

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4833476号
(P4833476)

(45) 発行日 平成23年12月7日 (2011. 12. 7)

(24) 登録日 平成23年9月30日 (2011. 9. 30)

(51) Int. Cl.

F I

G 0 6 F 17/22 (2006. 01)

G 0 6 F 17/22 5 0 2 G

G 0 6 F 17/21 (2006. 01)

G 0 6 F 17/21 5 9 2 F

G 0 6 F 17/21 5 9 2 J

請求項の数 33 (全 25 頁)

(21) 出願番号 特願2001-536715 (P2001-536715)
 (86) (22) 出願日 平成12年10月13日 (2000. 10. 13)
 (65) 公表番号 特表2003-527676 (P2003-527676A)
 (43) 公表日 平成15年9月16日 (2003. 9. 16)
 (86) 国際出願番号 PCT/US2000/028418
 (87) 国際公開番号 W02001/035249
 (87) 国際公開日 平成13年5月17日 (2001. 5. 17)
 審査請求日 平成19年10月15日 (2007. 10. 15)
 (31) 優先権主張番号 60/163, 902
 (32) 優先日 平成11年11月5日 (1999. 11. 5)
 (33) 優先権主張国 米国 (US)
 (31) 優先権主張番号 09/606, 807
 (32) 優先日 平成12年6月28日 (2000. 6. 28)
 (33) 優先権主張国 米国 (US)

前置審査

(73) 特許権者 500046438
 マイクロソフト コーポレーション
 アメリカ合衆国 ワシントン州 9805
 2-6399 レッドモンド ワン マイ
 クロソフト ウェイ
 (74) 代理人 110001243
 特許業務法人 谷・阿部特許事務所
 (74) 復代理人 100115624
 弁理士 濱中 淳宏
 (74) 復代理人 100115635
 弁理士 窪田 郁大
 (72) 発明者 カイーフ リー
 アメリカ合衆国 98072 ワシントン
 州 ウッディンビル 214 ウェイ ノ
 ース イースト 14233

最終頁に続く

(54) 【発明の名称】 モードレス入力で一方のテキスト形式を他方のテキスト形式に変換する言語入力アーキテクチャ

(57) 【特許請求の範囲】

【請求項 1】

入力文字列に基づいて1つまたは複数の区分からなる妥当な第1の言語のタイピング候補を生成し、各タイピング候補に対して、前記入力文字列が前記第1の言語のタイピング候補の間違いである可能性の確率を測定するための第1のタイピングモデルと、前記入力文字列に基づいて1つまたは複数の区分からなる妥当な第2の言語のタイピング候補を生成し、各タイピング候補に対して、前記入力文字列が前記第2の言語のタイピング候補の間違いである可能性の確率を測定するための第2のタイピングモデルと、言語モデルとを記憶した記憶装置を備えたコンピュータにおいて実行する方法であって、前記コンピュータのプロセッサが、

前記第1の言語と前記第2の言語とを含む入力文字列を受け取るステップと、

前記記憶装置に記憶された第1のタイピングモデルおよび第2のタイピングモデルを用いて、前記受け取られた入力文字列が前記第1の言語のタイピング候補または前記第2の言語のタイピング候補の間違いである可能性の確率を測定するステップと、

前記記憶装置に記憶された言語モデルを使用して、前記測定された確率がより高いタイピング候補の区分を選択して、前記言語の各々について言語テキストの1つまたは複数の候補文字列を生成し、生成された各候補文字列が使用される確率を測定するステップと、

各候補文字列に対して測定された確率に基づいて、前記入力文字列を置き換えるのに使用できる前記候補文字列を決定するステップと

を備えることを特徴とする方法。

【請求項 2】

(1) 前記選択されたタイピング候補が前記第 1 の言語の文字列である場合、前記タイピング候補を前記第 1 の言語の変換文字列に変換すること、および (2) 前記選択されたタイピング候補が前記第 2 の言語の文字列である場合、前記タイピング候補をそのまま残すこと、を実行することをさらに備えることを特徴とする請求項 1 に記載の方法。

【請求項 3】

前記第 1 の言語は主言語であり、前記第 2 の言語は前記主言語ほど頻繁に使用されない二次言語であることを特徴とする請求項 1 に記載の方法。

【請求項 4】

前記第 1 の言語の前記入力文字列は表音文字テキストを含み、前記第 2 の言語の前記入力文字列は非表音文字テキストを含むことを特徴とする請求項 1 に記載の方法。

10

【請求項 5】

前記第 1 の言語は中国語であり、前記第 2 の言語は英語であることを特徴する請求項 1 に記載の方法。

【請求項 6】

入力文字列に基づいて 1 つまたは複数の区分からなる妥当な第 1 の言語のタイピング候補を生成し、各タイピング候補に対して、前記入力文字列が前記第 1 の言語のタイピング候補の間違いである可能性の確率を測定するための第 1 のタイピングモデルと、前記入力文字列に基づいて 1 つまたは複数の区分からなる妥当な第 2 の言語のタイピング候補を生成し、各タイピング候補に対して、前記入力文字列が前記第 2 の言語のタイピング候補の間違いである可能性の確率を測定するための第 2 の言語に合わせて構成された第 2 のタイピングモデルと、言語モデルとを記憶した記憶装置を備えたコンピュータにおいて実行する方法であって、前記コンピュータのプロセッサが、

20

第 1 の言語および第 2 の言語の入力モードを切り替えることなく前記第 1 の言語と前記第 2 の言語とを含む入力文字列の入力を受け取るステップと、

前記記憶装置に記憶された第 1 のタイピングモデルおよび第 2 のタイピングモデルを用いて、前記受け取られた入力文字列が前記第 1 の言語のタイピング候補または前記第 2 の言語のタイピング候補の間違いである可能性の確率を測定するステップと、

前記記憶装置に記憶された言語モデルを使用して、前記第 1 の言語および前記第 2 の言語のタイピング候補の区分を選択して 1 つまたは複数の候補文字列を生成し、生成された各候補文字列が使用される確率に基づいて、前記入力文字列を置き換えるのに使用できる候補文字列を決定するステップと、

30

(1) 前記候補文字列の中の第 1 の言語に対応するタイピング候補を前記第 1 の言語の出力文字列に変換して前記出力文字列を出力すること、および (2) 前記候補文字列の中の第 2 の言語に対応するタイピング候補を出力すること、を実行することを備えることを特徴とする方法。

【請求項 7】

前記第 1 の言語と前記第 2 の言語とを含む前記入力文字列を単一の編集行に表示し、

前記単一の編集行に前記出力文字列または前記入力文字列を選択して表示することをさらに備えることを特徴とする請求項 6 に記載の方法。

40

【請求項 8】

前記第 1 の言語は中国語であり、前記第 2 の言語は中国語以外の言語であることを特徴する請求項 6 に記載の方法。

【請求項 9】

言語入力装置であって、

表音文字テキストと非表音文字テキストとの組み合わせで書かれた入力文字列を受け取るユーザインタフェースと、

前記入力文字列が、表音文字テキストで書かれた第 1 のタイピング候補の間違いである可能性を示す第 1 の確率を測定し、該測定された第 1 の確率が、前記入力文字列が非表音文字テキストで書かれた第 2 のタイピング候補の間違いである可能性を示す第 2 の確率よ

50

り高い場合に、前記入力文字列を置換できる第1のタイピング候補を出力するための第1のタイピングモデルであって、前記第1のタイピング候補はそれぞれ前記入力文字列から可能性のある間違いを修正した文字列を生成し、前記修正した文字列をセグメント化することにより生成される第1のタイピングモデルと、

前記第2の確率を測定し、該測定された第2の確率が、前記第1の確率より高い場合に、前記入力文字列を置換できる第2のタイピング候補を出力するための第2のタイピングモデルであって、前記第2のタイピング候補はそれぞれ前記入力文字列における可能性のある間違いを修正することにより文字列を生成し、前記修正した文字列をセグメント化することにより生成される第2のタイピングモデルと、

前記第1のタイピング候補について言語テキストで書かれた可能な変換文字列を供給する言語モデルと、

10

(1) 前記入力文字列の中の表音文字テキストを前記変換文字列のいずれか1つに変換すること、および(2) 前記入力文字列の中の非表音文字テキストを変換せずそのまま保持すること、を実行するように構成されている検索エンジンと

を備えることを特徴とする言語入力装置。

【請求項10】

前記第1の確率が前記第2の確率よりも高いときに、前記検索エンジンは前記入力文字列を前記変換文字列のいずれか1つに変換することを特徴とする請求項9に記載の言語入力装置。

【請求項11】

20

前記第1の確率が前記第2の確率よりも低いときに、前記検索エンジンは複数の前記第2の候補文字列のいずれか1つを出力することを特徴とする請求項9に記載の言語入力装置。

【請求項12】

前記表音文字テキストは第1の言語であり、前記非表音文字テキストは第2の言語であることを特徴とする請求項9に記載の言語入力装置。

【請求項13】

前記表音文字テキストはピンインであり、前記非表音文字テキストは英語であることを特徴とする請求項9に記載の言語入力装置。

【請求項14】

30

入力文字列に基づいて1つまたは複数の区分からなる妥当な第1の言語のタイピング候補を生成し、各タイピング候補に対して、前記入力文字列が前記タイピング候補の間違ひである可能性の確率を測定するための第1のタイピングモデルと、前記入力文字列に基づいて1つまたは複数の区分からなる妥当な第2の言語のタイピング候補を生成し、各タイピング候補に対して、前記入力文字列が前記タイピング候補の間違ひである可能性の確率を測定するための第2のタイピングモデルとを記憶した記憶装置を備えたコンピュータにおいて実行する方法であって、前記コンピュータのプロセッサが、

第1の言語と第2の言語とを含む入力文字列を受け取るステップと、

前記記憶装置に記憶された第1のタイピングモデルおよび第2のタイピングモデルを用いて、前記受け取られた入力文字列が前記第1の言語のタイピング候補または前記第2の言語のタイピング候補の間違ひである可能性の確率を測定するステップと、

40

前記入力文字列が前記第1の言語を含む第1のタイピング候補の間違ひである可能性を示す第1の確率に基づいて、前記入力文字列を置き換えるのに使用できる前記タイピング候補の区分を選択して、前記第1のタイピング候補を決定するステップと、

前記入力文字列が前記第2の言語を含む第2のタイピング候補の間違ひである可能性を示す第2の確率に基づいて、前記入力文字列を置き換えるのに使用できる1つまたは複数の前記タイピング候補を選択して、前記第2のタイピング候補を決定するステップと、

前記第1の確率が前記第2の確率よりも高い場合に、前記第1のタイピング候補を変換文字列に変換して出力文字列を導出し、

前記第1の確率が前記第2の確率よりも低い場合に、前記第2のタイピング候補を変換

50

せず出力文字列を導出することを備えることを特徴とする方法。

【請求項 1 5】

前記第 1 の言語は主言語であり、前記第 2 の言語は前記主言語ほど頻繁に使用されない二次言語であることを特徴とする請求項 1 4 に記載の方法。

【請求項 1 6】

前記第 1 の言語の前記入力文字列は表音文字テキストを含み、前記第 2 の言語の前記入力文字列は非表音文字テキストを含むことを特徴とする請求項 1 4 に記載の方法。

【請求項 1 7】

前記第 1 の言語は中国語であり、前記第 2 の言語は英語であることを特徴する請求項 1 4 に記載の方法。

【請求項 1 8】

前記入力文字列は中国語ピンインと英語との組み合わせであり、前記出力文字列は中国語漢字と英語との組み合わせであることを特徴とする請求項 1 4 に記載の方法。

【請求項 1 9】

前記第 1 および前記第 2 のタイピング候補をデータベースから取得することをさらに備えることを特徴とする請求項 1 4 に記載の方法。

【請求項 2 0】

前記第 1 の言語のトレーニングテキストを入力した複数のユーザから集めたデータをもとに、前記第 1 の確率を導出し、

前記第 2 の言語のトレーニングテキストを入力した複数のユーザから集めたデータをもとに、前記第 2 の確率を導出することをさらに備えることを特徴とする請求項 1 4 に記載の方法。

【請求項 2 1】

ユーザが入力している前記入力文字列と同じ行に前記出力文字列を表示することをさらに備えることを特徴とする請求項 1 4 に記載の方法。

【請求項 2 2】

プロセッサ上で実行すると請求項 1 4 に記載の方法を実行させるためのプログラムを記憶したことを特徴とする 1 つまたは複数のコンピュータ読み取り可能媒体。

【請求項 2 3】

入力文字列を受け取り、前記入力文字列から可能性のある間違いを修正した文字列を生成し、前記入力文字列および前記修正した文字列を、異なる区分を有するタイピング候補にセグメント化し、前記入力文字列が該セグメント化されたタイピング候補の間違いである可能性を示す第 1 のタイピング誤りの確率を決定する第 1 のタイピングモデルと、

前記入力文字列を受け取り、前記入力文字列における可能性のある間違いを修正することにより文字列を生成し、前記入力文字列および前記修正した文字列を、異なる区分を有するタイピング候補にセグメント化し、前記入力文字列が該セグメント化されたタイピング候補の間違いである可能性を示す第 2 のタイピング誤りの確率を決定する第 2 のタイピングモデルと、

前記第 1 および前記第 2 のタイピング誤りの確率のそれぞれに基づいて、前記セグメント化された複数のタイピング候補のいずれか 1 つを選択する検索エンジンと

を備えることを特徴とする言語入力装置。

【請求項 2 4】

前記第 1 のタイピングモデルは第 1 の言語を使用してトレーニングされ、前記第 2 のタイピングモデルは第 2 の言語を使用してトレーニングされることを特徴とする請求項 2 3 に記載の言語入力装置。

【請求項 2 5】

前記入力文字列は表音文字テキストと非表音文字テキストとを含み、前記第 1 のタイピングモデルは前記表音文字テキストに対しトレーニングされ、前記第 2 のタイピングモデルは前記非表音文字テキストに対してトレーニングされることを特徴とする請求項 2 3 に記載の言語入力装置。

10

20

30

40

50

【請求項 2 6】

前記第 1 のタイピングモデルは中国語を使用してトレーニングされ、前記第 2 のタイピングモデルは英語を使用してトレーニングされることを特徴とする請求項 2 3 に記載の言語入力装置。

【請求項 2 7】

前記入力文字列はピンインと英語とを含み、前記第 1 のタイピングモデルはピンインに対しトレーニングされ、前記第 2 のタイピングモデルは英語に対してトレーニングされることを特徴とする請求項 2 3 に記載の言語入力装置。

【請求項 2 8】

前記選択されたタイピング候補の出力文字列を供給する言語モデルをさらに備える特徴とする請求項 2 3 に記載の言語入力装置。

10

【請求項 2 9】

前記検索エンジンは前記入力文字列を前記出力文字列に変換することを特徴とする請求項 2 8 に記載の言語入力装置。

【請求項 3 0】

前記入力文字列を受け取り、前記出力文字列を共通の編集行に表示するユーザインタフェースをさらに備えることを特徴とする請求項 2 8 に記載の言語入力装置。

【請求項 3 1】

コンピュータを請求項 2 3 に記載の言語入力装置として機能させることを特徴とするプログラム。

20

【請求項 3 2】

入力文字列に基づいて妥当な第 1 の言語のタイピング候補を生成し、各タイピング候補に対して、前記入力文字列が第 1 の言語のタイピング候補の間違いである可能性の確率を測定するための第 1 のタイピングモデルと、前記入力文字列に基づいて妥当な第 2 の言語のタイピング候補を生成し、各タイピング候補に対して、前記入力文字列が第 2 の言語のタイピング候補の間違いである可能性の確率を測定するための第 2 のタイピングモデルとを記憶した記憶装置を備えたコンピュータのプロセッサで実行すると、コンピュータは、

第 1 の言語および第 2 の言語の入力モードを切り替えることなく前記第 1 の言語と前記第 2 の言語とを含む入力文字列の入力を可能にし、

前記記憶装置に記憶された第 1 のタイピングモデルおよび第 2 のタイピングモデルを使用して、前記第 1 の言語のタイピング候補および前記第 2 の言語のタイピング候補のそれぞれが間違って前記入力文字列のように入力された可能性を示す確率を測定し、該測定された確率がより高いタイピング候補から、前記入力文字列を置き換えるのに使用できる少なくとも 1 つのタイピング候補を決定し、前記コンピュータは、

30

(1) 前記入力文字列が前記第 1 の言語のタイピング候補の間違いである可能性を示す確率が前記第 2 の言語のタイピング候補の間違いである可能性を示す確率より高い場合、前記入力文字列を前記第 1 の言語の出力文字列に変換して前記出力文字列に出力することを実行し、(2) 前記入力文字列が前記第 1 の言語のタイピング候補の間違いである可能性を示す確率が前記第 2 の言語のタイピング候補の間違いである可能性を示す確率より低い場合、前記第 2 の言語の前記入力文字列を出力することを実行するためのプログラムを格納したことを特徴とする 1 つまたは複数のコンピュータ読み取り可能媒体。

40

【請求項 3 3】

プロセッサで実行すると、入力文字列に基づいて表音文字テキストで記載された 1 または複数の区分からなる妥当な第 1 の言語のタイピング候補を生成し、各タイピング候補に対して、前記入力文字列が前記第 1 の言語のタイピング候補の間違いである可能性の第 1 の確率を測定するための第 1 のタイピングモデルと、前記入力文字列に基づいて非表音文字テキストで記載された 1 または複数の区分からなる妥当な第 2 の言語のタイピング候補を生成し、各タイピング候補に対して、前記入力文字列が前記タイピング候補の間違いである可能性の第 2 の確率を測定するための第 2 のタイピングモデルと、言語モデルとを記憶した記憶装置を備えたコンピュータが、

50

表音文字テキストと非表音文字テキストとを含む入力文字列を受け取るステップと、
前記記憶装置に記憶された第1のタイピングモデルおよび第2のタイピングモデルに基づいて、前記第1の確率および前記第2の確率を測定するステップと、

前記記憶装置に記憶された言語モデルを使用して、前記タイピング候補の区分を選択して、1つまたは複数の候補文字列を生成するステップであって、

前記入力文字列を前記第1の言語のタイピング候補に置き換えるのに使用できる、前記表音文字テキストで書かれた第1の候補文字列を決定するステップと、

前記入力文字列を前記第2の言語のタイピング候補に置き換えるのに使用できる、前記非表音文字テキストで書かれた第2の候補文字列を決定するステップと

を含むステップと、

10

前記表音文字テキストで書かれた前記第1のタイピング候補について言語テキストで書かれた可能な変換文字列を関連付けるステップと、

前記第1の確率が前記第2の確率より高い場合に、前記入力文字列を前記第1の候補文字列と関連付けられている前記変換文字列に変換して前記表音文字テキストを前記言語テキストに変換するステップと、

前記第1の確率が前記第2の確率より低い場合に、前記第2の候補文字列を出力して前記非表音文字テキストを変換せずそのままにするステップと

を実行するためのプログラムを格納したことを特徴とする1つまたは複数のコンピュータ読み取り可能媒体。

【発明の詳細な説明】

20

【0001】

(技術分野)

本発明は、言語入力方法およびシステムに関する。より詳細には、本発明は、テキスト入力時に発生する誤植およびある言語形式から別の言語形式への変換時に発生する変換誤りの両方に対する誤り耐性(error tolerance)のある言語入力方法およびシステムを実現する。

【0002】

(発明の背景)

言語固有のワードプロセッサソフトが何年もの間存在している。より複雑なワードプロセッサは、スペルおよび文法の訂正など、ユーザに高等なツールを提供し、文書を起草する際に補助する。たとえば、多数のワードプロセッサは、スペルミスの単語や文法的に間違っているセンテンス構造を識別し、場合によっては、識別された誤りを自動的に訂正することができる。

30

【0003】

一般に、誤りがテキストに入り込む原因は2つある。原因の1つは、ユーザが単に、正しいスペルまたはセンテンス構造を知らないことである。ワードプロセッサは、ユーザが正しいスペルまたは言葉使いを選択する際に手助けするために提案することができる。第2のより一般的な誤りの原因は、ユーザが、正しいスペルまたは文法構造を知っていても単語またはセンテンスを間違えてコンピュータに入力することである。このような状況では、ワードプロセッサは、不適切に入力された文字列を識別し、意図した単語または語句に訂正するのに大変役立つことが多い。

40

【0004】

入力誤りは、ローマ字を使用しない言語用に設計されたワードプロセッサで起こりがちである。英語バージョンのQWERTYキーボードのような言語固有のキーボードは、多くの言語の場合、そのような言語がキーボードのキーとして使いやすく配列できるよりも多くの文字を有するため、存在していない。たとえば、多くのアジア言語は数千文字を含む。これだけ多くの様々な文字について別々のキーをサポートするキーボードを構築することは実際上不可能である。

【0005】

費用のかかる言語および方言固有のキーボードを設計するのではなく、言語固有の文書処

50

理システムは、ユーザが小さな文字セットキーボード（たとえば、Q W E R T Y キーボード）から表音テキスト（p h o n e t i c t e x t）を入力し、その表音テキストを言語テキストに変換することができる。「表音テキスト」は、所与の言語を発話したときに発生する音を表すが、「言語テキスト」はテキストで表示されたとき実際に書かれた文字を表す。たとえば、中国語では、ピンイン（P i n y i n）は表音テキストの一例であり、漢字は言語テキストの一例である。表音テキストを言語テキストに変換することにより、多くの様々な言語を、従来のコンピュータおよび標準Q W E R T Y キーボードを使用して言語固有のワードプロセッサによって処理できる。

【 0 0 0 6 】

表音入力を必要とするワードプロセッサでは、2種類の入力誤りが発生する可能性がある。1つの種類の誤りは、一般的なタイプミスである。しかし、テキストにタイプミスがないとしても、文書処理エンジンが表音テキストを意図しない文字テキストに誤って変換するという誤りもありえる。これら2つの問題の両方が、同じ表音テキスト入力文字列で作用した場合、一連の複数の誤りが生じることがある。状況によっては、語句またはセンテンスの文脈全体を時間をかけて調査しなければ、タイピングで生じる誤りが容易に追跡できない場合がある。

【 0 0 0 7 】

本明細書で説明する本発明は、表音テキストをタイプした場合のユーザによる前者のタイプの入力誤りを主に対象としているが、文書処理エンジンで生じた変換誤りに対する耐性も提供する。このようなタイピング誤りと関連する問題をよく説明するために、表音テキストであるピンインを言語テキストである漢字に変換する中国語ベースのワードプロセッサを考察する。

【 0 0 0 8 】

表音テキストを入力すると多くの場合タイピング誤りが増える理由はいくつかある。理由の1つは、英語用のキーボードでの平均タイピング精度は英語を話す国々の場合よりも中国の場合のほうが低いということである。第2の理由は、表音はそれほど頻繁に使用されているわけではないということである。初期の教育期間中、ユーザは、たとえば、英語を話すユーザが英語で単語を綴ることを教わるほど、表音綴りを勉強、学習する傾向はない。

【 0 0 0 9 】

表音テキスト入力中のタイピング誤りが増える第3の理由は、多くの人々が標準語ではなく方言を母語として話すというものである。表音テキストの起源である標準語は、第2言語である。特定の方言およびアクセントでは、話し言葉は、対応する適切な表音と一致しない場合があり、したがって、ユーザが表音テキストをタイプすることをより難しくする。たとえば、多くの中国人はさまざまな中国語方言を自分の第1言語として話しており、ピンインの起源である北京語を第2言語として教えられている。たとえば、いくつかの中国語方言では、一定の文脈で「h」と「w」を発音する際に区別がなく、他の方言では、「n g」と「n」についても同じことが言え、さらに他の方言では、「r」は明瞭に発音されない。そのため、北京語を第2言語として話す中国人ユーザは、ピンインを入力しようとしてタイピング誤りをしがちである。

【 0 0 1 0 】

タイピング誤りが増える理由としては他に、表音テキストをタイプしながら誤りをチェックすることが困難であるということが考えられる。これは、1つには、表音テキストが長く、読むことが困難な判読しにくい文字列になる傾向があるということによる。見たものがタイプしたものである英語ベースのテキスト入力と対照的に、表音テキストの入力は、「見たものが得られるもの（w h a t y o u s e e i s w h a t y o u g e t）」でないことが多い。むしろ、ワードプロセッサは、表音テキストを言語テキストに変換する。そのため、ユーザは一般に、表音テキストに誤りがないか調べることをしないが、表音テキストが言語テキストに変換されるまで多少待つ。

【 0 0 1 1 】

この最後の理由により、タイピング誤りは、ピンイン入力文脈ではきわめてうっとうしいものである。ピンイン文字列は、文字間にスペーシングがないため再検討、訂正が非常に難しい。その代わりに、ピンイン文字は、ピンイン文字で構成される単語の数に関係なく混在する。さらに、ピンイン - 漢字変換はすぐに実行されることはないが、追加ピンインを入力するにつれ正しい解釈が定式化され続ける。したがって、ユーザが間違っただけのピンイン記号をタイプした場合、単一の誤りが変換プロセスによって複合され、下流に伝播して、いくつかの追加誤りの原因となる場合がある。その結果、システムが漢字に確定変換しその後ユーザが誤りのあったことに気づくまでに、1回訂正するのにユーザは数回バックスペースを入力せざるをえないため、誤り訂正の時間が長くなる。システムによっては、元の誤りを明らかにできない場合さえある。

10

【0012】

表音入力中に間違いが頻繁に生じることが予想されるため、表音入力の誤りを許容できるシステムが必要である。システムは、表音列がわずかに誤りのある文字列を含むとしても正しい答えを返すのが望ましい。

【0013】

言語固有ワードプロセッサは他の問題に直面しており、これは、入力問題とは別であって、異なる言語から同じテキストに単語を入力するために2つの言語間でモードを切り替えることに関するものである。たとえば、技術用語（たとえば、Internet）などの英単語と翻訳が難しい用語（たとえば、頭字語、記号、姓、社名など）を含む文書草稿を中国語で作成することがよくある。従来のワードプロセッサでは、ユーザは異なる単語を入力するときに、一方の言語を他方の言語にモード切替をする必要がある。そのため、ユーザが異なる言語から単語を入力したい場合、ユーザはテキスト入力についての思考を停止し、モードをある言語から別の言語に切り替えて、単語を入力し、それからモードを最初の言語に戻す必要がある。このため、ユーザのタイピング速度が著しく低下し、ユーザは自分の注意をテキスト入力作業と、言語モードの変更という異質な制御作業との間で切り替える必要がある。

20

【0014】

したがって、モード切替を必要としない「モードレス」システムが必要なのである。モードを避けるために、システムは、タイプしている言語を検出し、文字シーケンスを1単語ずつ一方の言語または他方の言語に動的に変換することができなければならない。

30

【0015】

しかし、これは、両方の文脈で多くの文字列が適切な場合があるため、思ったほど簡単ではない。たとえば、多くの有効な英単語は有効なピンイン文字列でもある。さらに、ピンイン入力時に中国語文字間、および中国語文字と英単語との間にスペースがないため、曖昧さが増すことがある。

【0016】

たとえば、ユーザがピンイン入力テキスト「woshiyigezhongguoren」をタイプすると、システムはこの文字列を中国語文字「 」（一般に「私は中国人である」と翻訳される）に変換する。

【0017】

ときには、「woshiyigezhongguoren」とタイプする代わりに、ユーザは次のようにタイプする。

40

【0018】

wos*i*yigezhongguoren（誤りは「sh」と「s」の混同）
 wosh*i*yigezongguoren（誤りは「zh」と「z」の混同）
 wosh*y*igezhongguoren（誤りは、「y」の後の「i」の脱落）
 wosh*i*yigezhonggouren（誤りは、「ou」の並列）
 wosh*i*yigezhongguiren（誤りは「i」と「o」の混同）

【0019】

発明者は、中国語などの難しい外国語でスペルミスの修正を可能にする文書処理システム

50

と方法をすでに開発しており、自動言語認識機能により複数言語のモードレス入力が可能になっている。

【0020】

(発明の概要)

言語入力アーキテクチャにより、表音テキストから言語テキストに変換する際に発生するタイプミスおよび変換誤りを極力減らして、表音テキスト（たとえば、中国語のピンイン）の入力文字列を言語テキスト（たとえば、中国語の漢字）の出力文字列に変換する。言語入力アーキテクチャは、文書処理プログラム、電子メールプログラム、表計算ソフト、ブラウザなどさまざまな分野で実装することができる。

【0021】

一実施形態では、言語入力アーキテクチャは入力文字列、記号、またはその他のテキスト要素を受け取るユーザインタフェースを備える。入力文字列は、表音テキストおよび非表音テキスト、さらに1つまたは複数の言語を含むことができる。このユーザインタフェースにより、ユーザは、異なるテキスト形式または異なる言語の入力のモードを切り替えることなく単一の編集行に入力テキスト文字列を入力することができる。この方法により言語入力アーキテクチャは、ユーザが使いやすいように複数の言語のモードレス入力を実現している。

【0022】

この言語入力アーキテクチャはさらに、検索エンジン、1つまたは複数のタイピングモデル、言語モデル、および異なる言語用の1つまたは複数の用語集を備える。検索エンジンは、ユーザインタフェースから入力文字列を受け取り、入力文字列を1つまたは複数のタイピングモデルに分配する。各タイピングモデルは、各候補文字列が入力文字列として間違っ

【0023】

て入力された場合のタイピング誤り確率に基づいて入力文字列に置換できる有望なタイピング候補のリストを生成するように構成されている。有望なタイピング候補は、データベースに格納できる。

【0024】

一実施形態では、タイピングモデルは、入力テキストの文字列を読み取り、音節を対応するタイプされた各文字列の文字にマップすることによりトレーニングできる。タイプされた各文字が音節の1つにマップされる回数を表す頻度カウントを保持し、各音節のタイピングの確率をその頻度カウントから計算する。

【0025】

タイピングモデルは、入力文字列内に存在する可能性のあるタイプミスの原因となりうるタイピング候補の集まりを返す。タイピング候補は、入力文字列と同じ言語またはテキスト形式で書かれる。

【0026】

検索エンジンはタイピング候補を言語モデルに渡し、これが各タイピング候補の可能性のある変換文字列となる。より具体的には、言語モデルはトライグラム言語モデルであり、有望な変換出力文字列が前の2つのテキストエレメントに基づいて候補文字列を表す言語テキスト確率を求めようとする。変換文字列は、入力文字列と異なる言語または異なるテキスト形式で書かれている。たとえば、入力文字列は、中国語ピンインまたはその他の表音テキストで構成され、出力文字列は漢字またはその他の言語テキストで構成される。

【0027】

タイピングおよび言語モデルで求めた確率に基づき、検索エンジンは最高の確率を示す関

10

20

30

40

50

連するタイピング候補および変換候補を選択する。検索エンジンは、入力文字列（たとえば、表音テキストで書かれている）を言語モデルから返された変換候補からなる出力文字列に変換し、入力されたテキスト形式（たとえば、表音テキスト）を他のテキスト形式（たとえば、言語テキスト）に置き換える。この方法により、表音テキストの入力時のユーザによる入力誤りがなくなる。

【0028】

複数言語を使用する場合、出力文字列は変換候補と入力文字列の一部分（変換なし）との組み合わせを持つことができる。後者の例では、中国語ベースの言語入力アーキテクチャは、両方の変換されたピンイン - 漢字テキストを変換されていない英語テキストとともに出力する。

10

【0029】

ユーザインタフェースは、入力文字列の入力に続けて使用する同じ編集行に出力文字列を表示する。この方法では、変換は自動的に行われ、ユーザが追加テキストを入力すると同時にされる。

【0030】

図面全体を通して類似の構成要素および特徴を参照するのに同じ番号が使用されている。

【0031】

（好ましい実施形態の詳細な説明）

本発明は、言語のある形式（たとえば、表音バージョン）から言語の別の形式（たとえば、書き言葉バージョン）に変換する言語入力システムおよび方法に関連する。このシステムおよび方法は、テキスト入力時に発生するスペルミスおよび誤植ならびにある言語形式から別の言語形式への変換時に発生する変換誤りに対する誤り耐性がある。説明のため、汎用コンピュータで実行される文書処理プログラムの一般的文脈で本発明を説明する。ただし、本発明は、文書処理以外の異なる多くの環境に実装することができ、またさまざまな種類のデバイスで実施することができる。他に、電子メールプログラム、表計算ソフト、ブラウザなどでの文脈が考えられる。

20

【0032】

言語入力システムは、統計的言語モデルを採用して非常に高い精度を達成している。一実施例では、言語入力アーキテクチャは、自動的な最高確率ベースの方法で統計的言語モデリング (statistical language modeling) を使用し、単語をセグメント化し、用語集を選択し、トレーニングデータをフィルタ処理し、可能な最良の変換候補を求める。

30

【0033】

ただし、統計的センテンスベース言語モデリング (Statistical sentence-based language modeling) では、ユーザの入力が完全であると仮定している。実際には、ユーザの入力にはタイピング誤りやスペルミスが多数ある。したがって、言語入力アーキテクチャは、確率論的スペリングモデルを使用して、ありがちなタイピング誤りやスペルミスを許容しながら正しいタイピングを受け入れる1つまたは複数のタイピングモデルを含む。タイピングモデルを英語や中国語など複数言語についてトレーニングし、どれくらいの確からしさで入力シーケンスがある言語の単語であって別の言語の単語ではないかを識別するようにできる。両方のモデルは並列実行でき、その言語モデル（たとえば、中国語モデル）により誘導されて最も可能性の高い文字シーケンス（つまり、英語および中国語の文字）を出力する。

40

【0034】

例示的コンピュータシステム

図1は、中央処理装置 (CPU) 102、メモリ104、および入出力 (I/O) インタフェース106を備える例示的コンピュータシステム100を示している。CPU 102は、メモリ104およびI/Oインタフェース106と通信する。メモリ104は、揮発性メモリ（たとえば、RAM）および不揮発性メモリ（たとえば、ROM、ハードディスクなど）を表す。

50

【 0 0 3 5 】

コンピュータシステム 1 0 0 は、I / O インタフェース 1 0 6 を介して接続された 1 つまたは複数の周辺装置を備える。周辺装置実施例は、マウス 1 1 0、キーボード 1 1 2 (たとえば、英数字 Q W E R T Y キーボード、表音キーボードなど)、ディスプレイモニタ 1 1 4、プリンタ 1 1 6、周辺記憶装置 1 1 8、およびマイクロホン 1 2 0 を備える。たとえば、コンピュータシステムは、汎用コンピュータで実装できる。したがって、コンピュータシステム 1 0 0 は、メモリ 1 0 4 に格納され、C P U 1 0 2 で実行されるコンピュータのオペレーティングシステム (図に示されていない) を実装する。オペレーティングシステムは、ウィンドウ操作環境をサポートするマルチタスクオペレーティングシステムであるのが好ましい。適当なオペレーティングシステムの例として、M i c r o s o f t C o r p o r a t i o n 社の W i n d o w s (登録商標) ブランドのオペレーティングシステムがある。

10

【 0 0 3 6 】

ハンドヘルドデバイス、マルチプロセッサシステム、マイクロプロセッサベースのまたはプログラム可能な家電製品、ネットワーク P C、ミニコンピュータ、メインフレームコンピュータなど、他のコンピュータシステム構成を使用することに注意されたい。さらに、図 1 にはスタンドアローンのコンピュータが示されているが、通信ネットワーク (たとえば、L A N、インターネットなど) を介してリンクされているリモート処理デバイスによってタスクが実行される分散コンピューティング環境で言語入力システムを実行することもできる。分散コンピューティング環境では、プログラムモジュールをローカルとリモートの両方のメモリ記憶デバイスに配置できる。

20

【 0 0 3 7 】

データまたは文書処理プログラム 1 3 0 は、メモリ 1 0 4 に格納され、C P U 1 0 2 によって実行される。他のプログラム、データ、ファイルなども、メモリ 1 0 4 に格納できるが、説明を簡単にするため示していない。文書処理プログラム 1 3 0 は、表音テキストを受け取り、自動的に言語テキストに変換するように構成されている。より具体的には、文書処理プログラム 1 3 0 は、説明のためメモリ内に格納されプロセッサによって実行されるコンピュータソフトウェアとして実装されている言語入力アーキテクチャ 1 3 1 を実装する。文書処理プログラム 1 3 0 は、アーキテクチャ 1 3 1 に加えて他のコンポーネントも備えることができるが、そのようなコンポーネントは、文書処理プログラムに標準のものであると考えられるため、図に詳細に示したり、詳細に説明していない。

30

【 0 0 3 8 】

文書処理プログラム 1 3 0 の言語入力アーキテクチャ 1 3 1 は、ユーザインタフェース (U I) 1 3 2、検索エンジン 1 3 4、1 つまたは複数のタイピングモデル 1 3 5、言語モデル 1 3 6、および異なる言語用の 1 つまたは複数の用語集 1 3 7 を備える。アーキテクチャ 1 3 1 は、言語独立 (l a n g u a g e i n d e p e n d e n t) である。U I 1 3 2 および検索エンジン 1 3 4 は、汎用であり、どの言語でも使用できる。アーキテクチャ 1 3 1 は、言語モデル 1 3 6、タイピングモデル 1 3 5、用語集 1 3 7 を変更することにより特定の言語に合わせる。

40

【 0 0 3 9 】

検索エンジン 1 3 4 および言語モジュール 1 3 6 は一緒に使用することで、表音テキスト - 言語テキスト間のコンバータ 1 3 8 を形成する。タイピングモデル 1 3 5 の助けを借りて、コンバータ 1 3 8 はユーザのタイピング誤りおよびスペルミスに対し許容性を持つ。この開示の目的のために、「テキスト」は、1 つまたは複数の文字および / または文字以外の記号とする。「表音テキスト」は一般に、所与の言語を話すときに生じる音声を表す英数字テキストである。「言語テキスト」は、書き言葉を表す文字と非文字記号である。「非表音テキスト」は、所与の言語を話すときに生じる音声を表さない英数字テキストである。非表音テキストは、言語テキスト以外の書き言葉を表す句読点、特殊文字、および英数字テキストを含む場合がある。

【 0 0 4 0 】

50

おそらく、より一般的に述べると、表音テキストは、書いたときに欧文文字ベースの文字セットを使用しない所与の言語を話したときに出る音声を表す欧文文字セット（たとえば、英語のアルファベット）で表された英数字でよい。言語テキストは、所与の言語に対応する書かれた記号である。

【 0 0 4 1 】

説明の目的のために、ワードプロセッサ 1 3 0 は、中国語ベースのワードプロセッサの文脈で説明し、言語入力アーキテクチャ 1 3 1 はピンインを漢字に変換するように構成されている。つまり、表音テキストはピンインであり、言語テキストは漢字である。しかし、言語入力アーキテクチャは、言語独立であり、他の言語にも使用できる。たとえば、表音テキストは日本語の話し言葉の形態でよいが、言語テキストは漢字などの日本語の書き言葉を表す。アラビア語、韓国語、インド語、その他のアジア言語などを含むが、これだけに限定されない他の例も多数存在する。

10

【 0 0 4 2 】

表音テキストは、マウス 1 1 0、キーボード 1 1 2、またはマイクロホン 1 2 0 などの 1 つまたは複数の周辺入力デバイスを介して入力する。この方法で、ユーザは、キー入力または音声による表音テキスト入力が可能である。音声入力の場合、コンピュータシステムはさらに、話し言葉を受け取る音声認識モジュール（図に示されていない）を実装し、表音テキストに変換することができる。以下の説明では、キーボード 1 1 2 によるテキストの入力をフルサイズの標準英数字 Q W E R T Y キーボードで実行すると想定している。

【 0 0 4 3 】

U I 1 3 2 では、表音テキストを入力と同時に表示する。この U I は、グラフィカルユーザインタフェースであるのが好ましい。U I 1 3 2 の詳細な説明は、引用により本発明に取り込まれている、「L A N G U A G E I N P U T U S E R I N T E R F A C E」という表題の同時係属出願第____号にある。

20

【 0 0 4 4 】

ユーザインタフェース 1 3 2 は、表音テキスト (P) を検索エンジン 1 3 4 に渡し、さらに、これは表音テキストをタイピングモデル 1 3 7 に渡す。タイピングモデル 1 3 7 は、表音テキストに誤りが含まれていると思われる場合にユーザが意図した表音テキストの適当な編集結果と考えられるさまざまなタイピング候補 (T C₁, . . . , T C_N) を生成する。タイピングモデル 1 3 7 は、妥当な確率が設定されている複数のタイピング候補を検索エンジン 1 3 4 に渡し、さらに、これはタイピング候補を言語モデル 1 3 6 に渡す。この言語モデル 1 3 6 は、処理中のセンテンスの文脈の範囲内でタイピング候補を評価し、ユーザが意図した表音テキストの変換された形式を表すと考えられる言語テキストで書かれているさまざまな変換候補 (C C₁, . . . , C C_N) を生成する。変換候補は、タイピング候補と関連付けられている。

30

【 0 0 4 5 】

表音テキストから言語テキストへの変換は 1 対 1 変換ではない。同じかまたは類似した表音テキストが言語テキスト内の多数の文字または記号を表すことがある。したがって、表音テキストの文脈は、言語テキストへの変換前に解釈される。他方、非表音テキストの変換は、通常、表示される英数字テキストが英数字入力と同じである直接的な 1 対 1 変換となる。

40

【 0 0 4 6 】

変換候補 (C C₁, . . . , C C_N) が検索エンジン 1 3 4 に戻され、このエンジンにより、タイピング候補および変換候補のうちどれがユーザが意図するものである確率が最も高いかを判別する統計分析が実行される。確率が計算されると、検索エンジン 1 3 4 により、確率が最も高い候補が選択され、変換候補の言語テキストが U I 1 3 2 に返される。U I 1 3 2 では、表音テキストを変換候補の言語テキストに置き換え、同じ行に表示する。その間、新規入力した表音テキストが新規挿入された言語テキストの前の行に表示され続ける。

【 0 0 4 7 】

50

ユーザが言語テキストを検索エンジン 134 で選択したものから変更したい場合、ユーザインタフェース 132 に、選択が実際に意図した回答である可能性の高さの順序でランク付けられた他の高確率の候補の第 1 のリストが表示される。ユーザがまだ可能な候補に満足しない場合、UI 132 は可能なすべての選択肢を与える第 2 のリストを表示する。第 2 のリストは、確率またはその他の測定基準（たとえば、ストロークカウントまたは中国語文字の複雑さ）に関してランク付けることができる。

【0048】

言語入力アーキテクチャ

図 2 は、言語入力アーキテクチャ 131 の詳細を示している。アーキテクチャ 131 は、タイプミスおよび変換誤りの両方を含む、言語入力の誤り耐性をサポートする。UI 132、検索エンジン 134、言語モデル 136、およびタイピングモデル 135 に加えて、アーキテクチャ 131 はさらに、エディタ 204 およびセンテンス文脈モデル 216 を備える。センテンス文脈モデル 216 は、検索エンジン 134 に結合されている。

10

【0049】

ユーザインタフェース 132 は、表音テキスト（たとえば、中国語ピンインテキスト）および非表音テキスト（たとえば、英語）などの入力テキストを 1 つまたは複数の周辺装置（たとえば、キーボード、マウス、マイクロホン）から受け取り、その入力テキストをエディタ 204 に渡す。エディタ 204 は、検索エンジン 132 がタイピングモデル 135 および言語モデル 136 とともに入力テキストを、言語テキスト（たとえば、中国語の漢字テキスト）などの出力テキストに変換するよう要求する。エディタ 204 は、出力テキストを UI 132 に戻して表示させる。

20

【0050】

検索エンジン 134 は、ユーザインタフェース 132 から入力テキスト文字列を受け取ると、その入力テキスト文字列をタイピングモデル 135 の 1 つまたは複数に送り、さらにセンテンス文脈モデル 216 にも送る。タイピングモデル 135 は、入力テキスト内のタイピング誤りのアприオリな確率を測定する。タイピングモデル 135 は、入力誤り（たとえば、タイプミス）を修正することを実際に求めるユーザによって入力された入力テキストの有望なタイピング候補を生成して出力する。一実施形態では、タイピングモデル 135 は候補データベース 210 内で有望な候補を検索する。他の実施形態では、タイピングモデル 135 は統計ベースのモデリングを使用して、入力テキストの有望な候補を生成する。

30

【0051】

センテンス文脈モデル 216 は、オプションで、タイピングモデル 135 で使用する検索エンジン 132 にセンテンス内のすでに入力されているテキストを送ることができる。この方法では、タイピングモデルは、テキストの新しい文字列とセンテンスにすでに入力されているテキストの文字列との組み合わせに基づいて有望なタイピング候補を生成することができる。

【0052】

「タイピング誤り」、「誤植」、および「スペルミス」という用語は、相互に入れ替えることができ、入力テキストのキー入力時に生じた誤りを指すことは明白である。音声入力の場合、このような誤りは、音声入力の不適切な認識から生じることがある。

40

【0053】

タイピングモデル 135 は、有望なタイピング候補をすべて返すか、または確率の低い有望なタイピング候補を取り除き、それにより、高い確率の有望なタイピング候補のみを検索エンジン 134 に返すことができる。さらに、タイピングモデル 135 よりはむしろ検索エンジン 134 が取り除き機能を実行できることも明白であろう。

【0054】

本発明の一態様によれば、タイピングモデル 135 は、ありがちなタイプミスを観察するため数百あるいは数千のトレーナにセンテンス入力を依頼して集めた実際のデータ 212 を使用してトレーニングされる。以下では「タイピングモデルのトレーニング」という見

50

出しのもとでタイピングモデルおよびトレーニングについて詳述する。

【 0 0 5 5 】

検索エンジン 1 3 4 は、タイピングモデル 1 3 5 から返された有望なタイピング候補のリストを言語モデル 1 3 6 に送る。簡単にいうと、言語モデルは、語句またはセンテンスなど、指定された文脈内に単語またはテキスト文字列がある確率を測定する。つまり、言語モデルは、項目（単語、文字、英字など）のシーケンスを取り、そのシーケンスの確率を推定することができる。言語モデル 1 3 6 は、検索エンジン 1 3 4 から有望なタイピング候補と前のテキストとを組み合わせ、タイピング候補に対応する言語テキストの 1 つまたは複数の候補を生成する。

【 0 0 5 6 】

コーパスデータまたはその他の種類のデータ 2 1 4 を使用して、トライグラム言語モデル 1 3 6 をトレーニングする。トレーニングコーパス 2 1 4 は、新聞記事などの日々のテキストなどの一般的な任意の種類のデータ、あるいは特定の分野（たとえば、医薬品）を対象とするテキストなどの環境固有のデータとすることができる。言語モデル 1 3 6 のトレーニングは、文書処理技術の分野では知られており、ここでは詳述しない。

【 0 0 5 7 】

言語入力アーキテクチャ 1 3 1 は、入力テキスト文字列の入力時に生じる誤りを許容し、入力文字列となる単語およびセンテンスで最も確率の高いものを返そうとする。言語モデル 1 3 6 は、タイピングモデル 1 3 5 でユーザが入力した入力文字列に対しどのセンテンスが最も妥当かを判別する際に役立つ。2つのモデルは、入力された文字列 s が辞書、つまり $P(w | s)$ から認識可能で有効な単語 w である確率として統計的に記述することができる。ベイズの公式を使用すると、確率 $P(w | s)$ は次のように記述される。

【 0 0 5 8 】

【 数 1 】

$$P(w|s) = \frac{P(s|w) \cdot P(w)}{P(s)}$$

【 0 0 5 9 】

分母 $P(s)$ は、入力文字列が与えられたときに可能な意図した単語を比較する目的では同じである。したがって、分析は、分子の積 $P(s | w) \cdot P(w)$ のみに関係し、確率 $P(s | w)$ はスペルまたはタイピングモデルを表し、確率 $P(w)$ は言語モデルを表す。より具体的には、タイピングモデル $P(s | w)$ は、 X を入力するつもりの人が代わりに Y を入力することになる確率を記述するが、言語モデル $P(w)$ はセンテンス文脈を与えたときに特定の単語が生成されているべき確率を記述する。

【 0 0 6 0 】

ピンインを漢字に変換する文脈では、確率 $P(w | s)$ は $P(H | P)$ と言い換えることができ、 H は漢字文字列、 P はピンイン文字列を表す。目標は、 $P(H | P)$ を最大にする、最も確率の高い中国語文字 H を見つけることである。したがって、確率 $P(H | P)$ は、入力されたピンイン文字列 P が有効な漢字文字列 H である確率である。 P は固定されており、したがって $P(P)$ は与えられたピンイン文字列に対し一定であるため、ベイズ公式により、次のように確率 $P(H | P)$ が小さくなる。

【 0 0 6 1 】

$$H = \arg \max_H P(H | P) = \arg \max_H P(P | H) * P(H)$$

【 0 0 6 2 】

確率 $P(P | H)$ はスペルまたはタイピングモデルを表す。通常、漢字文字列 H は、さらに、複数の単語 $W_1, W_2, W_3, \dots, W_M$ に分割され、確率 $P(P | H)$ は次のように推定できる。

【 0 0 6 3 】

$$Pr(P | H) = P(P_{f(i)} | W_i)$$

【 0 0 6 4 】

10

20

30

40

50

$P_{f(i)}$ は、単語 W_i に対応するピンイン文字列のシーケンスである。

【0065】

従来技術の統計ベースのピンイン - 漢字変換システムでは、確率 $P(P_{f(i)} | W_i)$ は、 $P_{f(i)}$ が単語 W_i の受け入れ可能なスペルである場合に 1 に設定され、 $P_{f(i)}$ が単語 W_i の受け入れ可能なスペルでない場合に 0 に設定される。そのため、従来のシステムは、誤って入力された文字に対する耐性がない。一部のシステムでは、「南部訛りの発音」機能があり、このような問題に対応しているが、これはプリセット値確率 1 および 0 を採用している。さらに、このようなシステムは、データ駆動方式でないためタイピング誤りのうちごくわずかしが扱えない（実際のタイピング誤りから学習する）。

【0066】

対照的に、本発明で説明している言語入力アーキテクチャでは、タイピングモデルと言語モデルの両方を利用して変換を実行する。タイピングモデルでは、実際のコーパスから確率 $P(P_{f(i)} | W_i)$ をトレーニングすることにより誤って入力した文字に対する誤り耐性が可能になっている。タイピングモデルを構築する方法は多数ある。理論的には、すべての可能な $P(P_{f(i)} | W_i)$ をトレーニングできるが、実際には、パラメータが多すぎる。トレーニングする必要があるパラメータの個数を減らす 1 つの方法として、1 文字単語のみを考察し、発音が等価なすべての文字を単一の音節にマップする方法がある。中国語にはおよそ 406 個の音節があり、これは本質的に $P(\text{ピンインテキスト} | \text{音節})$ をトレーニングし、各文字に対応する音節にマップすることである。以下では「タイピングモデルのトレーニング」という見出しのもとでこれについて詳述する。

【0067】

言語入力アーキテクチャ 131 では、広範な確率が計算される。ピンイン - 漢字変換の目標は、確率 $P(P | H)$ を最大にする漢字文字列 H を見つけることである。これは、最大の確率を最良の漢字シーケンスとして求める W_i を選択することにより実行する。実際、よく知られている Viterbi Beam 検索のような効率的な検索方法を使用できる。Viterbi Beam 検索法の詳細については、「Automatic Speech Recognition」(Kluwer Academic Publishers、1989) という表題の Kai-Fu Lee の記事、および「Automatic Speech and Speaker Recognition - Advanced Topics」(Kluwer Academic Publishers、1996) という表題の Chin-Hui Lee、Frank K. Soong、Kuldip K. Paliwal の記事を記載されている。

【0068】

確率 $P(H)$ は、言語モデルを表し、所与の単語列のアプリオリな確率を測定する。統計的言語モデル構築の一般的な方法として、プレフィックスツリー風のデータ構造を利用して、知られているテキストのトレーニングセットから N グラム言語モデルを構築する方法がある。広く使用されている統計的言語モデルの一実施例として、 N グラムマルコフモデルがあり、これについては、Frederick Jelinek 著「Statistical Methods for Speech Recognition」(The MIT Press, Cambridge, Massachusetts、1997) に説明がある。プレフィックスツリーデータ構造 (a.k.a. サフィックスツリー、または PAT ツリー) の使用により、高レベルアプリケーションで言語モデルを素早く横断 (トラバース) し、上述のように、実質的にリアルタイムに実行する特性を持つ。 N グラム言語モデルでは、テキスト全体を通して文字列 (サイズ N) 内の特定のアイテム (単語、文字など) の出現数をカウントする。このカウントを使用して、そのアイテムの列の使用の確率を計算する。

【0069】

言語モデル 136 は、トライグラム言語モデル (つまり、 $N = 3$ とする N グラム) であるのが好ましいが、バイグラムが文脈によっては適当な場合がある。トライグラム言語モデルは、英語に適しており、また中国にも十分機能するが、大きなトレーニングコーパスを

10

20

30

40

50

利用すると想定している。

【 0 0 7 0 】

トライグラムモデルでは、次のように、次の文字を予測するためにテキスト文字列内の最も前の 2 つの文字を考慮する。

(a) 文字 (C) は、定義済み用語集を使用して離散言語テキストまたは単語 (W) にセグメント化され、ツリー内の各 W は 1 つまたは複数の C にマップされる。

(b) 前の 2 つの単語から単語のシーケンス (W_1, W_2, \dots, W_M) の確率を予測する。

【 0 0 7 1 】

【 数 2 】

$$P(W_1, W_2, W_3, \dots, W_M) \approx \prod P(W_n | W_{n-1}, W_{n-2}) \quad (1)$$

【 0 0 7 2 】

ただし、 $P()$ は言語テキストの確率を表す。

【 0 0 7 3 】

W_n は現在の単語である

W_{n-1} は前の単語である

W_{n-2} は W_{n-1} の前の単語である

【 0 0 7 4 】

図 3 は、ユーザによって入力された入力テキスト 3 0 0 の例を示しており、タイピングモデル 1 3 5 および言語モデル 1 3 6 に渡される。入力テキスト 3 0 0 を受け取ると、タイピングモデル 1 3 5 は入力テキスト 3 0 0 を異なる方法でセグメント化し、キーボード入力時に生じる可能性のあるタイプミスを考慮した有望なタイピング候補のリストを生成する。タイピング候補 3 0 2 は、各時間フレーム内に異なるセグメントがあり、前の単語の終了時間が現在の単語の開始時間となる。たとえば、候補 3 0 2 の上行は、入力文字列 3 0 0 「ma f a n g n i t r y y i s . . . 」を「ma」、「fan」、「ni」、「try」、「yi」などにセグメント分割する。タイピング候補 3 0 2 の第 2 行は、入力文字列「ma f a n g n i t r y y i s . . . 」を異なる形で「ma」、「fang」、「nit」、「yu」、「xia」などにセグメント分割する。

【 0 0 7 5 】

これらの候補は、データベースまたはその他の何らかのアクセス可能なメモリに格納できる。図 3 は単なる一例にすぎず、入力テキストに対する有望なタイピング候補は多数あり得ることは明白であろう。

【 0 0 7 6 】

言語モデル 1 3 6 は、センテンスの文脈で有望なタイピング候補 3 0 2 の各セグメントを評価し、関連する言語テキストを生成する。説明のため、有望なタイピングテキスト 3 0 2 の各セグメントおよび対応する有望な言語テキストはボックスにまとめられている。

【 0 0 7 7 】

これらの候補から、検索エンジン 1 3 4 は、候補のうちどれがユーザが意図するものである確率が最も高いかを判別する統計分析を実行する。各行内のタイピング候補は、互に関連がなく、検索エンジンは任意の行からさまざまなセグメントを自由に選択し、受け入れ可能な変換候補を定義することができる。図 3 の例では、検索エンジンは、ハイライト表示になっているタイピング候補 3 0 4、3 0 6、3 0 8、3 1 0、3 1 2、および 3 1 4 が最高の確率であることを示していると判断している。これらの候補は、左から右に連結され、候補 3 0 4 の後に候補 3 0 6 が続くなどして、入力テキスト 3 0 0 の受け入れ可能な解釈を形成することができる。

【 0 0 7 8 】

確率を計算した後、検索エンジン 1 3 4 は、確率が最高の候補を選択する。検索エンジンは次に、入力された表音テキストを選択した候補と関連する言語テキストに変換する。たとえば、検索エンジンは入力テキスト 3 0 0 をボックス 3 0 4、3 0 6、3 0 8、3 1 0

10

20

30

40

50

、 3 1 2、および 3 1 4 で示されている言語テキストに変換し、エディタ 2 0 4 を介して言語テキストをユーザインタフェース 1 3 2 に返す。句読点がユーザインタフェースに届いた後、つまり新しい入力テキスト文字列が新しいセンテンス内に入ると、タイピングモデル 1 3 5 は新しいセンテンス内の新しいテキスト文字列に対する操作を開始する。

【 0 0 7 9 】

一般的な変換

図 4 は、表音テキスト（たとえば、ピンイン）を言語テキスト（たとえば、漢字）に変換する一般的プロセス 4 0 0 を示している。このプロセスは、言語入力アーキテクチャ 1 3 1 によって実装されており、図 2 をさらに参照して説明する。

【 0 0 8 0 】

ステップ 4 0 2 で、ユーザインタフェース 1 3 2 は、ユーザが入力したピンインなどの表音テキスト列を受け取る。入力テキスト文字列には、1 つまたは複数のタイプミスが含まれる。UI 1 3 2 は、エディタ 2 0 4 を介して入力テキストを検索エンジン 1 3 4 に渡し、検索エンジンは入力テキストをタイピングモデル 1 3 5 とセンテンス文脈モデル 2 1 6 に分配する。

【 0 0 8 1 】

ステップ 4 0 4 で、タイピングモデル 1 3 5 は入力テキストに基づいて有望なタイピング候補を生成する。候補を導く一方法として、入力テキスト文字列を異なるパーティションに分割し、その入力文字列セグメントに最もよく類似する候補をデータベース内で検索する。たとえば、図 3 で、候補 3 0 2 は可能なセグメント「m a」、「f a n」などを示すセグメンテーションを持つ。

【 0 0 8 2 】

有望なタイピング候補が、検索エンジン 1 3 4 に返され、その後、言語モデル 1 3 6 に伝達される。言語モデル 1 3 6 は、有望なタイピング候補と前のテキストとを組み合わせ、タイピング候補に対応する言語テキストの 1 つまたは複数の候補を生成する。たとえば、図 3 の候補 3 0 2 を参照すると、言語モデルはボックス 3 0 2 内に言語テキストを可能な出力テキストとして返す。

【 0 0 8 3 】

ステップ 4 0 6 で、検索エンジン 1 3 4 は、候補のうちどれがユーザが意図するものである確率が最も高いかを判別する統計分析を実行する。表音テキストに最も可能性の高いタイピング候補を選択した後、検索エンジンは入力された表音テキストをタイピング候補と関連する言語テキストに変換する。この方法により、表音テキストの入力時のユーザによる入力誤りがなくなる。検索エンジン 1 3 4 は、エディタ 2 0 4 を介して誤りのない言語テキストを UI 1 3 2 に返す。ステップ 4 0 8 で、変換された言語テキストは、ユーザが表音テキストを入力し続けている UI 1 3 2 の画面上の同じ行内位置に表示される。

【 0 0 8 4 】

タイピングモデルのトレーニング

上で指摘したように、タイピングモデル 1 3 5 は、確率 $P(s | w)$ に基づいている。タイピングモデルでは、入力テキストを出力テキストに変換するのに使用できる異なるタイピング候補の確率を計算し、有望な候補を選択する。この方法で、タイピングモデルは、タイピング誤りが存在していても入力テキストの有望なタイピング候補を返すことにより誤りを許容する。

【 0 0 8 5 】

本発明の一態様は、実際のデータからのタイピングモデル $P(s | w)$ のトレーニングに関するものである。タイピングモデルは、数百または好ましくは数千などできる限り多くのトレーナによるテキスト入力に基づいて開発またはトレーニングされる。トレーナは同じまたは異なるトレーニングデータを入力し、入力されたデータとトレーニングデータとの間の分散をタイピング誤りとして捕らえる。目標は、同じトレーニングテキストをタイプさせ、タイピングでの誤りの個数またはタイピング候補に基づいて確率を求めることである。このようにして、タイピングモデルはトレーナのタイピング誤りの確率を学習する

10

20

30

40

50

。

【0086】

図5は、プロセッサ502、揮発性メモリ504、および不揮発性メモリ506を備えるトレーニング用コンピュータ500を示している。トレーニング用コンピュータ500では、ユーザが入力したデータ510から確率512（つまり、 $P(s|w)$ ）を求めるトレーニングプログラム508を実行する。トレーニングプログラム508は、プロセッサ502で実行するように図に示されているが、不揮発性メモリ506のストレージからプロセッサにロードされる。トレーニング用コンピュータ500は、進行中（on the fly）の入力時に、あるいは収集しメモリに格納した後にデータ510に基づいてトレーニングを行うように構成することができる。

10

【0087】

説明のため、中国語用に手直ししたタイピングモデルであって、中国語ピンインテキストが中国語文字テキストに変換されるものを考察する。この場合、数千人の人々にピンインテキストを入力してくれるよう勧誘する。好ましくは、タイピングでの誤りの種類および個数が類似するようにすることを目標として、数千個またはそれ以上のセンテンスを各人から収集する。タイピングモデルは、検索エンジンからピンインテキストを受け取り、入力文字列内の文字の置き換えに使用できる有望な候補を供給するように構成されている。

【0088】

タイピングモデル135をトレーニングするためにさまざまな手法を使用できる。一方法では、タイピングモデルは、単一文字テキストを考察し、すべての同等な発音の文字テキストを単一音節にマップすることにより直接トレーニングされる。たとえば、中国語ピンインには400を超える音節がある。音節を与える表音テキストの確率（たとえば、 $P(\text{ピンインテキスト} | \text{音節})$ ）をトレーニングし、各文字テキストを対応する音節にマップする。

20

【0089】

図6は、音節マッピングトレーニング手法600を示している。ステップ602で、トレーニングプログラム508は、トレーナが入力したテキスト文字列を読みとる。テキスト文字列は、センテンスでも、また単語および/または文字のその他のグループでもよい。プログラム508は、音節をテキストの文字列内の対応する英字に合わせるか、またはマップする（ステップ604）。各テキスト文字列では、各音節にマップした英字の頻度が更新される（ステップ606）。これは、ステップ608から「はい」分岐で表されているように、トレーナにより入力されたトレーニングデータに含まれるテキスト文字列ごとに繰り返される。最終的に、入力されたテキスト文字列は、中国語ピンインの多くのまたはすべての音節を表す。ステップ608から「いいえ」分岐により表されるように、すべての文字列が読み込まれたら、トレーニングプログラムは、各音節をタイプするユーザの確率 $P(\text{ピンインテキスト} | \text{音節})$ を決定する（ステップ610）。一実施形態では、タイピングの確率は、最初にすべての音節を正規化することにより決定される。

30

【0090】

各音節は、隠しマルコフモデル（HMM）として表すことができる。各入力キーは、HMMでマップされている状態のシーケンスとして表示できる。正しい入力および実際の入力をすりあわせて状態間の遷移確率を求める。異なるHMMを使用して、異なる技能レベルのタイピストをモデル化することができる。

40

【0091】

中国語で406個すべての音節をトレーニングするには、大量のデータが必要である。このデータ要件を緩和するために、異なる音節内の同じ文字を1つの状態として結びつける。これにより、状態の個数は27にまで減らされる（つまり、「a」から「z」までの26個の異なる文字に、不明な文字を表す文字1つ）。このモデルは、トライグラム言語モデルを利用するViterbi beam検索法に統合することもできる。

【0092】

さらに他のトレーニング手法では、トレーニングは文字の挿入

50

【 0 0 9 3 】

【 数 3 】

(つまり、 $\emptyset \rightarrow x$)、

【 0 0 9 4 】

文字の削除

【 0 0 9 5 】

【 数 4 】

(つまり、 $x \rightarrow \emptyset$)、

【 0 0 9 6 】

および一方の文字を他方に置換（つまり、 $x \rightarrow y$ ）などの単一文字編集の確率に基づく。このような単一文字編集の確率は次のように統計的に表すことができる。

【 0 0 9 7 】

置換： $P(x \text{ を } y \text{ で置換})$

挿入： $P(x \text{ を } y \text{ の前 / 後に挿入})$

削除： $P(x \text{ を } y \text{ の前 / 後に削除})$

【 0 0 9 8 】

各確率（ P ）は、本質的にバイグラムタイピングモデルであるが、隣接する文字を超えるかなり広い文脈のテキストを考慮した N グラムタイピングモデルに拡張することもできる。したがって、入力テキストの可能な文字列について、タイピングモデルは、まず正しい文字シーケンスを供給し、次に動的プログラミングを使用して正しい文字シーケンスを与えられた文字シーケンスに変換する最低コスト経路を求めることにより、すべての可能な文字シーケンスを生成する確率を持つ。コストは、最小数の誤り文字または他の何らかの測定基準として決めることができる。実際には、この誤りモデルは、V i t e r b i B e a m 検索法の一部として実装できる。

【 0 0 9 9 】

タイピング誤りまたはスペルミス以外の種類の誤りは、本発明の範囲内でトレーニングできることは明白であろう。さらに、異なるトレーニング手法を使用して、本発明の範囲から逸脱することなくタイピングモデルをトレーニングすることができることも明白であろう。

【 0 1 0 0 】

モードレス入力の多言語トレーニング

言語入力システムを悩ます他のやっかいな問題として、2つまたはそれ以上の言語を入力したときのモードの切り替えの必要性である。たとえば、中国語でタイプしているユーザは、英語の単語を入力したい場合がある。従来の入力システムでは、ユーザは英単語のタイピングと中国語の単語のタイプのモード切替が必要である。残念なことに、ユーザが切り替えを忘れやすいということである。

【 0 1 0 1 】

言語入力アーキテクチャ 131（図 1）をトレーニングして混合言語入力を受け入れるためにトレーニングすることができ、したがって、多言語文書処理システムにおいて2つまたはそれ以上の言語間のモード切替をなくすることができる。これは、「モードレス入力」と呼ばれる。

【 0 1 0 2 】

この言語入力アーキテクチャは、中国語と英語とを区別するなど、異なる言語の単語を自動的に識別するスペル/タイピングモデルを実装している。これは、多くの正当な英単語は正当なピンイン文字列であるため容易ではない。さらに、ピンイン、英語、および中国語文字の間にスペースが入らないため、入力時に曖昧さが増すことがある。以下のバイズ規則を使用すると、

【 0 1 0 3 】

10

20

30

40

50

$$H = \arg \max_H P(H | P) = \arg \max_H P(P | H) * P(H)$$

【0104】

目的関数は、英語ではスペルモデル $P(P | H)$ 、中国語では言語モデル $P(H)$ の2つの部分があることで特徴付けることができる。

【0105】

混合言語入力を取り扱う一方法として、第2言語からの単語を第1言語の特殊カテゴリとして取り扱うことにより、第1言語（たとえば、中国語）の言語モデルをトレーニングする方法がある。たとえば、第2言語からの単語を第1言語の単一単語として取り扱う。

【0106】

たとえば、中国語ベースの文書処理システムでは、英語キーボードを入力デバイスとして使用する。中国語ベースの文書処理システムで採用しているタイピングモデルは、英単語と中国語単語を混ぜたテキストでトレーニングされる中国語言語モデルである。

【0107】

混合言語入力を取り扱う第2の方法として、言語入力アーキテクチャで2つのタイピングモデル、中国語タイピングモデルと英語タイピングモデルを実装し、それぞれを別々にトレーニングする方法がある。つまり、中国語タイピングモデルは、上で説明した方法でトレーナにより入力された表音列などのキーボード入力のストリーム上でトレーニングされ、英語タイピングモデルは英語を話すトレーナによって入力された英語テキスト上でトレーニングされる。

【0108】

英語タイピングモデルは、以下の組み合わせとして実装することができる。

1. 中国語テキストに挿入された実際の英語上でトレーニングしたユニグラム言語モデル。このモデルは、多くの頻繁に使用される英単語を取り扱えるが、見たことのない英単語は予測できない。

2. 3音節確率の英語スペルモデル。このモデルは、すべての3音節シーケンスに対し確率が0でないが、英語に似た単語になる可能性のある単語については確率が高くなる。これは、実際の英単語からもトレーニングでき、見たことのない英単語も取り扱える。

【0109】

これらの英語モデルは、一般に、英語テキストに対しては非常に高い確率、英語テキストのように見える英字列には高い確率、非英語テキストには低い確率を返す。

【0110】

図7は、複数のタイピングモデル135(1)～135(N)を採用するために図2のアーキテクチャ131から修正された言語入力アーキテクチャ700を示している。各タイピングモデルは、特定の言語に合わせて構成されている。各タイピングモデル135は、単語と、特定の言語に共通する誤りを使用して別々にトレーニングされる。したがって、別々のトレーニングデータ212(1)～212(N)が、関連するタイピングモデル135(1)～135(N)について供給される。実施例では、英語に対して1つのタイピングモデル、中国語に対して1つのタイピングモデルというようにタイピングモデルを2つだけ使用している。ただし、言語入力アーキテクチャを修正して、2つよりも多いタイピングモデルを含めて、2つよりも多い言語の入力に対応するようにできることは明白であろう。また、言語入力アーキテクチャは、日本語、韓国語、フランス語、ドイツ語などの他の多くの多言語文書処理システムでも使用できることも指摘しておくべきであろう。

【0111】

言語入力アーキテクチャの操作時に、英語タイピングモデルは中国語タイピングモデルと並列に動作する。2つのタイピングモデルは互いに競合しており、入力したテキスト文字列が中国語文字列（誤りを含む）または英語文字列（さらに潜在的に誤りを含む）である可能性のある確率を計算することにより入力テキストが英語か中国語かを識別する。

【0112】

入力テキストの文字列またはシーケンスが明確に中国語ピンインテキストの場合、中国語タイピングモデルは英語タイピングモデルよりもかなり高い確率を返す。そこで、言語入

10

20

30

40

50

力アーキテクチャは、入力されたピンインテキストを漢字テキストに変換する。入力テキストの文字列またはシーケンスが明確に英語（たとえば、姓、頭字語（「I E E E」）、会社名（「M i c r o s o f t」）、技術（「I N T E R N E T」）、など）の場合、英語タイピングモデルは中国語タイピングモデルよりもかなり高い確率を示す。したがって、このアーキテクチャは、英語タイピングモデルに基づいて入力テキストを英語テキストに変換する。

【0113】

入力テキストの文字列またはシーケンスが曖昧な場合、中国語および英語タイピングモデルは、中国語および英語間の曖昧さを解消するためにさらなる文脈からより多くの情報が得られるまで確率を計算し続ける。入力テキストの文字列またはシーケンスが中国語にも英語にも似ていない場合、中国語タイピングモデルは英語タイピングモデルよりも許容性が低い。そのため、英語タイピングモデルは確率が、中国語タイピングモデルよりも高くなる。

【0114】

多言語変換を説明するために、ユーザが「私はINTERNETマガジンを読むのが好きだ」という意味のテキスト文字列「woa idu internet zazhi」を入力すると仮定する。初期文字列「woa idu」を受け取った後、中国語タイピングモデルは、英語タイピングモデルよりも高い確率となり、入力テキストのその部分を「

」に変換する。このアーキテクチャは、続いてタイプした曖昧な部分「i n t e r n e」を英字「t」がタイプされるまで探し続ける。このときに、英語タイピングモデルは、「INTERNET」について中国語タイピングモデルよりも高い確率を返し、言語入力アーキテクチャは入力テキストのこの部分を「INTERNET」に変換する。次に、中国語タイピングモデルは、「z a z h i」について英語タイピングモデルよりも高い確率を示し、言語入力アーキテクチャは入力テキストのその部分を「」に変換する。

【0115】

多言語入力変換

図8は、タイプミスとともに入力された多言語入力テキスト文字列を誤りのない多言語出力テキスト文字列に変換するプロセス800を示している。このプロセスは、言語入力アーキテクチャ700によって実装されており、図7をさらに参照して説明する。

【0116】

ステップ802で、ユーザインタフェース132は、多言語入力テキスト文字列を受け取る。これは、表音単語（たとえば、ピンイン）および少なくとも1つの他の言語（たとえば、英語）の単語を含む。入力テキストはさらに、ユーザが表音単語および第2言語の単語を入力したときのタイプミスも含む場合がある。UI 132は、エディタ204を介して多言語入力テキスト文字列を検索エンジン134に渡し、検索エンジンは入力テキストをタイピングモデル135（1）～135（N）とセンテンス文脈モデル216に分配する。

【0117】

タイピングモデルはそれぞれ、ステップ804（1）～804（N）によって表されるような入力テキストに基づいて有望なタイピング候補を生成する。ステップ806で、妥当な確率が設定された有望なタイピング候補が検索エンジン134に返される。ステップ808で、検索エンジン134がタイピング確率とともにタイピング候補を言語モデル136に送る。ステップ810で、言語モデルが有望なタイピング候補と前のテキストとを組み合わせ、センテンススペースの文脈を用意し、図3に関して上で説明したように、そのタイピング候補を通じて経路を選択することによりタイピング候補に対応する言語テキストの1つまたは複数の変換候補を生成する。ステップ812で、検索エンジン134は統計分析を実行して、ユーザが意図する最高の確率を示す変換候補を選択する。

【0118】

ステップ814で、テキスト文字列に対する最も有望な変換候補が出力テキスト文字列に

変換される。出力テキスト文字列は、言語テキスト（たとえば、漢字）と第2言語（たとえば、英語）とを含むが、タイピング誤りは省かれる。検索エンジン134は、エディタ204を介して誤りのない出力テキストをUI 132に返す。ステップ816で、変換された言語テキストは、ユーザが表音テキストを入力し続けているUI 132の画面上の同じ行内位置に表示される。

【0119】

上の例では、中国語が主言語であり、英語は第2言語である。2つの言語は両方とも、主言語として指定できることは明白であろう。さらに、2つよりも多い言語は混合入力テキスト文字列を形成することができる。

【0120】

10

結論

上の説明では、構造機能および/または方法論的動作に固有の言語を使用しているが、付属の特許請求の範囲で定義されている本発明は説明した特定の機能または動作に限られるわけではない。むしろ、特定の機能および動作は、本発明を実装する例示の形式として開示されている。

【図面の簡単な説明】

【図1】 言語入力アーキテクチャを実装する言語固有のワードプロセッサを有するコンピュータシステムのブロック図である。

【図2】 言語入力アーキテクチャの実施例のブロック図である。

【図3】 構文解析または異なる音節のセットにセグメント化されたテキスト文字列、および、テキスト文字列に誤りが含まれると仮定してそれらの音節を置き換えるのに使用することができる候補を例示する図である。

20

【図4】 言語入力アーキテクチャで実行される一般的変換を示す流れ図である。

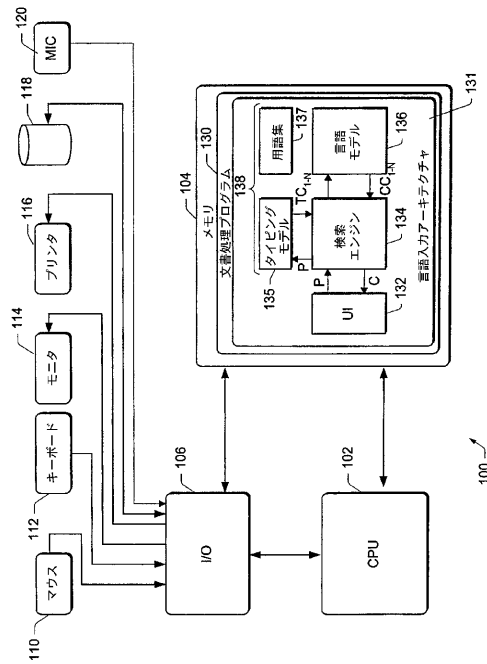
【図5】 言語入力アーキテクチャで採用されている確率ベースのモデルをトレーニングするために使用されるトレーニングコンピュータのブロック図である。

【図6】 ートレーニング手法を例示する流れ図である。

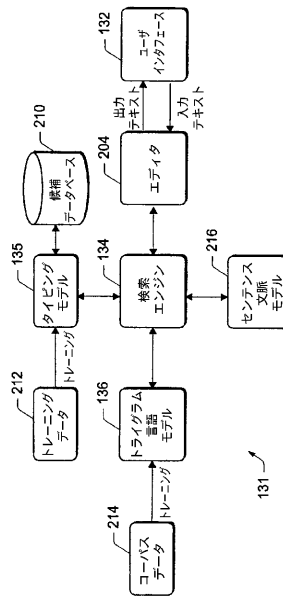
【図7】 複数のタイピングモデルを使用する、言語入力アーキテクチャの他の実施例のブロック図である。

【図8】 他言語変換プロセスを例示する流れ図である。

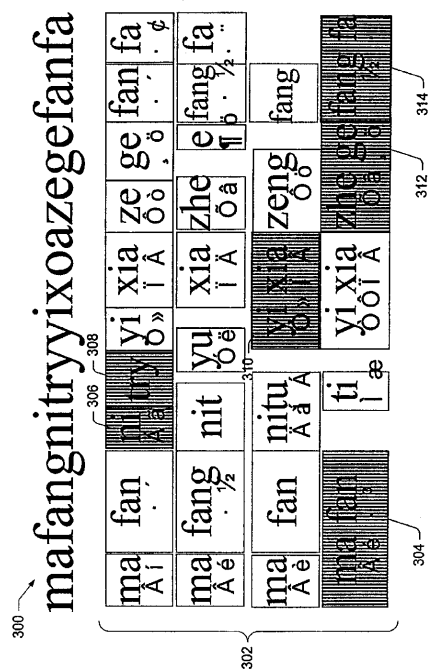
【 図 1 】



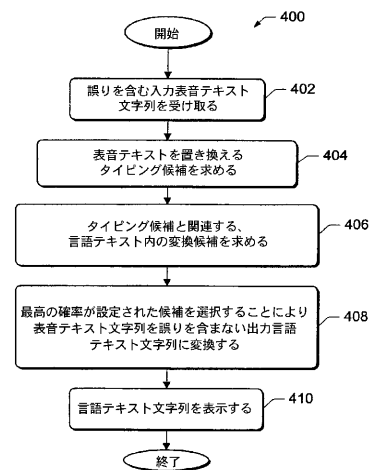
【 図 2 】



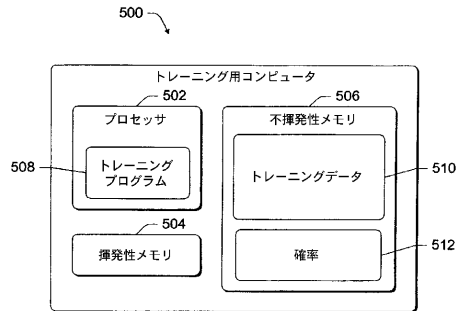
【 図 3 】



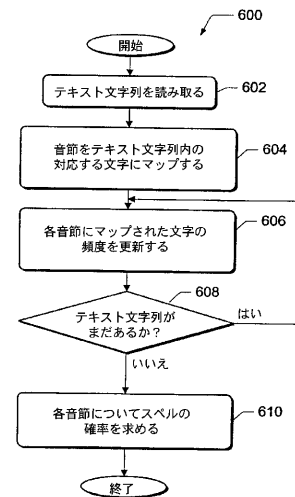
【 図 4 】



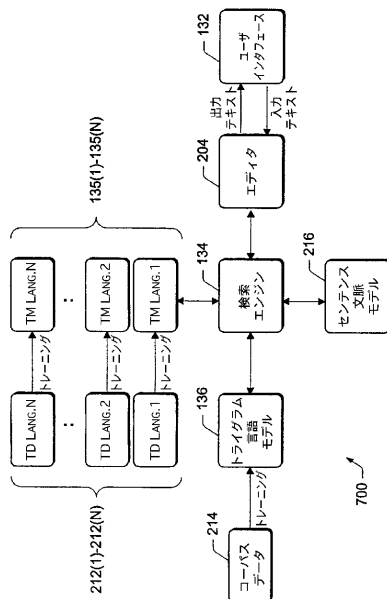
【図 5】



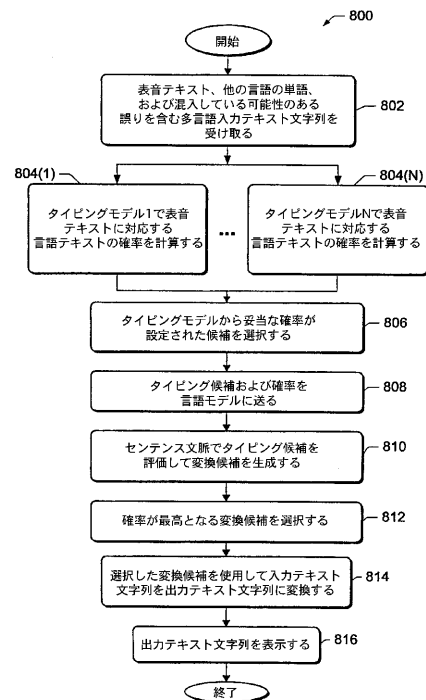
【図 6】



【図 7】



【図 8】



フロントページの続き

(72)発明者 ズエン チェン

中華人民共和国 北京 ハイディアン ディストリクト ズイチュン ロード (番地なし) ビルディング 9 ルーム 1801

(72)発明者 ジアン ハン

中華人民共和国 北京 ハイダン ディストリクト ドンワンズアン コミュニティー (番地なし) ビルディング 5 ルーム 702

審査官 成瀬 博之

(56)参考文献 特開平10-232863(JP,A)

特開平11-007448(JP,A)

特開平05-282360(JP,A)

特開平04-075162(JP,A)

Mark Kernighan, 外2名, A Spelling Correction Program Based on a Noisy Channel Model, Proc. of the 13th Conference on Computational Linguistics, 米国, Association for Computational Linguistics, 1990年, Vol. 2, pp. 205-210

(58)調査した分野(Int.Cl., DB名)

G06F 17/20-17/28