CRM分野へ向けた日本語処理機能のミドルウェア化

佐藤 研治　池田 崇博　中田 貴之　長田 誠也

NEC　マルチメディア研究所

{k-satoh@da, t-ikeda@di, t-nakata@bk, s-osada@cd}.jp.nec.com

1. はじめに

ブロードバンドの普及により、企業間の連絡手段はもとより顧客と企業の間の連絡手段においても、電子メールやWebページ等を利用する形態が日常化し、電子テキストを利用してコミュニケーションが図られている。この状況を背景として、各種システムやソリューションに対し、検索やテキストマイニング等の言語処理アプリケーション(以下APと略す)を導入する事例が飛躍的に多くなって来ている。言語処理APを組み込んだシステム開発の増加により、その開発において必ずしも自然言語処理に精通していない人がシステム開発を担当しなければならないという状況が生じている。

従来の形態素解析エンジンは、自然言語処理に詳しい人がより多くの情報をテキストから抽出することを目的として開発されており、自然言語処理に詳しくない人にとっては必要十分な情報を簡便に利用可能なツールにはなっていない。検索やテキストマイニングAPを開発する際の言語解析出力に対する期待を整理すると下記3点になる。

(A)そのままインデクス作成や統計処理可能な文字列のみ出力
(B)同一表記でも意味が異なる表現を区別する付加属性情報
(C)異なる表記でも意味が同一視可能な表現をまとめる付加属性情報

(A)については、「形態素」を出力するのではなく処理の基本単位となる文字列のみ出力し、付属語等を出力しない処理が必要である。そこで、形態素毎の解析ではなく文節毎に解析結果出力することとし、必要な文字列として「文節の代表表記」を定めた。

(B),(C)については、各々文節内情報および文節外情報に着目し、それらの組み合わせ全てに対し付加属性情報を抽出し提供することとした。

○文節内情報
(B)否定と肯定等、付属語で付与される意味情報を抽象化し付加属性として利用可能にする
(C)形態素の品詞ではなく、文節の意味的な使われ方に対し品詞的な分類軸を与える
○文節外情報
(B)他の文節との組み合わせにより意味が変わる表現を区別する係り受け情報を提供する
(C)同義語・類義語を同一視する為の同義語処理機能を提供する

これらの課題を検討し、検索やテキストマイニングといった自立語の利用が主である言語処理APの開発者を対象として、そのまま利用可能な文字列を解析出力する日本語処理ミドルウェア(以下、日本語ミドルと略す)を開発した。本稿では、上記課題を日本語ミドルの設計・開発においてどのように解決したかを順に説明する。

2. 日本語処理ミドルウェア

日本語ミドルでは、言語処理APが必要とする文字列のみを出力することを目的として、入力テキストに対応する出力の基本形式を「文節毎に区切られた文節代表表記の並び」とした。ここでの文節認定は、自立語一語で一文節という従来のアルゴリズムがベースとなっている。文節代表表記は基本は文節内の中心自立語(以下ヘッドと呼ぶ)である。

日本語ミドルでは、文節毎の代表表記出力機能に加え、下記5種類の機能を用意することで、従来の形態素解析を自然言語処理にあまり詳しくない人が使う際の困難を解決した。

- 文節代表表記まとめ上げ処理
- 文節品詞の付与
- 付属語概念処理
- 係り受け解析
- 同義語処理

以下、これら機能について順に述べる。

2.1. 文節代表表記まとめ上げ

従来の言語処理APにおいては、接辞や数詞のまとめ上げ処理は形態素解析後にAP毎に処理されていた。しかしながら、まとめ上げ処理を行うには言語知識が必要となり、ルールや辞書を利用して処理する必要がある。これら知識処理は自然言語処理に詳しくない開発者にとっては、開発やメンテナンスが非常に困難である為、ミドルウェア内でまとめ上げ処理まで一貫して実行することで、AP開発者に出力文字列をそのまま利用させることを可能とした。

また、AP間で連携処理を行う場合は、まとめ上げ処理が異なるとAP間で受け渡される単語単位が異なり処理が正しく行われない。例えばテキストマイニングAPが「5時10分」を一単語として扱うまとめ上げ処理を持っている場合に、「5時」と「10分」をばらばらに扱う分類APをその前段として利用した場合は、分類するかしないかといったユーザの利用次第で出力結果が揺れてしまうという困った結果を引き起こすことになる。複数の言語処理APを連携して利用する為には単語のまとめ上げ処理を同一にする必要があり、日本語ミドルはその同一化機能を容易に提供している。

日本語ミドルにおける文節代表表記の認定処理では、接辞および数詞のまとめ上げ処理を行う。具体的なまとめ上げ処理の例を以下に示す。

- 接頭辞: 前＋首相 → 前首相
- 接尾辞: 対称＋性 → 対称性
- 数詞と助数詞:10＋名 → 10名、
  5＋時＋10＋分 → 5時10分

接辞等のまとめ上げをどこまで行うかについては、名詞連続を結合するかどうかといった観点も併せ、単語の単位の多様性をどのように扱うかといった研究がなされている[1,2]。これら研究は、単語の短単位や長単位を辞書で管理し、アプリケーションにより必要な辞書を切り替えて利用するという考え方をベースとしている。しかしながら、検索と分類を連携させて使う場面を想定すると、検索出力では分類APと同じ単語単位を用いたいが、検索インデクス内ではより短い単位でも検索可能にしておきたい為、1つの辞書・形態素処理エンジンで短単位および長単位を組み合わせて利用できる必要がある。

日本語ミドルでは、基本方針として接辞等はまとめ上げて出力するが、上記のような要求に応えられるよう、まとめ上げた単位の中に存在する短単位の自立語を、その文節の付属情報として得られるオプションを用意している。

名詞連続については、現在は文節を分けており、まとめ上げ処理はしていない。将来的にまとめ上げ処理の導入を検討している。

2.2. 文節品詞

形態素解析では、単語間の接続処理等を高精度に行う為、品詞をかなり多く用いる傾向がある。しかし、言語処理APではあまり品詞が多いとフィルタリングに用いるのが困難である。また、「美しさ」という文節はヘッドの品詞は形容詞であるが、用法としては名詞的に用いられるといった品詞の変化の問題もある。名詞に接頭辞として「無」や「不」等が結合すると形容動詞化する現象も同様である。

日本語ミドルではAP開発者がこれら個別の言語現象を対処しなくて済むように、代表表記に対し文節全体としての機能を表す品詞情報を振っている。これを文節品詞と呼ぶ。文節品詞は文節の機能としての役割を表す品詞である。実際の言語処理APでは大きな単語の括りが必要とされている為、現在は「名詞類」、「動詞類」、「形容詞類」、「その他」の4種類を定義している。

文節品詞は、基本的にその文節のヘッドの品詞に応じて選択する。ヘッドの品詞と文節品詞との対応の例を表 1に示す。代表表記のまとめ上げ処理の際に、文節の機能的用法が変化する場合について、まとめ上げ処理毎にまとめ上げ後の文節品詞を定義している。この処理により「美しさ」という文節には「形容詞類」では無く「名詞類」が付与される。

表 1: 文節ヘッドの品詞と文節品詞との対応例

文節品詞	ヘッドの品詞
名詞類	名詞、サ変の名詞的用法、等
動詞類	動詞、サ変の動詞的用法、等
形容詞類	形容詞、形容動詞、等
その他	副詞、その他

2.3. 付属語概念処理

テキストマイニングや分類等のAPでは、例えば「書く」と「書かない」といった付属語が異なる文節に対し、同一視する処理と区別する処理をデータや目的により切り替えることが可能になると、より詳細な分析や分類が可能となる。こういった機能を提供

する為に、日本語ミドルでは、助動詞等の付属語によって表される概念(付属語概念)を複数定義し、文節の代表表記としての単語に加え、それに付随する付属語概念を付加情報として提供する。

付属語概念情報は、文節毎にその概念が文節に存在するかしないかのフラグとして与えている。このフラグのON／OFFを無視するか区別して用いるかによって、付属語が異なる文節に対し同一視処理か区別しての処理かを切り替えて利用可能である。日本語ミドルが提供している付属語概念の例を表 2に示す。

異なる概念を表す複数の付属語が付随する場合、解析結果として、対応する付属語概念すべてを提供する。付属語概念を含めた解析結果の例を、表3に示す。

表 2: 日本語ミドルが提供する付属語概念の例

付属語概念	付属語の例	文節の例
否定	ない、ず	書かない
容易	やすい	書きやすい
困難	にくい	書きにくい
完了	た	書いた
進行	ている	書いている

表 3: 日本語ミドルによる付属語概念処理の例

文	解析結果
その車は美しくなかった。	その 車 美しい(否定)(完了) 。
その車は美しくありません。	その 車 美しい(否定) 。

表3の結果に対し、否定フラグだけ利用し完了フラグを無視して処理すれば両者は同一視され、完了フラグまで考慮して処理すると両者は異なる結果となる。

なお、日本語ミドルでは、文節の代表表記とそれに付随する付属語概念セットから、表示用の標準的な代表表記文字列を再生成する機能も提供している。例えば、「読む」＋(否定)＋(容易)という入力から、「読みやすくない」という代表表記を再生成する。これにより、同じ意味を表す付属語を同一の付属語概念にまとめ上げてテキスト処理を行った後、AP利用者向けに、その付属語概念を含む分かりやすい表記を提示することが可能となる。

2.4. 係り受け解析

検索やマイニングAPにおいて、単語単体の出現性だけでは、テキストに記述されている内容を的確にとらえられない場合がある。例えば、「デザインが良い。」という文と「色が良い。」という文を比較すると、どちらにも「良い」という単語が出現するが、「良い」の対象は、前者が「デザイン」であるのに対し、後者は「色」であり、「良い」という単語は共通でもそれぞれの文の内容は異なっている。このような差異を区別して処理する為には、文全体の解析木は必要でなく、単語間の係り受け情報のみが得られれば十分である。

現在は、係り受け関係を出力するには構文解析を利用する必要があるが、構文解析は処理時間がかかる為、こうした目的に対しては係り受け情報の代替として、1文内あるいは隣接する数単語内での単語共起情報を利用することが多かった[3]。

日本語ミドルでは、従来の構文解析処理で利用される文法を用いずに、高速に係り受け解析を行い、文節間の係り受けペア情報を提供する。解析手法としては、文節に対し係り属性(連用、連体)と受け属性(被連用、被連体)を付与し、それらの近接の対応関係を組み合わせることによって係り受け関係にある文節ペアを決定する。この処理により、線形オーダーの高速な解析処理を実現している。また、係り受けペア情報が不要なAPを考慮し、オプションにより解析をスキップできる。

例として、「家の中からネコの鳴き声が聞こえる。」という入力に対して係り受け解析を行った結果を表4に示す。

表 4: 日本語ミドルによる係り受け解析の例

文節(代表表記)	係り先文節
家	中
中	聞こえる
ネコ	鳴き声
鳴き声	聞こえる
聞こえる	
。	

2.5. 同義語処理

多くの言語処理APでは、同義や類義の単語を同一視したり、短縮形や異表記等の差異を吸収する為に同義語処理を備えている。この同義語処理

は、一般に文字列一致により検出しその文字列範囲を置き換えたり、名詞のみ置き換え可能であるといった、単純な置き換えのみが可能な場合が多い。これらの同義語処理では、置き換え後の助動詞概念処理や文節品詞の設定が行えないといった問題がある。

このため、日本語ミドルにおける同義語処理では、代表語(同義語置換後の単語)に対し品詞を与えるとともに、助動詞概念や係り受けを含む文節に対して、同義語ルールの設定を可能とした。例えば、「似合わない」(すなわち「似合う」+否定概念)を「不似合い」という代表語に置換することもできるように、同義語ルールには、付属語概念も含めて記述が可能である。ルール中に付属語概念が指定されている語がある場合、その概念も含めてマッチしたときのみ認定する。

同義語辞書の例(表 5)と、その辞書に従って同義語処理を行った解析例(表 6)を示す。表 6の例のように、元の語に付属語概念が付与されている場合には、代表語にも付属語概念が付与される。付属語概念込みで同義語ルールが定義されている場合には、ルールにマッチしなかった付属語概念が最終的な解析結果に残る。

表 5: 同義語辞書の例

代表語	同義語ルール
不似合い	似合う(否定)

表 6: 日本語ミドルによる同義語処理の例

例文	解析結果
その車は彼には似合わないだろう。	その 車 彼 似合う(否定)(推量) 。 不似合い(推量)

3. 考察

従来の形態素解析に、代表表記抽出処理、付属語概念処理、係り受け解析、同義語処理を統合し、共通の API を設定したことで、自然言語処理にあまり詳しくないAP開発者でも、出力される文字列をそのまま利用し、提供される付加属性情報を活用し、比較的容易に言語処理APを開発することが可能となった。また、システムやソリューション内の複数の言語処理APの基盤を日本語ミドルで共通化することで、以下の3点の効果が得られた。

- 複数の言語処理APで扱う単語単位が揃いAP間の連携処理が適切に行われる
- 同義語処理等の共通処理をAP毎に開発する必要が無く開発期間・コストが低減される
- 複数のAPを含むソリューションでユーザ辞書・同義語辞書が共通化されることで、辞書メンテナンス期間・コストが低減される

4. おわりに

検索やテキストマイニングAPからの言語解析出力に対する期待を整理し課題を抽出することで、AP開発者に比較的容易に言語処理APを開発することを可能とする日本語処理ミドルウェアの設計および開発を行った。日本語ミドルを活用することで、言語処理AP間の連携動作の実現および開発コスト・期間の削減という効果が得られた。

今後の改良として、入力テキストに名詞連続が存在した場合に、複数文節をまとめ上げることで名詞連続を1つの代表表記にすることが考えられる。名詞連続をまとめ上げる方が良いかまとめ上げない方が良いかは、言語処理APの適応領域毎に異なってくると考えられる為、まとめ上げのレベルを設定したり、まとめ上げた代表表記の元の形態素の組み合わせもオプションにより利用可能にするといった何らかの工夫が必要になると思われる。

また、今後日本語ミドルの高速化も検討している。現在、大規模なカスタマサポートセンターでは、顧客からの問い合わせメールが1日に1万通程も送られる状況にある。これは3ヶ月程度のメールを分類・分析する為には100万通のメールを処理する必要があることを示しており、日本語処理ミドルの高速化が必要である。方針としては、各々の文毎に別々のCPU で処理する並列化および分散化による高速化を検討している。

参考文献

[1] 小椋秀樹: 話し言葉コーパスの単位認定基準について,「話し言葉の科学と工学」ワークショップ講演予行集, pp. 21-28, (2001)

[2] 浅原正幸, 他: 語長変換を考慮したコーパス管理システム, 情報処理学会論文誌, Vol43, No.7, (2002)

[3] 山田剛一, 他: 複合語マッチングと共起情報を併用する情報検索, 情報処理学会論文誌, Vol39, No8, (1998)