

【特許請求の範囲】**【請求項 1】**

単語を受け付け、

受け付けた前記単語に第 1 の変換パラメータ及び第 2 の変換パラメータをそれぞれ適用して、前記単語に応じた第 1 のベクトル及び第 2 のベクトルを生成し、

生成した前記第 1 のベクトル及び前記第 2 のベクトルに基づき、前記単語に応じた新たな第 3 のベクトルを生成する

処理をコンピュータに実行させることを特徴とする生成プログラム。

【請求項 2】

前記第 3 のベクトルを生成する処理は、生成した第 1 のベクトルと前記第 2 のベクトルとを連結して前記第 3 のベクトルを生成することを特徴とする請求項 1 に記載の生成プログラム。

10

【請求項 3】

前記第 3 のベクトルを生成する処理は、生成した第 1 のベクトルと前記第 2 のベクトルとを演算することにより、前記第 3 のベクトルを生成することを特徴とする請求項 1 又は 2 に記載の生成プログラム。

【請求項 4】

前記第 1 のベクトル及び第 2 のベクトルを生成する処理は、前記第 1 のベクトルとは次元が異なる前記第 2 のベクトルを生成することを特徴とする請求項 1 乃至 3 のいずれか 1 つに記載の生成プログラム。

20

【請求項 5】

前記第 1 のベクトル及び第 2 のベクトルを生成する処理は、前記単語が前記第 1 の変換パラメータ又は前記第 2 の変換パラメータに含まれない未知語であると判定した場合、前記未知語であることを示すベクトルを生成することを特徴とする請求項 1 乃至 4 のいずれか 1 つに記載の生成プログラム。

【請求項 6】

前記第 1 のベクトル及び第 2 のベクトルを生成する処理は、前記未知語であることを示すベクトルとして、0 ベクトル又は乱数ベクトルを生成することを特徴とする請求項 5 に記載の生成プログラム。

【請求項 7】

前記第 1 のベクトル及び第 2 のベクトルを生成する処理は、前記未知語であることを示すベクトルとして、未知語であると判定されなかった前記第 1 のベクトル又は第 2 のベクトルを複製したベクトルを生成することを特徴とする請求項 5 に記載の生成プログラム。

30

【請求項 8】

前記第 1 のベクトルと前記第 2 のベクトルとの組み合わせを用いて学習処理を行うことにより、前記未知語であることを示すベクトルを生成する処理をさらにコンピュータに実行させることを特徴とする請求項 5 に記載の生成プログラム。

【請求項 9】

複数の判別対象の文章に含まれる単語またはフレーズの素性に関する学習を実行した学習モデルを用いて、前記第 3 のベクトルから、受け付けた前記単語と前記複数の判別対象の文章との判別結果を取得する処理をさらにコンピュータに実行させることを特徴とする請求項 1 乃至 8 のいずれか 1 つに記載の生成プログラム。

40

【請求項 10】

前記判別結果を取得する処理は、前記第 1 のベクトル、前記第 2 のベクトル及び前記第 3 のベクトルのうち、いずれか 1 つ又は 2 つ以上の組み合わせを選択し、選択されたベクトルに対応する前記学習モデルを用いて、受け付けた前記単語と前記複数の判別対象の文章との判別結果を取得することを特徴とする請求項 9 に記載の生成プログラム。

【請求項 11】

前記受け付ける処理は、複数の前記単語を含むテキストデータを受け付け、

前記第 1 のベクトル及び第 2 のベクトルを生成する処理は、前記テキストデータに応じ

50

た前記第 1 のベクトル及び第 2 のベクトルを生成し、

前記第 3 のベクトルを生成する処理は、前記テキストデータに応じた新たな第 3 のベクトルを生成することを特徴とする請求項 1 乃至 10 のいずれか 1 つに記載の生成プログラム。

【請求項 12】

前記第 1 のベクトル及び第 2 のベクトルを生成する処理は、受け付けた前記単語に第 3 の変換パラメータをさらに適用して、前記単語に応じた第 4 のベクトルをさらに生成し、

前記第 3 のベクトルを生成する処理は、生成した前記第 1 のベクトル及び前記第 2 のベクトルに加えて、さらに前記第 4 のベクトルに基づき、前記単語に応じた新たな第 3 のベクトルを生成することを特徴とする請求項 1 乃至 11 のいずれか 1 つに記載の生成プログラム。

10

【請求項 13】

コンピュータが、
単語を受け付け、

受け付けた前記単語に第 1 の変換パラメータ及び第 2 の変換パラメータをそれぞれ適用して、前記単語に応じた第 1 のベクトル及び第 2 のベクトルを生成し、

生成した前記第 1 のベクトル及び前記第 2 のベクトルに基づき、前記単語に応じた新たな第 3 のベクトルを生成する
処理を行う生成方法。

【請求項 14】

20

単語を受け付ける受付部と、

受け付けた前記単語に第 1 の変換パラメータ及び第 2 の変換パラメータをそれぞれ適用して、前記単語に応じた第 1 のベクトル及び第 2 のベクトルを生成する第 1 ベクトル処理部と、

生成した前記第 1 のベクトル及び前記第 2 のベクトルに基づき、前記単語に応じた新たな第 3 のベクトルを生成する第 2 ベクトル処理部と、
を有する生成装置。

【発明の詳細な説明】

【技術分野】

【0001】

30

本発明は、生成プログラム、生成方法及び生成装置に関する。

【背景技術】

【0002】

言語処理において、複数のコーパスを併用する技術が知られている。例えば、機械翻訳において、複数のコーパスを併用することで、翻訳結果の品質を高め、翻訳の応答時間を短縮する技術が知られている。当該技術は、第 1 言語に属する語句と第 2 言語に属する語句を対応付けて格納した複数の翻訳用辞書を用いる。当該技術は、1 つ以上の語句を含む基準情報を受け入れ、複数の翻訳用辞書と基準情報とを比較して、当該基準情報に対する各翻訳用辞書の類似度を求める。当該技術は、類似度をもとに、各翻訳用辞書を検索する際の優先度を規定する。また、当該技術においては、辞書ではなく、文章の集まりであるコーパスを指定する場合がある。

40

【先行技術文献】

【特許文献】

【0003】

【特許文献 1】特開 2004 - 355217 号公報

【発明の概要】

【発明が解決しようとする課題】

【0004】

上記技術においては、いずれかのコーパスの特徴が失われるという問題が生じる。例えば、FAQ などの複数の文章を格納したデータベースを用いて、文章を形態素解析により

50

分解し、機械学習により単語ベクトルを生成する際に、コーパスから生成される、単語をベクトルに変換する変換パラメータを用いる。この場合において、例えば、一方のコーパスから生成された変換パラメータを選択した場合、選択されなかったコーパスの特徴は、検索結果に反映されない。一方、2つのコーパスを統合したコーパスを用いて変換パラメータを生成する際に、統合前の各コーパスに含まれる特徴が失われる場合がある。

【0005】

一つの側面では、変換パラメータに含まれる特徴を保持できる生成プログラム、生成方法及び生成装置を提供することを目的とする。

【課題を解決するための手段】

【0006】

一つの態様において、生成プログラムは、単語を受け付け、単語に第1の変換パラメータ及び第2の変換パラメータをそれぞれ適用して、単語に応じた第1のベクトル及び第2のベクトルを生成する処理をコンピュータに実行させる。生成プログラムは、生成した第1のベクトル及び第2のベクトルに基づき、単語に応じた新たな第3のベクトルを生成する処理をコンピュータに実行させる。

10

【発明の効果】

【0007】

一つの態様によれば、変換パラメータに含まれる特徴を保持できる。

【図面の簡単な説明】

【0008】

20

【図1】図1は、実施例1における回答抽出処理の一例を示す図である。

【図2】図2は、実施例1における生成装置の一例を示す図である。

【図3】図3は、実施例1におけるコーパスモデルの一例を示す図である。

【図4】図4は、実施例1における検索時学習モデルの一例を示す図である。

【図5】図5は、実施例1における回答記憶部の一例を示す図である。

【図6】図6は、実施例1におけるベクトル結合処理の一例を示す図である。

【図7】図7は、実施例1における回答抽出処理の一例を示すフローチャートである。

【図8】図8は、背景技術におけるベクトル生成処理の一例を示す図である。

【図9】図9は、実施例1におけるベクトル生成処理の一例を示す図である。

【図10】図10は、実施例2におけるベクトル生成処理の一例を示す図である。

30

【図11】図11は、実施例2における生成装置の一例を示す図である。

【図12】図12は、実施例2における回答抽出処理の一例を示すフローチャートである。

【図13】図13は、実施例2におけるベクトル生成処理の別の一例を示す図である。

【図14】図14は、実施例3におけるベクトル結合処理の一例を示す図である。

【図15】図15は、実施例3におけるベクトル生成処理の一例を示す図である。

【図16】図16は、実施例3におけるベクトル生成処理の別の一例を示す図である。

【図17】図17は、ハードウェア構成例を示す図である。

【発明を実施するための形態】

【0009】

40

以下に、本願の開示する生成プログラム、生成方法及び生成装置の実施例を図面に基づいて詳細に説明する。なお、この実施例によりこの発明が限定されるものではない。また、以下に示す各実施例は、矛盾を起こさない範囲で適宜組み合わせても良い。

【実施例1】

【0010】

実施例1における、後に説明する生成装置10は、コーパスモデルを用いて生成される単語ベクトルを用いた回答抽出処理をコンピュータに実行させる。実施例1における回答抽出処理は、例えばコールセンターの質疑応答集（FAQ）などを参照し、入力された質問文に対する適切な応答を抽出する処理である。なお、生成装置10は、サーバ、パーソナルコンピュータ、タブレットなどのコンピュータ装置の一例である。また、コーパスモ

50

デルは、変換パラメータの一例である。

【0011】

実施例1における回答抽出処理について、図1を用いて説明する。図1は、実施例1における回答抽出処理の一例を示す図である。図1に示すように、実施例1における回答抽出処理においては、事前学習処理において、コーパスを学習することにより、後に説明するコーパスモデルA121a及びコーパスモデルB121bが生成される。なお、以下において、コーパスモデルA121a及びコーパスモデルB121bを区別せずに表現する場合に、コーパスモデル121と表記する場合がある。

【0012】

実施例1におけるコーパスは、複数の文章の集まりであり、例えば百科事典サイトや新聞データベース、検索エンジンなどの外部サーバから取得されるテキスト群である。また、実施例1におけるコーパスは、例えば顧客データベースやコールセンターの質疑応答ログなど、生成装置10を利用する会社などの内部情報であってもよい。実施例1においては、例えばコーパスAが百科事典サイトであり、コーパスBが内部情報である。

【0013】

実施例1におけるコーパスモデル121は、例えば、コーパスに含まれる文章群を形態素解析によって単語に区分し、区分された単語群に対する分散表現の単語ベクトル(Bag of words)を学習することにより生成される。単語の分散表現とは、各単語を、単語間の各特徴要素である複数の次元に対して、定量化した連続値で表現した、多次元のベクトルである。なお、単語の分散表現は、例えばWord2Vecなどの公知の技術により学習することができるため、詳細な説明は省略する。

【0014】

また、コーパスモデル121は、例えば単語ベクトルと正解ラベルとを関連付けた教師データを、ニューラルネットワークなどに入力して、入力ベクトルと正解ラベルとの関係性を学習し、学習済みモデルを生成することにより得られる。

【0015】

次に、推論処理において、生成装置10は、まず、質問文などのテキストの入力を受け付ける。図1に示すように、生成装置10は、例えば「PCの電源が入らない」などの入力テキスト1000を受け付ける。次に、生成装置10は、入力テキストに対して形態素解析を行い、入力テキスト1000を「PC, の, 電源, が, 入ら, ない, 。」のような単語群1001に区分する。

【0016】

そして、生成装置10は、区分された各単語を、予め生成されたコーパスモデル121に入力して、単語ベクトルを取得する。生成装置10は、例えば入力単語「PC」をコーパスモデルA121aに入力することにより、単語ベクトルAを取得する。同様に、生成装置10は、入力単語「PC」をコーパスモデルB121bに入力することにより、単語ベクトルBを取得する。図1に示すように、コーパスモデルA121a及びコーパスモデルB121bは、それぞれ異なるコーパスから得られたものであるため、取得される単語ベクトルA及びBも相互に異なる。なお、単語ベクトルは、第1のベクトル及び第2のベクトルの一例である。

【0017】

次に、生成装置10は、取得された単語ベクトルA及びBを結合する。生成装置10は、例えば単語ベクトルAに、単語ベクトルBを連結することにより、単語ベクトルCを生成する。同様に、生成装置10は、区分されたその他の入力単語についても推論処理及び結合処理を繰り返すことにより、入力テキスト「PCの電源が入らない」に対応する分散表現を得る。なお、単語ベクトルCは、第3のベクトルの一例である。

【0018】

一方、生成装置10は、入力テキストを用いた検索対象とする回答データについても、事前学習処理と同様の処理により得られる、後に説明する検索時学習モデル122を有する。図1に示すように、検索時学習モデル122は、後に説明する回答記憶部123に記

10

20

30

40

50

憶されたタイトルを形態素解析によって区分し、区分された単語群に対して単語ベクトルを学習することにより得られる。

【 0 0 1 9 】

そして、検索処理において、生成装置 1 0 は、入力テキストに対応する分散表現を、検索時学習モデル 1 2 2 に入力することにより、入力テキストに対応する回答を取得する。例えば、生成装置 1 0 は、回答記憶部 1 2 3 を参照し、タイトル「電源ボタンを押しても、パソコンが起動しません。」に対応する回答「対処方法は～」を取得する。そして、生成装置 1 0 は、取得した回答を出力する。

【 0 0 2 0 】

このように、生成装置 1 0 は、入力単語に対して、第 1 のモデルから生成する第 1 ベクトルと第 2 のモデルから生成する第 2 ベクトルとを用いて生成した第 3 ベクトルを生成するので、各モデルの特徴を残したベクトルで機械学習ができる。

10

【 0 0 2 1 】

[機能ブロック]

次に、本実施例における生成装置 1 0 の一例について、図 2 を用いて説明する。図 2 は、実施例 1 における生成装置の一例を示す図である。図 2 に示すように、本実施例における生成装置 1 0 は、記憶部 1 2 0 と、制御部 1 3 0 とを有する。

【 0 0 2 2 】

記憶部 1 2 0 は、プログラムやデータを記憶する記憶装置の一例であり、例えばメモリやプロセッサなどである。この記憶部 1 2 0 は、コーパスモデル A 1 2 1 a 及びコーパスモデル B 1 2 1 b、検索時学習モデル 1 2 2 並びに回答記憶部 1 2 3 を記憶する。

20

【 0 0 2 3 】

実施例 1 におけるコーパスモデル A 1 2 1 a 及びコーパスモデル B 1 2 1 b は、コーパスから生成される、単語と、分散表現の単語ベクトルとを対応付けた辞書である。図 3 は、実施例 1 におけるコーパスモデルの一例を示す図である。図 3 に示すように、実施例 1 におけるコーパスモデル A 1 2 1 a は、「単語」と、「ベクトル」とを対応付けて記憶する。なお、コーパスモデル B 1 2 1 b についても、コーパスモデル A 1 2 1 a と同様の構成であるため、詳細な説明は省略する。

【 0 0 2 4 】

検索時学習モデル 1 2 2 は、回答データに含まれるタイトルを形態素解析によって区分し、区分された単語群に対して単語ベクトルを学習することにより得られる学習モデルである。図 4 は、実施例 1 における検索時学習モデルの一例を示す図である。図 4 に示すように、検索時学習モデル 1 2 2 は、「ID」と、「ベクトル」とを対応付けて記憶する。

30

【 0 0 2 5 】

図 4 において、「ID」は、回答データを一意に識別する識別子 (Identifier) を記憶する。「ベクトル」は、当該 ID の回答データから得られた分散表現の単語ベクトルを記憶する。

【 0 0 2 6 】

回答記憶部 1 2 3 は、図 1 に示すように、回答記憶部 1 2 3 には、例えば予め図示しない他のサーバ等から取得した情報が記憶される。図 5 は、実施例 1 における回答記憶部の一例を示す図である。図 5 に示すように、回答記憶部 1 2 3 は、「タイトル」と、「回答」とを「ID」に対応付けて記憶する。

40

【 0 0 2 7 】

図 5 において、「タイトル」は、回答データに対応する質問文を記憶する。「回答」は、回答データの本文を記憶する。

【 0 0 2 8 】

図 2 に戻って、制御部 1 3 0 は、生成装置 1 0 全体を司る処理部であり、例えばプロセッサなどである。制御部 1 3 0 は、受付部 1 3 1、単語抽出部 1 3 2、ベクトル抽出部 1 3 3、ベクトル結合部 1 3 4、回答検索部 1 3 5 及び出力部 1 3 6 を有する。なお、受付部 1 3 1、単語抽出部 1 3 2、ベクトル抽出部 1 3 3、ベクトル結合部 1 3 4、回答検索

50

部 1 3 5 及び出力部 1 3 6 は、プロセッサが有する電子回路の一例やプロセッサが実行するプロセスの一例である。

【 0 0 2 9 】

受付部 1 3 1 は、入力テキストを受け付け、単語抽出部 1 3 2 に出力する。受付部 1 3 1 は、例えば図示しない生成装置 1 0 の操作部を通じて、利用者から入力テキストを受け付ける。また、受付部 1 3 1 は、例えば図示しない生成装置 1 0 の通信部を通じて、利用者が操作する端末から入力テキストを受け付けてもよい。

【 0 0 3 0 】

単語抽出部 1 3 2 は、入力テキストに対する形態素解析を行う。単語抽出部 1 3 2 は、受付部 1 3 1 から入力テキストの出力を受けると、入力テキストを形態素解析によって図 1 に示すような単語群に区分する。単語抽出部 1 3 2 は、区分した単語群を、ベクトル抽出部 1 3 3 に出力する。なお、単語抽出部 1 3 2 は、形態素解析以外のその他の手法によって、入力テキストを単語群に区分してもよい。

10

【 0 0 3 1 】

ベクトル抽出部 1 3 3 は、区分された単語群に対応する単語ベクトルを、コーパスモデル 1 2 1 ごとに複数抽出して、ベクトル結合部 1 3 4 に出力する。なお、ベクトル抽出部 1 3 3 は、第 1 ベクトル処理部の一例である。

【 0 0 3 2 】

ベクトル抽出部 1 3 3 は、単語をコーパスモデル A 1 2 1 a に入力することにより、単語ベクトル A を抽出する。同様に、ベクトル抽出部 1 3 3 は、単語をコーパスモデル B 1 2 1 b に入力することにより、単語ベクトル B を抽出する。

20

【 0 0 3 3 】

ベクトル結合部 1 3 4 は、ベクトル抽出部 1 3 3 から出力されたベクトルを結合する。ベクトル結合部 1 3 4 は、例えば、単語ベクトル A に、単語ベクトル B を連結することにより、結合された単語ベクトル C を生成する。なお、ベクトル結合部 1 3 4 は、第 2 ベクトル処理部の一例である。

【 0 0 3 4 】

図 6 は、実施例 1 におけるベクトル結合処理の一例を示す図である。図 6 に示すように、ベクトル結合部 1 3 4 は、単語ベクトル A の各要素の後ろに、単語ベクトル B の各要素を連結することにより、単語ベクトル C を生成する。この場合において、例えば単語ベクトル A 及び B がそれぞれ 1 0 0 次元のベクトルである場合、単語ベクトル C は 2 0 0 次元のベクトルとなる。

30

【 0 0 3 5 】

[処理の流れ]

次に、本実施例における処理について、図 7 を用いて説明する。図 7 は、実施例 1 における回答抽出処理の一例を示すフローチャートである。図 7 に示すように、生成装置 1 0 の受付部 1 3 1 は、図示しない利用者から、入力テキストを受け付けるまで待機する (S 1 0 0 : N o) 。受付部 1 3 1 は、入力テキストを受け付けたと判定した場合 (S 1 0 0 : Y e s) 、単語抽出部 1 3 2 に入力テキストを出力する。

【 0 0 3 6 】

単語抽出部 1 3 2 は、出力された入力テキストを形態素解析によって単語に区分する。そして、単語抽出部 1 3 2 は、区分された単語を抽出して、ベクトル抽出部 1 3 3 に出力する (S 1 0 1) 。

40

【 0 0 3 7 】

ベクトル抽出部 1 3 3 は、単語をコーパスモデル A 1 2 1 a に入力することにより、単語ベクトル A を抽出する (S 1 1 0) 。同様に、ベクトル抽出部 1 3 3 は、単語をコーパスモデル B 1 2 1 b に入力することにより、単語ベクトル B を抽出する (S 1 2 0) 。そして、ベクトル抽出部 1 3 3 は、抽出した単語ベクトル A 及び単語ベクトル B を、ベクトル結合部 1 3 4 に出力する。

【 0 0 3 8 】

50

ベクトル結合部 134 は、出力された単語ベクトル A 及び単語ベクトル B を結合して、単語ベクトル C を生成する (S 130)。そして、ベクトル結合部 134 は、入力テキストから区分された全ての単語について、ベクトル化が完了したか否かを判定する (S 140)。

【0039】

ベクトル結合部 134 は、ベクトル化が完了していないと判定した場合 (S 140: No)、S 101 に戻って処理を繰り返す。一方、ベクトル結合部 134 は、ベクトル化が完了したと判定した場合 (S 140: Yes)、単語ごとに生成された単語ベクトル C を結合して、入力テキスト全体をベクトル化し (S 150)、回答検索部 135 に出力する。

10

【0040】

回答検索部 135 は、出力された入力テキスト全体のベクトルを、検索時学習モデル 122 に入力することにより、入力テキストに類似する回答を検索し (S 160)、検索結果を出力部 136 に出力する。そして、出力部 136 は、出力された検索結果を、図示しない通信部又は表示部を通じて出力する (S 170)。

【0041】

[効果]

以上説明したように、本実施例における生成プログラムは、単語を受け付け、単語に第 1 の変換パラメータ及び第 2 の変換パラメータをそれぞれ適用して、単語に応じた第 1 のベクトル及び第 2 のベクトルを生成する処理をコンピュータに実行させる。生成プログラムは、生成した第 1 のベクトル及び第 2 のベクトルに基づき、単語に応じた新たな第 3 のベクトルを生成する処理をコンピュータに実行させる。これにより、変換パラメータに含まれる特徴を保持できる。

20

【0042】

また、生成プログラムは、複数の判別対象の文章に含まれる単語またはフレーズの素性に関する学習を実行した学習モデルを用いて、第 3 のベクトルから、受け付けた単語と前記複数の判別対象の文章との判別結果を取得する処理をコンピュータに実行させてもよい。これにより、文章の検索処理において、変換パラメータに含まれる特徴を、検索結果に反映させることができる。

【0043】

本実施例におけるベクトル生成処理の効果について、図 8 及び図 9 を用いて説明する。図 8 は、背景技術におけるベクトル生成処理の一例を示す図である。背景技術においては、図 8 に示すように、2 つのコーパス A 及び B を結合した総合コーパスから、コーパスモデルが生成される。

30

【0044】

この際、例えば 2 つのコーパスの大きさに偏りがある場合など、いずれかのコーパスに含まれる特徴が、コーパスモデルに反映されずに失われる可能性がある。この場合、入力単語を当該コーパスモデルに入力して得られる単語ベクトルにおいて、各コーパスの特徴が反映されない場合がある。これにより、単語ベクトルを学習モデルに入力することにより得られる回答においても、コーパスの特徴が失われる場合がある。

40

【0045】

一方、実施例 1 においては、図 9 に示すように、コーパスそのものを結合するのではなく、コーパス A 及び B から生成された各コーパスモデル A 及び B から得られた単語ベクトルを結合する。図 9 は、実施例 1 におけるベクトル生成処理の一例を示す図である。図 9 に示すように、実施例 1 における検索処理においては、単語ベクトル A 及び単語ベクトル B の特徴が残る、図 6 に示すような単語ベクトル C を用いる。単語ベクトル A 及び単語ベクトル B は、それぞれコーパス A 及びコーパス B の特徴を残すものであるため、単語ベクトル C を学習モデルに入力することにより得られる回答においても、各コーパスの特徴を反映させることができる。

【0046】

50

また、生成プログラムは、例えば生成した第1のベクトルと第2のベクトルとを連結して第3のベクトルを生成する処理をコンピュータに実行させてもよい。これにより、各ベクトルに含まれる要素を、そのまま残すことができる。

【0047】

なお、生成プログラムは、複数の単語を含むテキストデータを受け付け、テキストデータに応じた第1のベクトル及び第2のベクトルを生成し、テキストデータに応じた新たな第3のベクトルを生成する処理をコンピュータに実行させてもよい。つまり、単語素性の系列は複数であっても単数であってもよい。これにより、単語ではなく、質問文などの自然言語の文章を用いた検索ができる。なお、テキストデータに含まれる各単語から生成された各単語ベクトルをニューラルネットワークに順次入力するのではなく、各単語ベクトルを結合した1つの入力データをニューラルネットワークに入力する手法を採用することもできる。

10

【実施例2】

【0048】

単語ベクトルを生成する際に、入力単語がいずれかのコーパスに含まれない場合、すなわちコーパスから生成されるコーパスモデルに当該入力単語が登録されていない場合がある。そこで、実施例2においては、いずれかのコーパスに登録されていない入力単語があることを示す、結合された単語ベクトルを生成する構成について説明する。なお、いずれかのコーパスに登録されていない入力単語を、以下において「未知語」と表記する場合がある。

20

【0049】

実施例2における生成プログラムは、未知語であることを示す単語ベクトルを、もう一方の当該未知語を含むコーパスに対応する単語ベクトルと結合する。図10は、実施例2におけるベクトル生成処理の一例を示す図である。図10に示すように、コーパスモデルB121bにおいては、コーパスモデルA121aに含まれる「ノートPC」という単語が登録されていない。

【0050】

この場合において、生成プログラムは、入力単語が「ノートPC」である場合、コーパスモデルA121aを参照して、単語ベクトルAを抽出する処理をコンピュータに実行させる。一方、生成プログラムは、コーパスモデルB121bに「ノートPC」が登録されていないため、単語ベクトルBを抽出する処理を実行させることができない。

30

【0051】

この場合において、生成プログラムは、入力単語が未知語であることを示す単語ベクトルB2を生成し、単語ベクトルAと結合する処理をコンピュータに実行させる。

【0052】

例えば、生成プログラムは、全ての要素が0であるベクトルを、単語ベクトルAに連結させたベクトルC2を生成する処理をコンピュータに実行させる。また、生成プログラムは、単語ベクトルAを複製して、単語ベクトルAに連結したベクトルC3を生成する処理をコンピュータに実行させてもよい。さらに、生成プログラムは、後に説明する未知語ベクトル記憶部224に予め記憶された未知語ベクトルを、単語ベクトルAに連結させたベクトルC4を生成する処理をコンピュータに実行させてもよい。

40

【0053】

[機能ブロック]

次に、当該生成プログラムを実行する生成装置について、図11を用いて説明する。図11は、実施例2における生成装置の一例を示す図である。なお、以下の実施例において、先に説明した図面に示す部位と同一の部位には同一の符号を付し、重複する説明は省略する。

【0054】

図11に示すように、本実施例における生成装置20は、記憶部220と、制御部230とを有する。記憶部220は、プログラムやデータを記憶する記憶装置の一例であり、

50

例えばメモリやプロセッサなどである。この記憶部 1 2 0 は、コーパスモデル A 1 2 1 a 及びコーパスモデル B 1 2 1 b、検索時学習モデル 1 2 2 並びに回答記憶部 1 2 3 に加えて、さらに未知語ベクトル記憶部 2 2 4 を記憶する。

【 0 0 5 5 】

未知語ベクトル記憶部 2 2 4 は、入力単語が未知語であることを示す未知語ベクトルを記憶する。実施例 2 において、未知語ベクトルは、例えば乱数ベクトルであるが、後に説明するように、コーパスモデル A 及びコーパスモデル B を学習することにより生成されたベクトルであってもよい。未知語ベクトル記憶部 2 2 4 に記憶される情報は、例えば図示しない生成装置 2 0 の管理者により予め入力され、又は後に説明する未知語ベクトル処理部 2 3 7 により入力される。

10

【 0 0 5 6 】

制御部 2 3 0 は、生成装置 2 0 全体を司る処理部であり、例えばプロセッサなどである。制御部 2 3 0 は、受付部 1 3 1、単語抽出部 1 3 2、ベクトル結合部 1 3 4、回答検索部 1 3 5 及び出力部 1 3 6 に加えて、さらにベクトル抽出部 2 3 3 及び未知語ベクトル処理部 2 3 7 を有する。なお、ベクトル抽出部 2 3 3 及び未知語ベクトル処理部 2 3 7 も、プロセッサが有する電子回路の一例やプロセッサが実行するプロセスの一例である。

【 0 0 5 7 】

ベクトル抽出部 2 3 3 は、ベクトル抽出部 1 3 3 と同様に、入力単語が区分された単語群に対応する単語ベクトルを、コーパスモデルごとに複数抽出して、ベクトル結合部 1 3 4 に出力する。その際、ベクトル抽出部 2 3 3 は、各コーパスモデルに入力単語が登録されているか否かを判定する。

20

【 0 0 5 8 】

ベクトル抽出部 2 3 3 は、コーパスモデルに入力単語が登録されていないと判定した場合、未知語ベクトル処理部 2 3 7 に、未知語ベクトルの取得要求を出力する。そして、ベクトル抽出部 2 3 3 は、取得した未知語ベクトルを、当該コーパスモデルに対応する単語ベクトルとして、ベクトル結合部 1 3 4 に出力する。

【 0 0 5 9 】

未知語ベクトル処理部 2 3 7 は、ベクトル抽出部 2 3 3 から未知語ベクトルの取得要求を受けると、未知語ベクトルをベクトル抽出部 2 3 3 に出力する。未知語ベクトル処理部 2 3 7 は、図 1 0 に示すように、例えば、全ての要素が 0 であるベクトルを、未知語ベクトルとして出力する。

30

【 0 0 6 0 】

また、未知語ベクトル処理部 2 3 7 は、例えばベクトル抽出部 2 3 3 から抽出済みの単語ベクトルを含む未知語ベクトルの取得要求を受け、取得要求に含まれる抽出済みの単語ベクトルを、未知語ベクトルとして出力してもよい。さらに、未知語ベクトル処理部 2 3 7 は、例えば未知語ベクトル記憶部 2 2 4 に記憶された未知語ベクトルを取得して、ベクトル抽出部 2 3 3 に出力してもよい。

【 0 0 6 1 】

[処理の流れ]

次に、本実施例における処理について、図 1 2 を用いて説明する。図 1 2 は、実施例 2 における回答抽出処理の一例を示すフローチャートである。なお、以下の説明において、図 7 に示すステップと同じ符号については同様のステップであるため、詳細な説明を省略する。

40

【 0 0 6 2 】

図 1 2 に示すように、生成装置 2 0 のベクトル抽出部 2 3 3 は、単語がコーパスモデル A 1 2 1 a に登録済みであるか否かを判定する (S 1 0 2)。ベクトル抽出部 2 3 3 は、単語がコーパスモデル A 1 2 1 a に登録済みであると判定した場合 (S 1 0 2 : Y e s)、単語ベクトル A を抽出し (S 1 1 0)、S 1 1 2 に移行する。

【 0 0 6 3 】

一方、ベクトル抽出部 2 3 3 は、単語がコーパスモデル A 1 2 1 a に登録済みではない

50

と判定した場合 (S 1 0 2 : N o)、未知語ベクトル処理部 2 3 7 に、未知語ベクトルの取得要求を出力する。

【 0 0 6 4 】

未知語ベクトル処理部 2 3 7 は、未知語ベクトルの取得要求の出力を受けると、未知語ベクトルをベクトル抽出部 2 3 3 に出力し (S 1 1 1)、S 1 1 2 に移行する。

【 0 0 6 5 】

同様に、ベクトル抽出部 2 3 3 は、単語がコーパスモデル B 1 2 1 b に登録済みであるか否かを判定する (S 1 1 2)。ベクトル抽出部 2 3 3 は、単語がコーパスモデル B 1 2 1 b に登録済みであると判定した場合 (S 1 1 2 : Y e s)、単語ベクトル B を抽出し (S 1 2 0)、S 1 3 0 に移行する。

10

【 0 0 6 6 】

一方、ベクトル抽出部 2 3 3 は、単語がコーパスモデル B 1 2 1 b に登録済みではないと判定した場合 (S 1 1 2 : N o)、未知語ベクトル処理部 2 3 7 に、未知語ベクトルの取得要求を出力する。

【 0 0 6 7 】

未知語ベクトル処理部 2 3 7 は、未知語ベクトルの取得要求の出力を受けると、未知語ベクトルをベクトル抽出部 2 3 3 に出力し (S 1 2 1)、S 1 3 0 に移行する。

【 0 0 6 8 】

[効果]

以上説明したように、実施例 2 における生成プログラムは、単語が第 1 の変換パラメータ又は第 2 の変換パラメータに含まれない未知語であると判定した場合、未知語であることを示すベクトルを生成する処理をコンピュータに実行させる。これにより、入力単語が未知語であることを示す、結合された単語ベクトルを生成できる。

20

【 0 0 6 9 】

なお、生成プログラムは、未知語であることを示すベクトルとして、0 ベクトル又は乱数ベクトルを生成する処理をコンピュータに実行させてもよい。また、生成プログラムは、未知語であることを示すベクトルとして、未知語であると判定されなかった第 1 又は第 2 のベクトルを複製したベクトルを生成する処理をコンピュータに実行させてもよい。これにより、未知語ベクトルを任意の形で生成できる。

【 0 0 7 0 】

なお、未知語ベクトル処理部 2 3 7 は、未知語ベクトルとして、乱数ベクトルの代わりに、コーパスモデル A 及びコーパスモデル B を学習することにより、未知語ベクトルを生成してもよい。

30

【 0 0 7 1 】

図 1 3 は、実施例 2 におけるベクトル生成処理の別の一例を示す図である。図 1 3 に示すように、未知語ベクトル処理部 2 3 7 は、コーパスモデル A 及びコーパスモデル B に共通して含まれる「PC」、「スマホ」、「タブレット」などの各単語の単語ベクトルを用いて学習することにより、未知語ベクトル B 3 を生成する。

【 0 0 7 2 】

このように、実施例 2 における生成プログラムは、第 1 のベクトルと第 2 のベクトルとの組み合わせを用いて学習処理を行うことにより、未知語であることを示すベクトルを生成する処理をコンピュータに実行させてもよい。これにより、各コーパスの特徴に即した未知語ベクトルを生成できる。

40

【 実施例 3 】

【 0 0 7 3 】

さて、これまで本発明の実施例について説明したが、本発明は上述した実施例以外にも、種々の異なる形態にて実施されてよいものである。そこで、以下に異なる実施例を説明する。

【 0 0 7 4 】

例えば、ベクトル結合部 1 3 4 が、単語ベクトル A と単語ベクトル B とを連結する構成

50

について説明したが、これに限られず、単語ベクトル A と単語ベクトル B とを演算するような構成であってもよい。

【0075】

図14は、実施例3におけるベクトル結合処理の一例を示す図である。図14に示すように、実施例3における図示しないベクトル結合部334は、単語ベクトル A と単語ベクトル B とを加算することにより、結合された単語ベクトル C を生成する。

【0076】

このように、生成プログラムは、生成した第1のベクトルと第2のベクトルとを演算することにより、第3のベクトルを生成する処理をコンピュータに実行させてもよい。これにより、結合前の単語ベクトルの次元を増加させることなく、複数の単語ベクトルを結合させることができる。

10

【0077】

また、ベクトル結合部134が、次元が同一である単語ベクトル A と単語ベクトル B とを連結する構成について説明したが、これに限られず、単語ベクトル A と単語ベクトル B とで次元が異なるような構成であってもよい。

【0078】

図15は、実施例3におけるベクトル生成処理の一例を示す図である。図15に示すように、実施例3におけるコーパスモデル A ' 3 2 1 a は、200次元の単語ベクトルを記憶する。一方、実施例3におけるコーパスモデル B ' 3 2 1 b は、100次元の単語ベクトルを記憶する。この場合において、図示しないベクトル抽出部333は、200次元の単語ベクトル A と、100次元の単語ベクトル B とをそれぞれ抽出してもよい。また、図示しないベクトル結合部334は、200次元の単語ベクトル A と、100次元の単語ベクトル B とを結合することにより、300次元の単語ベクトル C を生成してもよい。

20

【0079】

このように、生成プログラムは、第1のベクトルとは次元が異なる第2のベクトルを生成する処理をコンピュータに実行させてもよい。これにより、各コーパスモデルに対応する各単語ベクトルの重み付けを変化させることができる。

【0080】

また、2つのコーパスから生成されたコーパスモデル A 及びコーパスモデル B を用いる構成について説明したが、コーパスモデルの数はこれに限られず、3つ以上のコーパスモデルを用いるような構成であってもよい。

30

【0081】

図16は、実施例3におけるベクトル生成処理の別の一例を示す図である。図16に示すように、実施例3においては、図10に示す例と同様のコーパス A から生成されたコーパスモデル A 及びコーパス B から生成されたコーパスモデル B に加えて、コーパス N から生成されたコーパスモデル N がさらに用いられる。この場合において、図示しないベクトル抽出部333は、単語ベクトル A と、単語ベクトル B とに加えて、コーパスモデル N を用いて単語ベクトル N をさらに抽出してもよい。また、図示しないベクトル結合部334は、単語ベクトル A と、単語ベクトル B とに加えて、単語ベクトル N をさらに結合して、単語ベクトル C を生成してもよい。

40

【0082】

このように、生成プログラムは、単語に第3の変換パラメータをさらに適用して、単語に応じた第4のベクトルをさらに生成する処理をコンピュータに実行させてもよい。また、生成プログラムは、第1のベクトル及び第2のベクトルに加えて、さらに第4のベクトルに基づき、単語に応じた新たな第3のベクトルを生成する処理をコンピュータに実行させてもよい。これにより、3つ以上のコーパスの特徴を保持した単語ベクトルを利用することができる。

【0083】

ところで、ベクトルを結合すると、ベクトルの次元が大きくなり、検索処理に時間を要する場合がある。そこで、第1のベクトル、第2のベクトル及び第3のベクトルのうち、

50

いずれか1つ又は2つ以上の組み合わせを選択するような構成であってもよい。この場合において、選択されたベクトルに対応する学習モデルを用いて、受け付けた単語と複数の判別対象の文章との判別結果を取得するような構成であってもよい。これにより、処理負荷に応じて適切な大きさのベクトルを採用することができる。

【0084】

なお、実施の形態は日本語の文書を対象とする分散学習に限られず、例えば英語や中国語などのその他の言語の文書を用いてもよい。

【0085】

[ニューラルネットワーク]

また、事前学習処理には、例えばRNN（再帰型ニューラルネットワーク：Recurrent Neural Network）や、CNN（Convolutional Neural Network）などの任意のニューラルネットワークを用いることができる。また、学習の手法としては、誤差逆伝播（error Back Propagation：BP）法など、公知の様々な手法を採用することができる。なお、ニューラルネットワークは、例えば入力層、中間層（隠れ層）、出力層から構成される多段構成であり、各層は複数のノードがエッジで結ばれる構造を有する。各層は、「活性化関数」と呼ばれる関数を持ち、エッジは「重み」を持ち、各ノードの値は、前の層のノードの値、接続エッジの重みの値、層が持つ活性化関数から計算される。なお、計算方法については、公知の様々な手法を採用できる。

10

【0086】

[システム]

この他、上記文書中や図面中で示した処理手順、制御手順、具体的名称、各種のデータやパラメータを含む情報については、特記する場合を除いて任意に変更することができる。

20

【0087】

また、図示した各装置の各構成要素は機能概念的なものであり、必ずしも物理的に図示の如く構成されていることを要しない。すなわち、各装置の分散や統合の具体的な形態は図示のものに限られない。つまり、その全部または一部を、各種の負荷や使用状況などに応じて、任意の単位で機能的または物理的に分散・統合して構成することができる。例えば、図11に示すベクトル抽出部233と未知語ベクトル処理部237とを統合してもよい。また、図2に示す回答検索部135を、生成装置10とは別のコンピュータにより実現してもよい。すなわち、推論処理及び結合処理とがそれぞれ異なる筐体で実行されてもよい。さらに、各装置にて行なわれる各処理機能は、その全部または任意の一部が、CPUおよび当該CPUにて解析実行されるプログラムにて実現され、あるいは、ワイヤードロジックによるハードウェアとして実現され得る。

30

【0088】

[ハードウェア構成]

図17は、ハードウェア構成例を示す図である。図17に示すように、生成装置10は、通信インタフェース10a、HDD（Hard Disk Drive）10b、メモリ10c、プロセッサ10dを有する。なお、以下においては実施例1における生成装置10について説明するが、その他の実施例における生成装置も、同様の構成により実現できる。

40

【0089】

通信インタフェース10aは、他の装置の通信を制御するネットワークインタフェースカードなどである。HDD10bは、プログラムやデータなどを記憶する記憶装置の一例である。

【0090】

メモリ10cの一例としては、SDRAM（Synchronous Dynamic Random Access Memory）等のRAM（Random Access Memory）、ROM（Read Only Memory）、フラッシュメモリ等が挙げられる。プロセッサ10dの一例としては、CPU（Central Processing Unit）、DSP（Digital Signal Processor）、FPGA（Field Programmable Gate Array）、PLD（Programmable Logic Device）等が挙げられる。

50

【0091】

また、生成装置10は、プログラムを読み出して実行することで学習方法を実行する情報処理装置として動作する。つまり、生成装置10は、受付部131、単語抽出部132、ベクトル抽出部133、ベクトル結合部134、回答検索部135及び出力部136と同様の機能を実行するプログラムを実行する。この結果、生成装置10は、受付部131、単語抽出部132、ベクトル抽出部133、ベクトル結合部134、回答検索部135及び出力部136と同様の機能を実行するプロセスを実行することができる。なお、この他の実施例でいうプログラムは、生成装置10によって実行されることに限定されるものではない。例えば、他のコンピュータまたはサーバがプログラムを実行する場合や、これらが協働してプログラムを実行するような場合にも、本発明を同様に適用することができる。

10

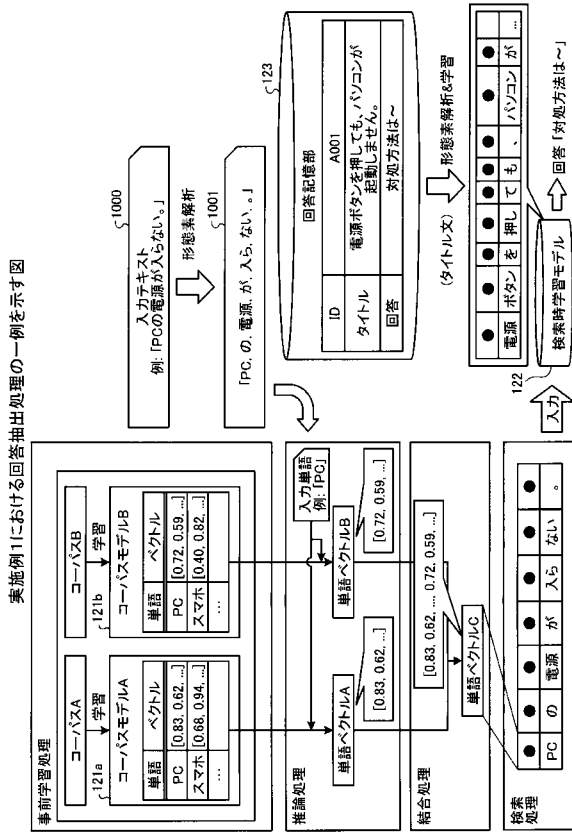
【符号の説明】

【0092】

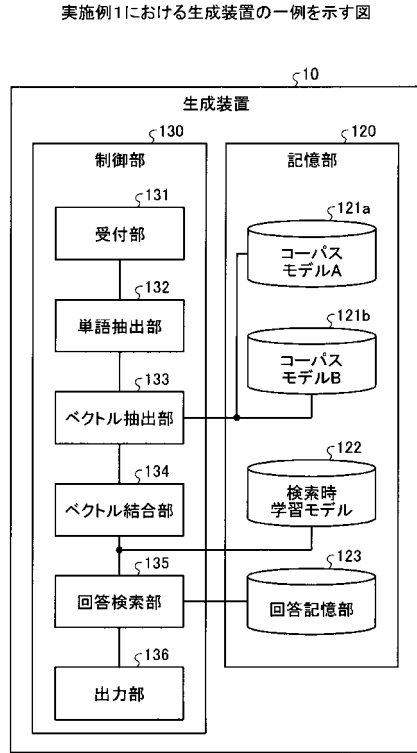
- 10、20 生成装置
- 120、220 記憶部
- 121a コーパスモデルA
- 121b コーパスモデルB
- 122 検索時学習モデル
- 123 回答記憶部
- 224 未知語ベクトル記憶部
- 130、230 制御部
- 131 受付部
- 132 単語抽出部
- 133、233 ベクトル抽出部
- 134 ベクトル結合部
- 135 回答検索部
- 136 出力部
- 237 未知語ベクトル処理部

20

【 図 1 】



【 図 2 】



【 図 3 】

実施例1におけるコーパスモデルの一例を示す図

121a

単語	ベクトル
PC	[0.83, 0.62, 0.49, ...]
スマホ	[0.68, 0.94, 0.51, ...]
ノートPC	[0.95, 0.34, 0.77, ...]
タブレット	[0.05, 0.64, 0.64, ...]
...	

【 図 5 】

実施例1における回答記憶部の一例を示す図

123

ID	タイトル	回答
A-001	電源ボタンを押しても、パソコンが起動しません。	対処方法は～
A-002	...	
...		
B-001		
...		

【 図 4 】

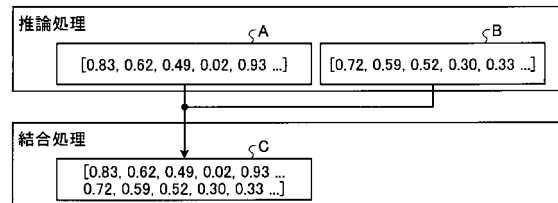
実施例1における検索時学習モデルの一例を示す図

122

ID	ベクトル
A-001	[0.56, 0.94, 0.22, ...]
A-002	[0.31, 0.58, 0.16, ...]
...	
B-001	[0.69, 0.34, 0.95, ...]
...	

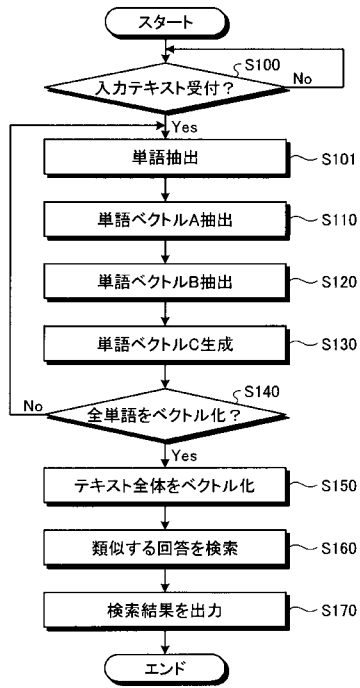
【 図 6 】

実施例1におけるベクトル結合処理の一例を示す図



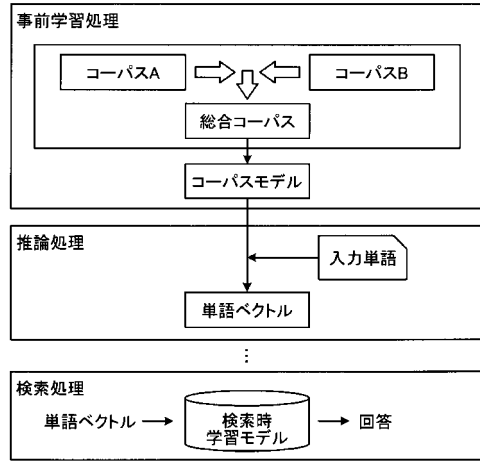
【 図 7 】

実施例1における回答抽出処理の一例を示すフローチャート



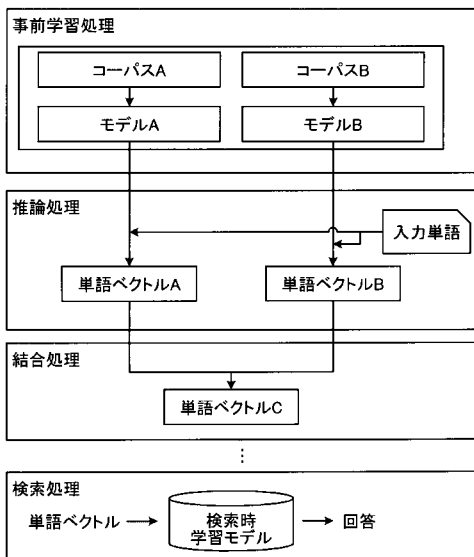
【 図 8 】

背景技術におけるベクトル生成処理の一例を示す図



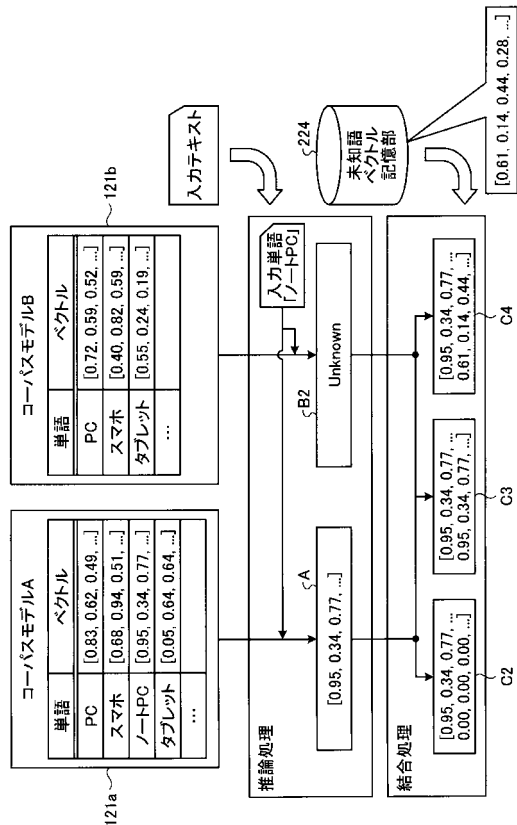
【 図 9 】

実施例1におけるベクトル生成処理の一例を示す図



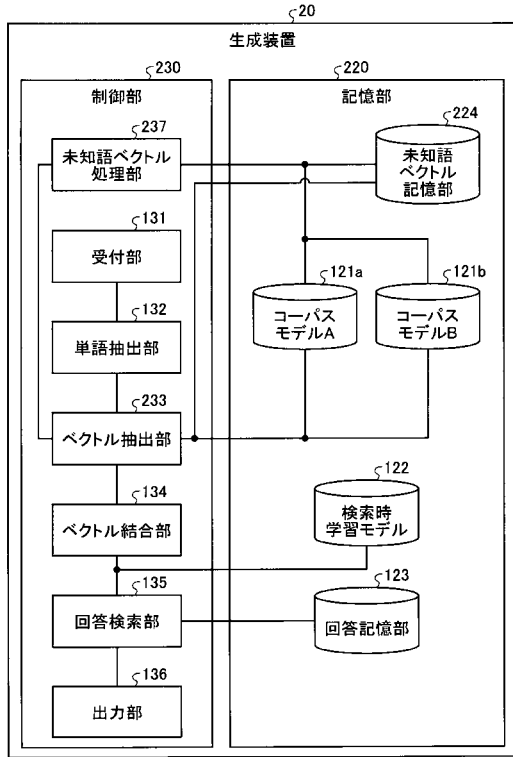
【 図 10 】

実施例2におけるベクトル生成処理の一例を示す図



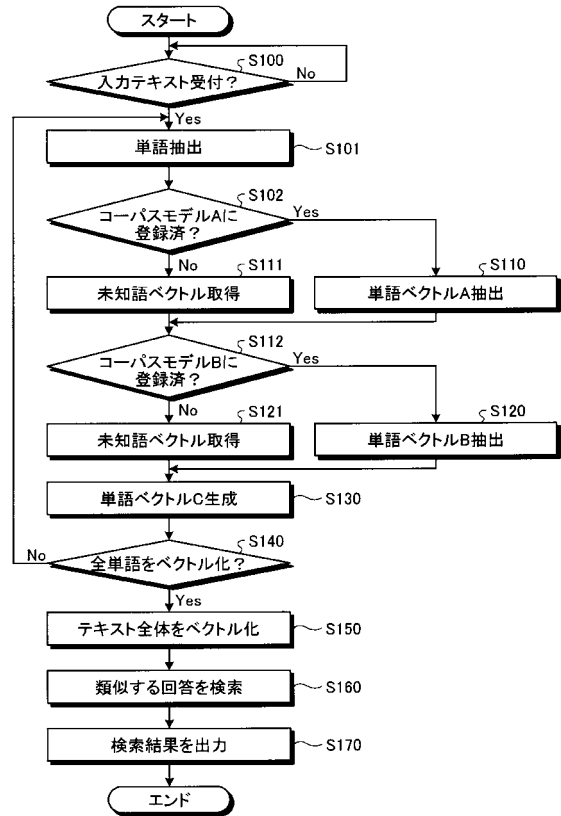
【 図 1 1 】

実施例2における生成装置の一例を示す図



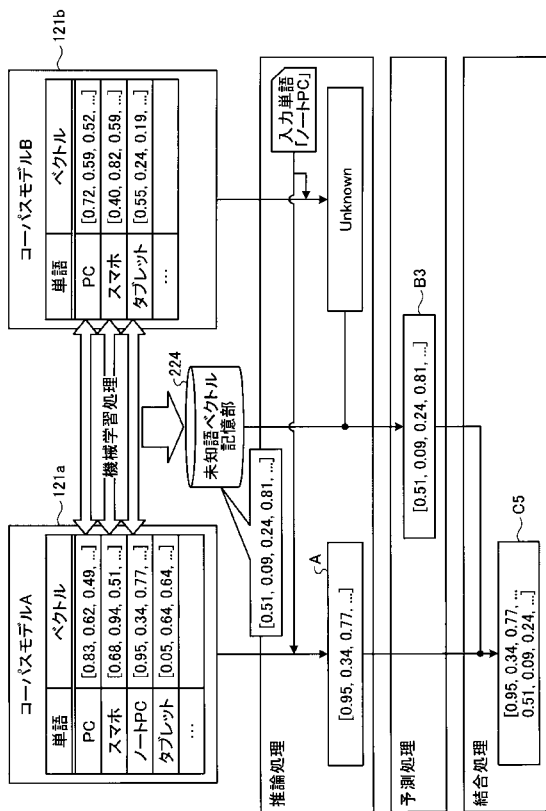
【 図 1 2 】

実施例2における回答抽出処理の一例を示すフローチャート



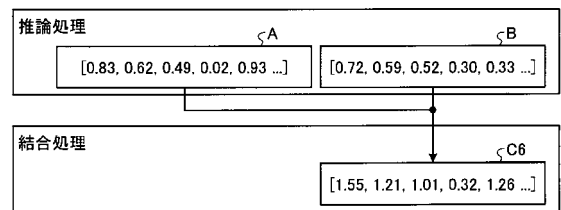
【 図 1 3 】

実施例2におけるベクトル生成処理の別の一例を示す図



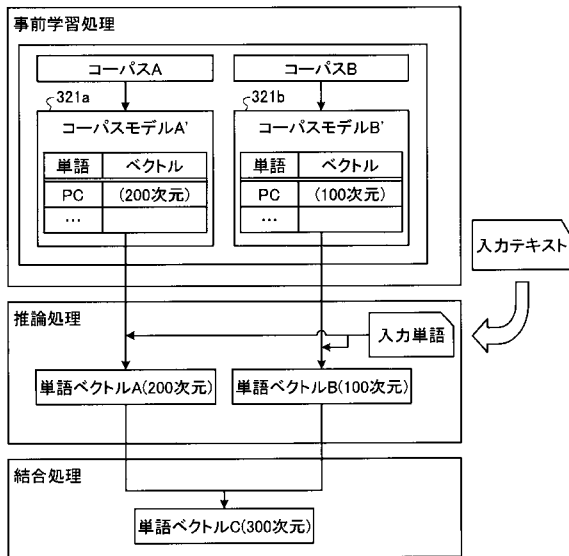
【 図 1 4 】

実施例3におけるベクトル結合処理の一例を示す図



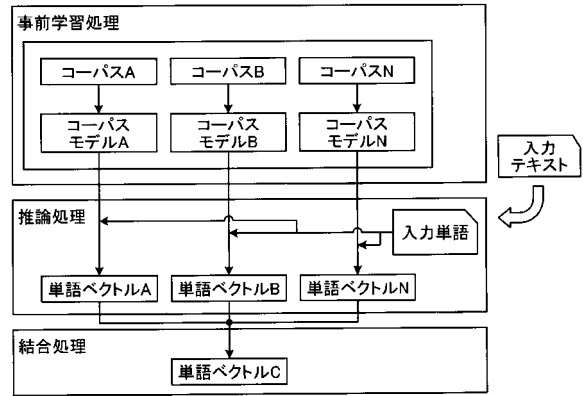
【 図 1 5 】

実施例3におけるベクトル生成処理の一例を示す図



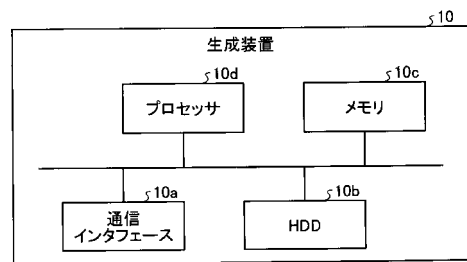
【 図 1 6 】

実施例3におけるベクトル生成処理の別の一例を示す図



【 図 1 7 】

ハードウェア構成例を示す図



フロントページの続き

- (72)発明者 戸田 隆道
神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
- (72)発明者 吉田 卓矢
神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
- (72)発明者 岩本 華代子
神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
- (72)発明者 高橋 直紀
神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
- Fターム(参考) 5B091 AA15 CA02 CC02 EA01 EA02