)の(*X)」というパターンにマッチする（X は「施設」「学校」など）場合に，下位語が場所らしいと判断した．下位語が上の基準で場所と判定された場合には，上位語中の自治体名を地名辞書で検索してマッチした住所を下位語の住所として付与する（他の細かいヒューリスティクスも用いているが，紙面の都合上，省略する）．最終的に，255,273 エントリを持つ辞書（**場所辞書**）を得た．

最後に，地名辞書と場所辞書をマージして**地名・場所辞書**とし，この辞書をトライに変換して効率的に接頭辞検索ができるようにして使用する．

5.2. 地名・場所の曖昧性解消

地名・場所辞書は曖昧性が大きく，正しく曖昧性解消できなければ，場所に関する正しい回答の抽出は不可能である．

将来的には本格的な開発が必要であるが，現時点では以下で示す単純な方法を用いている．この方法の目的は，他に何も情報が無ければ最も広範囲な地域を表す住所，直前に曖昧性解消された住所がある場合には，それと最も整合性のある住所を選ぶことである．

- **住所候補の生成:** ツイートの各文に対し地名・場所辞書で最左・最長一致検索を行い，名詞句に住所候補を付与する．その際，名詞句全体がマッチしない場合でも，名詞句の範囲内で最左のマッチを選び，できるだけ住所を付与する．なお，1 文字の地名・場所は誤ったマッチである可能性が大きいため，ここでは無視する．また，曖昧性を減らすため，住所は青森・岩手・宮城・福島・栃木・千葉各県のものだけに限定する．また，さらに固有表現認識器を用いて不要なマッチをさらに取り除くことも試みており，マッチした地名・場所が，固有表現認識で「場所」でも「組織」でもない場合に取り除く．しかし，現時点では固有表現認識を行うことによる明確な効果は確認されていない．
- **曖昧性解消:** ツイートの先頭から住所候補に対して以下の処理を行う．
  - **直前に曖昧性解消された住所がない場合:** 候補のうち，県・郡・市 （郡部の場合は町）部分が検索キーと一致するものを，この優先度で選ぶ．つまり，より広い地域レベルで検索キーと一致しているものを優先する．例えば「福島」の場合には，最も広範囲な「福島県」が選択される．該当するものが無ければ，列挙された最初の候補を選ぶ．候補は文字列の長さで昇順にソートしてあるので，近似的ではあるができるだけ広い地域レベルでの候補が選択される．
  - **直前に曖昧性解消された住所がある場合:** 直前で曖昧性解消された住所と県・郡・市（町）まで一致するものがあればそれを選択する．無ければ，県・郡まで一致するものを選択する．それも無ければ，県まで一致するものを選択する．該当が無ければ，曖昧性解消された住所がない場合と同じように候補を選ぶ．

5.3.イベントの場所の認識と補完

場所に関する質問に適切に答えるためには，地名・場所の曖昧性解消を行った後，文中の動詞などで表される各イベントが起きた/起きる場所を正しく認識し，係り受け DB の作成に反映させる必要がある．またこの際，場所の包含性も考慮する必要がある．

現在のシステムでは，「イベントの場所は直前に出現した地名・場所である」という仮定を置き，図 3 で示す方法を用いて元の文の係り受け解析結果を操作し，直前の地名・場所（ツイートが複数文の場合は前方の文も考慮する）の可能な文字列に場所を表す助詞「で」を加えたもの

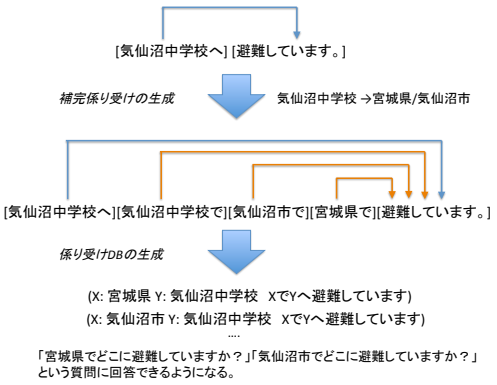


図 3: 補完係り受け解析の生成とその効果

を，イベントを表す動詞等に係るように付け加えた新たな係り受け解析結果（**補完係り受け解析**）を生成する．

例えば，図 3 のように「気仙沼中学校へ避難しています」という文があった場合，「避難」イベントの場所は，直前の場所である「気仙沼中学校」と認識され、さらに地名・場所辞書により「気仙沼中学校 →宮城県/気仙沼市」であると分かっているとすると，図に示すような補完操作が行われ、補完係り受け解析が生成される．元の係り受け解析の代わりに，この補完係り受け解析を係り受け DB 生成モジュールへ入力すれば，補完された場所に関連する質問にも対応した係り受け DB が生成される．

なお，元の文の係り受け解析にオープンソースで公開されている高速な日本語係り受け解析器 J.DepP（http://www.tkl.iis.u-tokyo.ac.jp/~ynaga/jdepp/）を使用している．

ここまでで述べたように，曖昧性の解消，補完係り受け解析の生成ともに，簡単なルールベースの手法であるが，ツイート中の前方の文まで考慮しているため，ある程度グローバルな処理と言うことができる．

5.4.場所・地名辞書による回答のフィルタリング

一般的に，質問が問うているものの種類を認識し，それに合致しない回答を除去することで質問応答の精度を向上させることができる．現在のシステムでは，質問が場所を問うている場合に，地名・場所辞書の情報を利用して，場所らしい回答のみを残すようにしている．一方，質問が場所を問うていない場合には，場所らしい回答を取り除くようにしている．場所かどうかの判定は次のように行っている．

- 回答の接頭辞が地名・場所辞書にマッチする場合は，場所であるとする（ただし，マッチの長さが 1 文字のときには、回答自体も一文字でなければならない）
- 地名・場所辞書にマッチしない場合，場所らしい接尾辞をもっていれば場所であるとする．この接尾辞は上位下位関係から場所辞書を抽出した際の上位語「X の Y」の Y の部分である（ただし，回答がこの接尾

辞と完全に一致する場合は除く．例えば「公共施設」などの抽象的すぎる回答が場所と認識されないようにするためである．）

6. システムの作成

ここまでで述べた方法を実装したシステムを，東日本大震災時のツイートのデータに対して適用した．元としたデータは，2011 年 3 月 10 日から 2011 年 4 月 4 日までに投稿された東日本大震災に関連する約 2.2 億件のツイート（提供元:（株）ホットリンク ）である．ここから，キーワードやハッシュタグなどを元にさらに絞り込んだ約 4,400 万ツイートを使用した．

元のツイートの係り受け解析結果の容量は合計で約 78.8GB であったが，補完係り受け解析を生成した場合には 93.5GB へと増加した．しかし，この増加量は極端に大きいわけではなく，その後の係り受け DB 生成の計算量への影響も押さえられている．

作成したシステムの動作の様子は，図 1 で示した通りであり．種々の実装上の工夫により，当初目的としていた場所に関連する質問に対して，ある程度の回答を得ることができるようになった．

7. 今後の課題

現在のシステムを実用とするには，さらに改良が必要であり，今後，次に挙げるような点で改良を行っていく予定である．

- 質問応答のエンジン部分について，我々は，１文中に回答がない場合でも機械学習や推論を用いて回答を抽出する技術 [3] [4] や，コアとなる言い換え・含意獲得の精度を向上させる技術[5]の開発を既に行っており，これらは再現率の向上に有効であると考えられる．また，現在は質問の形式がかなり限定されたものになっているが，複雑な質問文に対応するための技術の開発も現在行っている．質問が問うているものの種類にしたがって回答をフィルタリングする方法についても，場所に限定しない一般的な方法を開発中である．これらの最新の成果を取り込んだ新エンジンを用いることで，さらなる性能向上を目指す．
- 場所の取り扱いについては，機械学習などを利用したより高精度の手法を開発する．また，「時間」に関しても場所とほぼ同様の問題があるため，対処を行う必要があると考えている．
- ツイートに対する係り受け解析の精度も向上させる必要がある．特に，現在の一般的な係り受け解析器では，ツイートなどでよく見られる助詞などの省略により精度が低下しやすいため，これへの対処法を開発する予定である．
- システムの定量的な評価とチューニングに使用するため，質問と回答の組からなる評価データの作成を計画している．災害時に特に重要になると考えられる質問を３００種類程度選定し，各々の質問に対して，その回答を含むようなツイートを列挙し，さらに，そこから抽出されるべき回答をアノテートしたようなデータとする予定である．その際，システムの再現率のより正確な評価を可能とするため，列挙するツイートができるだけ多くなるような工夫を行う．なお，回答を作成する質問の選定の際に実際のニーズをよりよく反映するため，現在，被災地や支援を行った団体等へのヒアリングを並行して進めている．
- システムを実際に災害時に使用する場合には，刻々と発信されるツイートの情報を即座に質問応答システムへ反映させる必要がある．それを可能とするためのリアルタイムでの係り受け DB の更新などの機構の開発も行う予定である．

8. まとめ

本稿では，災害時に適切な状況把握・判断を行うための情報を提供する情報配信基盤技術の一環として，質問応答システム「一休」を災害時の情報に適用した試みについて述べた．今後さらに改良を行い，実用的なシステムとする予定である．

謝辞

本研究で利用しているデータは，(株)ホットリンク様よりご提供頂きました．ここに記して感謝致します．また，J.DepP に関して助言をいただいた吉永直樹氏に感謝いたします．

参考文献

[1] Graham Neubig, Yuichiroh Matsubayashi, Masato Hagiwara, and Koji Murakami. Safety information mining – what can NLP do in a disaster –. In Proceedings of IJCNLP 2011, 2011.

[2] Stijn De Saeger, Kentaro Torisawa, Jun'ichi Kazama, Kow Kuroda, and Masaki Murata. Large scale relation acquisition using class dependent patterns. In Proceedings of ICDM'09, 2009.

[3] Stijn De Saeger, Kentaro Torisawa, Masaaki Tsuchida, Jun'ichi Kazama, Chikara Hashimoto, Ichiro Yamada, Jong Hoon Oh, Ist- van Varga, and Yulan Yan. Relation acquisition using word classes and partial patterns,. In Proceedings of EMNLP 2011, 2011.

[4] Masaaki Tsuchida, Kentaro Torisawa, Stijn De Saeger, Jong Hoon Oh, Jun'ichi Kazama, Chikara Hashimoto, and Hayato Ohwada. Toward finding semantic relations not written in a single sentence: An inference method using auto-discovered rules. In Proceedings of IJCNLP 2011, 2011.

[5] Julien Kloetzer, Stijn De Saeger, Kentaro Torisawa, Motoki Sano, Jun Goto, Chikara Hashimoto, and Jong Hoon Oh. Supervised recognition of entailment between patterns. 言語処理学会第 18 回 年次大会, 2012.