MedInput: 病名の自動予測補完による医療テキスト入力支援ツールの構築

矢野 憲　　岩尾 友秀　　荒牧 英治
奈良先端科学技術大学院大学
{yanoken, iwao, aramaki}@is.naist.jp

1 はじめに

文書作成は、パソコンの登場当初からその用途の上位を占めている．8 ビット CPU を搭載した当時のパソコンは非力だったため、日本語入力は単漢字変換や熟語変換が一般的であり、長い文章を入力するのは骨の折れる作業だった．実用的なワープロソフトが普及するのは 16 ビット CPU を搭載したパソコンが登場してからのことになる．これまで，様々な日本語 FEP(Front End Processor) が現れては，消えていった．現在の IME(Input Method Editor) は人工知能を一部取り入れた構造をしており，直前に利用した変換候補を優先的にサジェストするなど，使いやすくなってきている．しかし，医療の現場などで作成される文書の日本語入力に関しては，非文法的な表現，低頻度の複雑な複合名詞，省略型の多用などの特徴があるため，既存の入力方法だけでは，ユーザーに過度に負担を強いることも少なくない．このため，本稿では，病名など複雑な固有名詞を多く含む医療文書作成を支援する知的なテキスト入力支援ツールの構築を行ったので報告する．

2 医療向けテキスト入力

医療分野では，普段使用しない複雑な固有名詞が頻繁に表れ，それらの「平仮名」から「漢字」への仮名漢字変換に時間がかかるという問題があった．電子カルテなどで省略語などがよく用いられるのは，こういった要因がある．

このため，ジャストシステムの ATOK などでは，医療向けの辞書を提供している[1]．また，現在広く利用されている代表的な IME には，ユーザー独自の仮名漢字変換辞書を登録することができるようなっている．例えば、図 2 は Baidu IME へ病名固有名詞の仮名読みとその漢字変換をユーザー辞書として登録した例を示している．

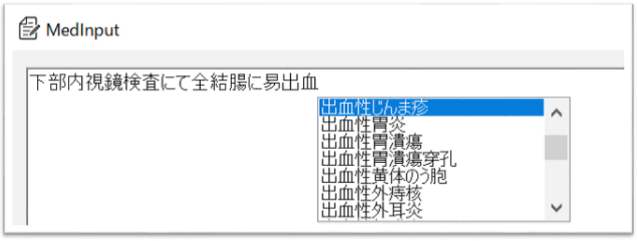

図 1: 病名のサジェスト表示による入力支援

しかし，ユーザー辞書を利用した場合には、その漢字変換候補がどんなアプリケーションのテキスト入力時にも候補として表示されるという問題がある．例えば，医療と無関係な文書作成中にこのような不要な病名に関する変換候補が表示されるのは好ましくない．医療用と一般用とで複数の IME を使い分けることも考えられるが、 IME はできるだけ使い慣れたものを使用したいというのが一般的である．

本稿で述べるテキスト入力支援ツール (図 1) は，IME による仮名漢字変換後の入力された単語，文字の接頭辞から推測される病名，症状名をサジェストし，提示された選択肢から項目を選ぶことで病名の入力を補完する機能を提供する．サジェストの表示は，IME での仮名漢字変換後に行うため，基本的にどの IME とも共存して利用可能である．提案するツールはテキストエディタとしての利用を想定しており，編集後にテキスト文書として保存したり，クリップボード経由で他のアプリケーションへデータ転送する利用形態を想定している．

なお，提案手法は，Microsoft Visual Studio の統合開発環境等で使用されている変数名，関数名，メソッド名の自動補完システム（IntelliSense) から着想を得て，開発を行った．

[1] http://www.justsystems.com/jp/products/dic_iryo/

図 2: Baidu IME へのユーザー辞書登録

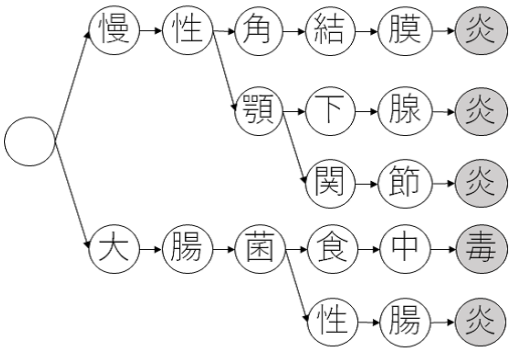

図 3: トライによる病名辞書の構築

3 辞書の構築

辞書作成のソースデータは病名の一覧が記載されたテキストファイルである．病名の一覧データは MEDIS（医療情報システム開発センター）から提供されているものを用いた[2]．このテキストデータからトライ構造 [2] による文字ベースの辞書を構築した．トライは，辞書に登録する各見出し語の共通接頭辞を併合することにより構築される木構造である．病名検索では，トライの根節点から葉節点に向かって，検索文字列の各文字を先頭から 1 回たどるだけで，入力文字列の先頭から始まる全ての接頭辞を探索することができる．従ってトライ全体に格納されている見出し語つまり病名の総数に関係なく，検索文字列の長さに比例した計算時間で検索が終了する．

例として，以下の 5 つ病名があった場合，図 3 のようにトライ辞書に登録される．

慢性角結膜炎
慢性顎下腺炎
慢性顎関節炎
大腸菌食中毒
大腸菌性腸炎

図 3 で、背景が灰色のノードは病名の末尾であること示している．表 1 にトライ構築に用いた病名辞書のサイズと構築されたトライの第 1 層のノード数，全ノード数第 1 層ツリーの平均ノード数，第 1 層ツリーの平均深度を示した．

病名数	25403
病名修飾語数	2277
トライ第 1 層のノード数	1057
トライ全ノード数	109975
第 1 層ツリーの平均ノード数	104
第 1 層ツリーの平均深度	3.68

表 1: 病名の辞書サイズとトライのノード数

4 入力サジェストの提示

病名のサジェストは，図 1 に示すリストボックスウィンドウがポップアップで表示される．ユーザーはリストから，入力しようとしている病名があればそれを選択することで，その該当部分に選択された病名が挿入される．該当がないの場合は Esc キーにより，ポップアップを閉じることができる．なお，ポップアップはモードレスで動作しているため，ポップアップ表示中も文字入力を継続して行うことが可能である．

4.1 入力サジェスト表示のタイミング

入力サジェストの表示はテキストが更新される毎に判断を行う．判断には，キャレットの直前にある文字列を取得して，その文字列をキーワードとして，トライ辞書の検索を行う．たとえば先ほどの例のトライ辞書があって，キーワードが「慢性」の場合，検索結果は，

慢性角結膜炎
慢性顎下腺炎
慢性顎関節炎

の 3 つの病名がヒットし，キーワードが「大腸」の場合には，

大腸菌食中毒
大腸菌性腸炎

の 2 つの病名がヒットする．

しかし，文字（単語）が入力される毎に，辞書検索を行いヒットがあった場合にサジェストを行うことに

[2]http://www.medis.jp/

は問題がある．例えば，辞書には以下のように「た」で始まる病名が含まれているが，入力文が　「〜した」など動詞の末尾で終了した場合などに，最後の助動詞“た”を病名の開始と判断し，以下の病名がヒットしてしまうという課題がある．

たこつぼ型心筋症
たこ壺型心筋障害

このように，文法上，入力された文字（単語）が病名の可能性が低い場合，入力サジェストの提示を抑止する必要がある．

5　病名固有名詞の判定

入力サジェスト提示の判断を行うため，提案手法は，既存の文字ベースの病名抽出ツール [3] を利用する．つまり，キーワードの検索を行う前に，該当文字（単語）が含まれる一文を抽出し，それを病名抽出ツールに適応し，キーワードが固有名詞となる可能性を判定する．1 文の抽出はキャレットの位置から前後方向に文の区切り文字（‘. ’, ‘。’, ’:’, ’;’ ）を検出することで行った。ここで抽出する 1 文は完結した文ではなく，入力途中の文の場合もある．文が完結していない場合，病名抽出ツールは，“病名”と予想される文字列に対して，できるだけ正しい IOB のタグ付けを行うように推論する．

例えば，文頭で以下のように完結していない文を病名抽出ツールにかけると以下のようなタグ付けを行う．

```
胃 B-病名
癌 O
が O
```

胃癌は病名の始まりだと正しく予想できているが，文が未完結であるため病名の終了を正しく認識できていない．もちろん，文が完結に近い場合，例えば文末に表れる助動詞「た」などが病名の一部でないことは，病名抽出の結果から直ちに判断可能になる．文が未完結のために病名の終了を正しく検出できない課題は，未完結の文からの病名抽出を学習することである程度改善することが予想される．また，病名抽出ではなく形態素解析による品詞情報を取得することも有効である．

Algorithm 1 に，病名の入力補完サジェストの表示制御アルゴリズムを示した．なお，キャレットとは入力文字の挿入位置を示す縦棒「|」である．

Algorithm 1 入力サジェストの表示制御アルゴリズム

```
フレームワークよりテキスト入力バッファの変更通知イベントを受信
Sent ← キャレットが含まれる文の抽出
抽出された Sent から病名抽出および形態素解析を行う
char ← キャレットの直前の文字
if char が病名の一部 then
  char（=Sent[k]) を末尾とした前方最大 5 文字目からの文字列をキーワードとして検索を行う
  for i = −4 to 0 do
    Keyword ← Sent[k + i : k]
    Candidates = SearchTrieTree(Keyword)
    if Candiates ≠ null then
      Candidates を候補としてポップアップで入力サジェスト表示する
      for-loop から抜ける
    end if
  end for
else
  入力サジェスト表示を抑止
end if
```

キーワードの探索をキャレットの前方から何文字目から始めるかはパラメータとして設定される．上記のアルゴリズムではキーワードの最大文字列長を 5 と設定している．ユーザーが候補リストの中から項目を選択して実行キーが押された場合，キーワードをその候補で置き換える．Esc キーが押された場合は，入力サジェストを閉じる．なお，候補リストが表示されている間に，ユーザーは継続して入力を行うことが可能であり，新しい単語（文字）が入力される毎に，上記のアルゴリズムに従って，入力サジェストの表示判断を行い，候補リストを更新する．

6　病名の一括抽出と校正支援

病名抽出を入力された文章全体に対して適応することで，文章中に含まれる全ての病名の確認と，病名の標準化を行うことができる．図 4 に病名の一括抽出と事実性の判定を行った例を示す．抽出された陽性の病名は赤で，陰性の病名は青で表示される．処理方法としは，初めに，文章を文に分割し，1 文ごとに前述した病名抽出ツール [3] により病名を抽出している．抽出された病名は同時に自動的に陽性か陰性の極性判断

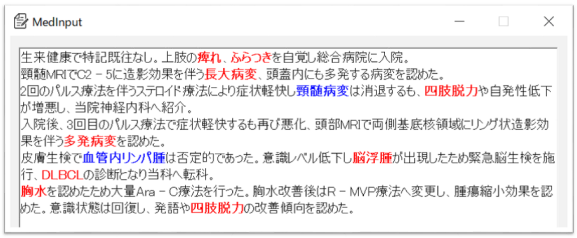

図 4: 病名の一括抽出と事実性の判定

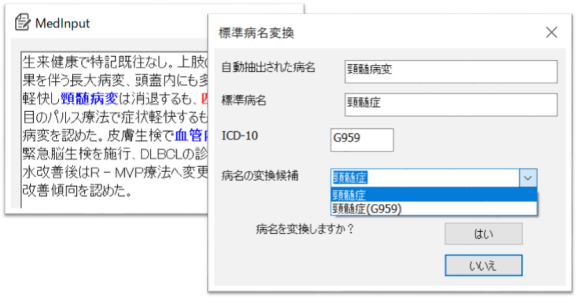

図 5: 病名の変換. 青字で表示されている病名「頚髄病変」をマウスの右クリックで押下すると病名変換ダイアログが表示される.

がなされる．なお，病名の陽性，陰性の判断基準の詳細については，文献 [4][1] に詳しく述べている．

6.1 病名の変換

マウス操作により，抽出された赤字または青字の病名を右クリックすると，その病名に対する標準名および ICD-10 コードが表示される (図 5)．ユーザーはここでサジェストされた標準病名に置き換えることが可能である．病名抽出で抽出された病名が辞書に登録されたどの病名とも一致しなかった場合は，サジェストは表示されない．

7 考察

病名をサジェストする際，頻度については考慮しなかったが，トライのノード情報として頻度情報を追加することで，高頻度の病名から優先的に提示することも可能である．病名の頻度情報は，すでにある病名のマスター辞書から得るか，もしくはアプリケーションの起動中にユーザーが選択した病名の履歴を保存しておき，それを頻度情報として利用する方法が考えられる．また，提案した入力支援ツールの評価方法については今後の課題である．

8 まとめ

本稿では医療分野におけるテキスト入力支援ツールについてに述べた．病名のような複雑な固有名詞の入力はユーザーにかなりの負担を強いることも少なくない．提案したような入力支援ツールを利用することで，医療従事者の文書作成の労力を軽減することが可能になると考えている．医療文書作成における病名入力を簡素化することで，医療テキストの質・量ともに改善が期待でき，そのデータを 2 次活用する場合においても利益になるであろう．

なお，使用した文字ベースの病名抽出ツール [3] は文字ベースの双方向 LSTM を用いて学習したモデルからネットワークの重みパラメータだけを抽出し，C++ による学習済推論モデルとして実装しているため，病名抽出の処理では深層学習のフレームワークを必要としない。このために、MedInput は、単体の Windows プログラムとして実行可能である．なお，本稿で紹介した医療向け入力支援ツール (MedInput) を将来的に一般に利用できる形で公開する予定である．

謝辞

この研究の一部は日本学術振興会補助金番号 JP16H06395 および 16H06399，ならびに厚生労働科学研究費補助金番号 28030301 によって支援された．

参考文献

[1] 荒牧英治, 若宮翔子, 矢野憲, 永井宥之, 岡久太郎, 伊藤薫. 病名アノテーションが付与された医療テキストコーパスの構築. 自然言語処理「言語処理の応用システム」特集号, Vol. 25, No. 1, 2017.

[2] 徳永拓之. 日本語入力を支える技術. 技術評論社, 2012.

[3] 矢野憲, 伊藤薫, 若宮翔子, 荒牧英治. 深層学習による医療テキストからの固有表現抽出器の開発とその性能評価. 第 31 回人工知能学会全国大会. 人工知能学会, 2017.

[4] 矢野憲, 若宮翔子, 荒牧英治. 医療テキスト解析のための事実性判定と融合した病名表現認識器. 第 23 回言語処理学会年次大会. 言語処理学会, 2017.