(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第4299963号
(P4299963)
(45)発行日　平成21年7月22日(2009.7.22)　(24)登録日　平成21年4月24日(2009.4.24)

(51)Int.Cl.　　F I
G06F 17/21 (2006.01)　　G06F 17/21 550A
G06F 17/30 (2006.01)　　G06F 17/30 170A
　　G06F 17/30 210D

請求項の数 6　(全 12 頁)

(21)出願番号　特願2000-302321 (P2000-302321)
(22)出願日　平成12年10月2日 (2000.10.2)
(65)公開番号　特開2002-117019 (P2002-117019A)
(43)公開日　平成14年4月19日 (2002.4.19)
審査請求日　平成19年10月1日 (2007.10.1)

(73)特許権者　398038580
ヒューレット・パッカード・カンパニー
HEWLETT-PACKARD COMPANY
アメリカ合衆国カリフォルニア州パロアルト　ハノーバー・ストリート　3000
(74)代理人　100081721
弁理士　岡田　次生
(72)発明者　清水　裕之
東京都杉並区高井戸東３丁目２９番２１号　日本ヒューレット・パッカード株式会社内

最終頁に続く

(54)【発明の名称】意味的まとまりに基づいて文書を分割する装置および方法

(57)【特許請求の範囲】
【請求項１】
電子化された文書の文末の左右に所定幅の窓を設定し、該左右の窓に含まれるタームの類似度を計算する手段と、
前記文書を分析し、該類似度に基づいて文末ごとの分割点尤度を求める手段と、
前記分割点尤度に基づいて前記文書を文書セグメントに分割する手段と、
を備え、
前記分割する手段は、分割された前記文書セグメントが指定サイズに基づいて定められるしきい値より大きいとき、該文書セグメント内で最もよい分割点尤度を持つ位置で該文書セグメントを分割し、前記分割された文書セグメントが前記指定サイズより予め定めた程度以上小さいとき、分割前の文書セグメントに戻り、次によい分割点尤度を持つ位置で該文書セグメントを分割する、文書分割装置。
【請求項２】
前記類似度を計算する手段は、cを文末位置として、複数（Ｌ）の異なる窓幅についてそれぞれ分割点尤度f(c)を計算し、こうして得られた複数の分割点尤度に基づいて総合的な分割点尤度F(c)を計算する、請求項１に記載の文書分割装置。
【請求項３】
分割された文書セグメント間の類似度を計算し、類似度が予め定めたしきい値以上の文書セグメントに関連づけリンクを形成するようプログラムされた請求項１に記載の文書分割装置。

【請求項４】
電子化された文書の文末の左右に所定幅の窓を設定し、該左右の窓に含まれるタームの類似度を計算するステップと、
前記文書を分析し、前記類似度に基づいて文末ごとの分割点尤度を求めるステップと、
前記分割点尤度に基づいて前記文書を文書セグメントに分割するステップと、
をコンピュータに実行させるためのプログラムを記録したコンピュータ読み取り可能な記録媒体であって、
前記分割するステップは、分割された前記文書セグメントが指定されたサイズに基づいて定められるしきい値より大きいとき、該文書セグメント内で最もよい分割点尤度を持つ位置で該文書セグメントを分割し、分割された文書セグメントのサイズが指定サイズより予め定めた程度以上小さいとき、分割前の文書セグメントに戻り、次によい分割点尤度を持つ位置で該文書セグメントを分割する、前記記録媒体。
【請求項５】
前記類似度を計算するステップは、cを文末位置として、複数（Ｌ）の異なる窓幅についてそれぞれ分割点尤度f(c)を計算し、こうして得られた複数の分割点尤度に基づいて総合的な分割点尤度F(c)を計算する、請求項４に記載の記録媒体。
【請求項６】
前記プログラムは、分割された文書セグメント間の類似度を計算し、類似度が予め定めたしきい値以上の文書セグメントに関連づけリンクを形成するステップをコンピュータに実行させる、請求項４に記載の記録媒体。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】
この発明は、文書の分割技術に関し、より具体的には意味的まとまりにしたがって文書を分割する文書分割技術に関する。
【０００２】
【従来の技術】
文書検索により検索された文書が複数の話題を含むような大きな文書の場合、ユーザは表示された文書からユーザにとって必要な部分を探す必要がある。この場合、予め話題ごとに分割した文書セグメントを検索対象にすることができれば、直ちにその文書セグメントを表示することができ、ユーザがさらに必要な部分を探す必要がなくなる。このように文書を話題ごとに分割することができると、様々な文書処理が容易になる。
【０００３】
文書分割方法としては、特開平１１－２４２６８４号公報、特開２０００－２３５５７４号公報、特開平１０－７２７２４号公報等に記載がある。特開平１１－２４２６８４号公報は、文書を隣接文間の関連だけでなく、広域的な関連も考慮に入れた文書分割装置を提案し、特開２０００－２３５５７４号公報は、文書を形式段落等で分割し、段落間の関連度を要素とするような正方行列から分割点を求める方法を提案し、特開平１０－７２７２４号公報は、複数の窓から各位置における関連度を計算し、各階層ごとに話題境界を求め、それらを統合することによって、話題境界を認定していく方法を提案している。
【０００４】
【発明が解決しようとする課題】
上記のような方法を用いることによって文書を話題ごとに分割することは可能である。しかしこれらの方法はサイズなどを考慮に入れてないため、特に携帯電話やPDAなどの表示画面が小さいなどのリソースの制限がある機器では、分割された文書セグメントを表示する際にユーザはスクロール等の操作を行う等の必要があったり、文書セグメントのサイズが機器の記憶領域の制限を越えていることがある。このように従来の文書分割手法によって分割された文書セグメントは、必ずしもユーザや端末装置にとって好ましい分割単位にはなっていない。
【０００５】

したがって、文書を意味的なまとまりおよび指定された文書セグメント・サイズに従って分割する手法に対する必要性がある。さらには、携帯電話、PDA等の画面の小さい機器でもユーザに読み易い文書セグメント群を提供する技術に対する必要性がある。

【０００６】

【課題を解決するための手段】

この発明の一つの側面によると、文書分割装置は、電子化された文書を分析し、意味的まとまりに基づいて文末ごとの分割点尤度を求める手段と、前記分割点尤度および指定された文書セグメント・サイズに基づいて前記文書を文書セグメントに分割する手段と、を有する。

【０００７】

また、この発明のもう一つの側面によると、文書分割装置は、電子化された文書を分析し、意味的まとまりに基づいて文末ごとの分割点尤度を求める手段と、前記分割点尤度に基づいて前記文書を文書セグメントに分割する手段と、を備え、分割された前記文書セグメントが指定されたサイズに基づいて定められるしきい値より大きいとき、該文書セグメント内で最もよい分割点尤度を持つ位置で該文書セグメントを分割するようプログラムされている。

【０００８】

この発明の一つの形態によると、文書は指定されたサイズと同程度のサイズを持つ文書セグメント群に分割される。まず、文書中の各文末位置においてその前後に設定された窓に含まれる文書部分間の類似度を計算し、類似度曲線を求める。得られた類似度曲線から各位置における分割点尤度を計算する。そして分割点尤度のよい位置から順に分割点として文書を分割していき、全ての文書セグメントが指定されたサイズと同程度のサイズになるまで分割していく。

【０００９】

【発明の実施の形態】

次に図面を参照して発明の一つの実施形態を説明する。図１は、この発明の一実施例のシステムの全体的な構成を示す機能ブロック図である。この実施例は、ハードウェア的には汎用のコンピュータ、ワークステーションまたはパーソナルコンピュータで構成される。この発明を実現するコンピュータ・プログラムを汎用のコンピュータ上で走らせることにより、この発明を実施することができる。図１に示す各ブロックは、このコンピュータ・プログラムによって実現される機能を示す。

【００１０】

分割対象となる電子化された文書１を受け取ると、形態素解析部２は、文書中の単語を切り出し、各単語に品詞情報を付加する。窓サイズ設定部３は、文書に含まれる隣接する文章の間の類似度を測定するための窓サイズを設定する。この窓サイズは、文末位置から左右に予め決められた長さとする。類似度測定部４は窓サイズ設定部３で設定された各位置における左右窓に含まれる文書部分間の類似度を測定し、類似度曲線を生成する。

【００１１】

分割点尤度計算部５は、類似度測定４で求められた類似度曲線から各文末位置における分割点尤度を計算する。分割点決定部６は、分割点尤度計算部５で求められた分割点尤度を用いて、最も大きな文書セグメントの中でもっともよい分割点尤度を持った位置を分割点として選択する。文書１が分割されていないプロセスの開始部分においては、文書１の全体が最も大きな文書セグメントとなる。

【００１２】

サイズ比較部１１は、分割点決定部６で決定された文書セグメントの候補のサイズを出力先の機器が指定する文書セグメントのサイズに基づいて定めたサイズしきい値と比較し、文書セグメント候補のサイズがこのしきい値よりも大きいときは、その文書セグメント候補の中で最もよい分割点尤度を持つ位置を分割点として選択する。文書セグメント生成部７は、こうして得られた文書セグメント候補を文書セグメント集合とし、集合内の全ての文書セグメントが指定されたサイズより小さくなるまで分割点決定部６に戻り、サイズ比

較部１１による処理を受ける。

【００１３】

関連度計算部８は文書セグメント生成部７で生成された文書セグメント間の類似度を計算し、その類似度を用いて文書セグメント間の関連付けを行う。リンク生成部は、関連度計算部８による計算結果に基づいて内容的に関連性の高い文書セグメント間にリンクを生成する。こうしてリンクを生成された文書セグメントが要求元のＰＤＡ、携帯電話などの端末装置に文書セグメントを送信する。

【００１４】

一つの実施形態では、この発明の文書分割装置は、インターネット環境で使われる。たとえばユーザがＰＤＡを用いてインターネット経由でウェブ・サイトにアクセスし、データを検索し、その結果をＰＤＡのブラウザに表示する。この場合、ウェブ・サイトは、ＰＤＡに送信する文書をこの発明の文書分割装置により、ＰＤＡの表示スクリーンに合わせたサイズの文書セグメントに分割して送信する。文書セグメントは、ＨＴＭＬ文書に変換され、関連文書セグメントに対するハイパーリンクを埋め込んでインターネット経由でＰＤＡに送信される。文書セグメントのサイズは、ＰＤＡにおける表示サイズに合わせられており、ボタンをクリックする操作により、次の文書セグメントまたは意味的な関連の強い文書セグメントに飛ぶことができるので、小さな表示スクリーン上でも快適に文書を見ることができる。

【００１５】

図２および３は、文書分割アルゴリズムのフローチャートである。図４は、文書セグメント間関連付けアルゴリズムのフローチャートを示す。まず図２を参照すると、Ｎ個の文書、Ｍ個の単語を含む電子文書Ｄおよび最適セグメントサイズＳを受け取る（202）。ここで最適セグメントサイズとはユーザによって指定された文字数、またはＰＤＡ、携帯電話などの表示装置の表示文字数から規定されるもので、例えば１００文字表示できる端末装置の場合だと最適セグメントサイズＳは１００文字が選ばれる。

【００１６】

次のステップ203では、入力された電子文書Ｄに対して形態素解析を行い、文書中の単語を切り出し、その単語に品詞情報を与える。そしてその中から２回以上現れる名詞をタームt$_i$として取り出し、タームリストT(=t$_1$，t$_2$，t$_3$，…,t$_n$)を生成する（204）。

【００１７】

続いてステップ205で、窓の幅Ｂを設定する。窓幅Ｂは、最初は、文書に含まれる単語の数Ｍの、たとえば1/5に設定する。こうして文書に含まれる文章のそれぞれの文末位置の左右に幅Ｂの窓を設定する（206）。そして先ほど求めたタームを要素とするベクトルW=(w$_{t1}$，w$_{t2}$，w$_{t3}$，…、w$_{tn}$)　を左右の窓に含まれる文書部分からそれぞれ求める。ここでw$_{t1}$は窓の中に含まれる文書中におけるタームt$_1$の出現頻度である。求まった２つのベクトルから余弦測度sim(b$_l$,b$_r$)を求め、それをその位置における類似度とする（207）。余弦測度は次に示す（１）式で求められる。

【００１８】

【数１】

$$sim(b_l, b_r) = \frac{W_{bl} \times W_{br}}{\sqrt{W_{bl}^{\ 2} + W_{br}^{\ 2}}} \qquad \cdots\cdots\cdots\cdots (1)$$

【００１９】

ここで、b$_l$，b$_r$はそれぞれ左の窓、右の窓に含まれる文書部分を表す。また、W$_{bl}$，W$_{br}$はそれぞれ左の窓、右の窓に現れるタームの出現頻度を表すベクトルである。（１）式で求められる類似度は、左右の窓に共通して現れるタームの数が多いほど大きな値（最大１）になり、共通のものがないときには０になる。つまり、この値が大きいときは左右の窓で共通の話題を扱っている可能性が高くなり、小さいときは話題の境界である可能性が高い。

【００２０】
図２に示すサフィックスｉは、文書に含まれる文章の番号を示し、文書の頭から1,2,3・・・・NまでのＮ個の文章が含まれるものとする。ｉがＮに達するまで（ステップ209がＮＯになるまで）、ｉをインクリメントし（211）、類似度の計算を各文末位置に対して行う。こうして、文書についての類似度曲線が得られる。図５から８は、次の表に示す入力文書に対する類似度曲線を示す。
【００２１】
【表１】
The community of mostly volunteer programmers that has built Linux into a formidable operating system is getting some help from computer industry giants. International Business Machines Inc., Intel Corp., Hewlett-Packard Co. and NEC Corp. are announcing Wednesday that they will create a laboratory with an investment of several million dollars where programmers can test Linux software on the large computer systems that are common in the corporate world. The lab is expected to open by the end of the year near Portland, Ore. Linux is an "open source" operating system that anyone can modify, as long as the modifications are made available for free on the Internet. It has a devoted following among programmers, who collaborate on software projects over the Web. These software engineers can usually only test software on their own desktop computers, part of the reason Linux is now rarely used on larger computers. "The Open Source Development Lab will help fulfill a need that individual Linux and open source developers often have: access to high-end enterprise hardware," said Brian Behlendorf, creator of the open source Web server software Apache. Irving Wladawsky-Berger, the head of IBM's Linux group, said the lab would help companies run hardware from different vendors together, as well as let run "clusters" of computers working as one. The four main sponsors said they will contribute several millions of dollars to the project. The lab is also backed by smaller companies that specialize in Linux products, like Red Hat Inc., Turbolinux Inc., Linuxcare Inc. and VA Linux Systems Inc., as well as Dell Computer Corp. and Silicon Graphics Inc. The founding companies said the lab will be run by a nonprofit organization that will select the software projects that gain access to the lab in an "open, neutral process." Linux is seen as an alternative to proprietary operating systems like Microsoft's Windows and Apple OS. Its backers say the publicly available source code, or software blueprint, makes it more flexible and reliable. Analyst Bill Claybrook at Aberdeen Group said the project sponsors are backing Linux because it gives them a chance to influence an operating system for their computers. "These companies see that they can play a much more important role in developing Linux than they can in, let's say Windows, because Microsoft pretty much decides what to put in Windows," he said.
【００２２】
図５から８における横軸は各文末の位置を表し、縦軸は類似度を表す。また、図５から８における窓幅は、左右の窓にそれぞれに含まれる単語数である。
【００２３】
こうして求められた類似度曲線から各文末位置ｃに対して、分割点尤度ｆ(c)を求める。分割点尤度f (c)は以下の式から求められる。ステップ209において、ｉ＝Ｎになると、すなわち、文書に含まれるすべての文章についてＢ＝Ｍ／５という条件下での類似度曲線が得られると、ｉ＝１にセットし（212）、最初の文章の文末位置に対する分割点尤度を計算する（213）。この計算は、ｉがＮに達するまでｉをインクリメントさせて（216）、繰り返される。
【００２４】

【数２】

$$f(c) = \alpha \times fs(c) + \beta \times fg(c) \quad \cdots\cdots\cdots\cdots (2)$$

$$fs(c) = 1 - s(c) \quad \cdots\cdots\cdots\cdots (3)$$

$$fg(c) = \frac{(s(c-) - s(c)) + (s(c+) - s(c))}{2} \quad \cdots\cdots\cdots\cdots (4)$$

【００２５】

ここで、s(c)は各文末位置ｃにおける類似度、s(c-)は文末位置ｃの１つ前の文の文末位置における類似度、s(c+)は文末位置ｃの１つ後ろの文の文末位置における類似度であり、α、βはパラメータで実験によって求まるものである。

【００２６】

（２）式の分割点尤度は類似度が極小な位置や類似度の遷移が大きいときに大きな値をとり、類似度が大きい、あまり類似度の遷移がない時に小さな値をとるようになる。

【００２７】

ｉがＮに達すると（ステップ215の判断がＮＯになると）、窓幅Ｂを最初の設定の１／２にセットしてステップ206以下のプロセスを繰り返す。そしてこの処理が完了すると、さらにその１／２の窓幅に設定してステップ206以下のプロセスを繰り返す。この繰り返し処理は、ｊが、類似度曲線の総数であるＬに達するまで、すなわちステップ217における判断がNOになるまで、L回繰り返される。

【００２８】

こうしてそれぞれの窓幅に対して求められたＬ個の分割点尤度ｆ（ｃ）を用いて、入力文書Ｄに対する総合的な分割点尤度Ｆ（ｃ）を求める。

【００２９】

【数３】

$$F(c) = \sum^{L} \gamma_j \times \log f_j(c) \quad \cdots\cdots\cdots\cdots (5)$$

【００３０】

ここでf_j（ｃ）はｉ番目の類似度曲線から求まった分割点尤度ｆ（ｃ）であり、γ_jは各類似度曲線に対する重み係数であり、γ_jとしては例えば、１番大きな窓幅の分割点尤度に対して１、その次に対して１／２、その次に１／４と与える。以下、この実施例では文書の分割は、式（５）で求めた分割尤度曲線をもとに行う。図９は、こうして求められた分割尤度曲線を示す。

【００３１】

次に図３に示すプロセスに移る。分割前の文書全体を文書セグメントR_0で表すことにする（301）。ステップ302において、文書セグメント集合Ｒの中から最も大きいサイズのセグメントR_iを選択する。初期状態では、文書セグメント集合Ｒは文書全体である文書セグメントR_0だけを要素とする集合である。

【００３２】

ステップ303に移り、選択された文書セグメントR_iのサイズをセグメントサイズ閾値Th_{size}と比較する。セグメントサイズ閾値Th_{size}は、指定されたサイズすなわち最適セグメントサイズＳに基づいて決められる。例えば、セグメントサイズ閾値Th_{size}を最適セグメントサイズＳの１．１倍にすると、最適セグメントサイズを１０％超えるサイズまでの文書セグメントを許容するようことになる。

【００３３】

セグメントＲｉのサイズが閾値Th_{size}より大きいときは、ステップ305に進み、セグメントＲｉ内で最もよい分割点尤度ｆを持つ文末位置ｃを分割点として選択する。ステップ307において、その文書セグメントR_iを分割し、新しい文書セグメントＲｌ’、Ｒｒ’を生成する。分割された文書セグメントＲｌ’、Ｒｒ’が指定されたサイズＳより小さす

ぎる場合（308）は、分割前の文書セグメントＲｉに戻し、その中で次によい分割点尤度を持つ位置を分割点として選択し、セグメントに分割する（309）。

【００３４】

こうして、指定サイズＳにたいして小さすぎないセグメントＲｌ’またはＲｒ’が得られると、文書セグメント集合ＲからＲ$_{i}$を削除し、新たにＲｌ’、Ｒｒ’を文書セグメント集合Ｒに加える（311）。

【００３５】

次いで、ステップ302に戻り、全ての文書セグメントの中で最も大きい文書セグメントのサイズがセグメントサイズ閾値Ｔｈ$_{size}$より小さくなるまで、すなわちステップ303の判断がＮＯになるまで、閾値Ｔｈ$_{size}$より大きいサイズの文書セグメントについてステップ305以下のプロセスが繰り返えされる。この様に分割点尤度のよいものから順番に分割していくことによって、文書の大局的な話題の区切りを保持しつつ同程度のサイズをもつ文書セグメントを生成していくことが可能になる。

【００３６】

次の表２に最適セグメントサイズを４００文字と指定し、表１の入力文書Ｄを分割した際の文書セグメント群を示す。各セグメントのサイズが指定された通り、４００文字程度になっている。また、表３に文書セグメントをマークアップ言語の形で表した例を示す。

【００３７】

【表２】

文書セグメント１

The community of mostly volunteer programmers that has built Linux into a formidable operating system is getting some help from computer industry giants . International Business Machines Inc. , Intel Corp. , Hewlett-Packard Co. and NEC Corp . are announcing Wednesday that they will create a laboratory with an investment of several million dollars where programmers can test Linux software on the large computer systems that are common in the corporate world .

文書セグメント２

The lab is expected to open by the end of the year near Portland , Ore . Linux is an " open source " operating system that anyone can modify , as long as the modifications are made available for free on the Internet . It has a devoted following among programmers , who collaborate on software projects over the Web . These software engineers can usually only test software on their own desktop computers , part of the reason Linux is now rarely used on larger computers .

文書セグメント３

" The Open Source Development Lab will help fulfill a need that individual Linux and open source developers often have : access to high-end enterprise hardware , " said Brian Be , creator of the open source Web server software Apache . Irving Wladawsky-Berger , the head of IBM's Linux group , said the lab would help companies run hardware from different vendors together , as well as let run " clusters " of computers working as one .

文書セグメント４

The four main sponsors said they will contribute several millions of dollars to the project . The lab is also backed by smaller companies that specialize in Linux products , like Red Hat Inc. , Turbolinux Inc. , Linuxcare Inc. and VA Linux Systems Inc. , as well as Dell Computer Corp. and Silicon Graphics Inc . The founding companies said the lab will be run by a nonprofit organization that will select the software projects that gain access to the lab in an " open , neutral process . "

文書セグメント５

Linux is seen as an alternative to proprietary operating systems like Microsoft'

s Windows and Apple OS . Its backers say the publicly available source code , or software blueprint , makes it more flexible and reliable . Analyst Bill Claybrook at Aberdeen Group said the project sponsors are backing Linux because it gives them a chance to influence an operating system for their computers . " These companies see that they can play a much more important role in developing Linux than they can in , let's say Windows , because Microsoft pretty much decides what to put in Windows , " he said .

【００３８】

【表３】

```
<?xml version="1.0" encoding="Shift_JIS" ?>
- <text>
  - <block id="0">
      <start_sentence href="input.xml#xpointer(@id=s0)" />
      <end_sentence href="input.xml#xpointer(@id=s1)" />
    - <block_body>
        The community of mostly volunteer programmers that
        has built Linux into a formidable operating system is
        getting some help from computer industry giants .
        International Business Machines Inc. , Intel Corp. ,
        Hewlett-Packard Co. and NEC Corp. are announcing
        Wednesday that they will create a laboratory with an
        investment of several million dollars where
        programmers can test Linux software on the large
        computer systems that are common in the corporate
        world .
      </block_body>
    </block>
  </text>
```

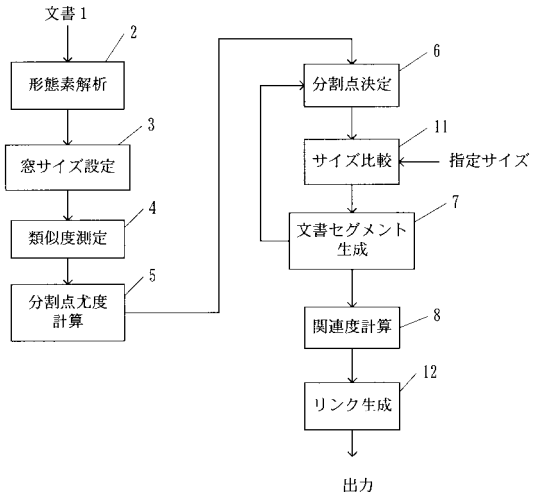
【００３９】

次に図４を参照して、文書セグメントの関連付け処理を説明する。以上のプロセスによって求められた文書セグメント間、または重要語と文書セグメント間の類似度ｑを（１）式を用いて計算し（402）、類似度ｑが関連閾値$Th_{relevant}$より大きい時は（403）、文書セグメント間で似たような話題について書いてあると判断し、関連付けリンクを挿入する（405）。関連閾値$Th_{relevant}$としては例えば０．５を用いる。また、ユーザがよく関連しているセグメントだけの表示を希望する場合や関連するセグメント全ての表示を希望する場合があるので、この発明の一実施態様では、関連閾値$Th_{relevant}$はユーザが指定

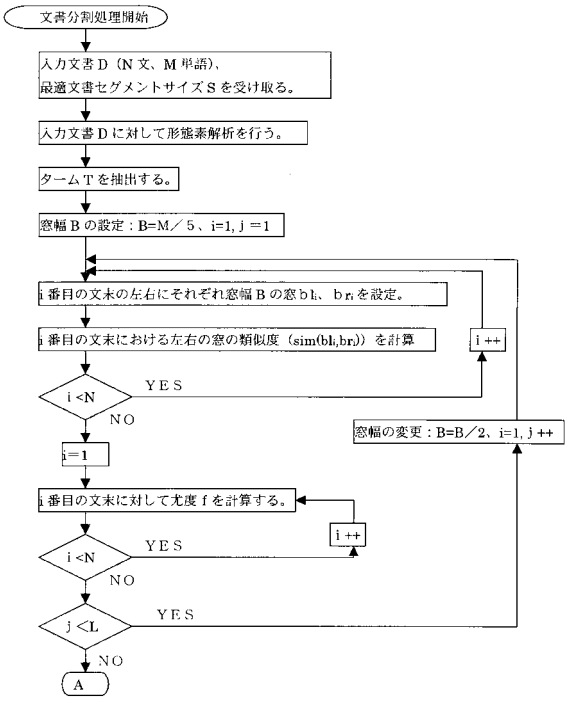
するようにする。
【００４０】
話題的に類似性のある文書セグメント間のハイパーリンク化はマークアップ言語でそれぞれの文書セグメントに埋め込まれる。また、リンク先としては１つの文書セグメントに限らず、複数の文書セグメントに対してはられる。例えば、文書セグメントを表すマークアップ言語としてＸＭＬのＸｐｏｉｎｔｅｒを用いれば複数の文書セグメントに対してリンクをはることができ、１つの文書セグメントから複数の関連セグメントを表示する等の機構がブラウザ上で実装可能になる。
【００４１】
以上に具体的な実施例について述べた本発明は、英語文書のみを対象とするわけでなく、日本語等の他言語文書に対してもその言語の形態素解析を行えば、同様な処理で文書分割を行うことができる。
【００４２】
本発明では、文書を指定されたサイズと同程度の文書セグメントに分割するので、携帯端末等の小さな画面でも、ユーザに対して以下に示すように効率よく文書を提示することができる。文書セグメントは画面サイズに合わせて生成できるので、ユーザは一目でその文書セグメントが必要かそうでないかを判断することができる。一実施形態では文書セグメントを画面サイズの合わせて生成できるので、文書を表示する際に文書セグメント単位でスクロールができる。
【００４３】
一実施の形態では、話題として類似する文書セグメント間に関連付けを行っているので、ユーザは簡単に関連する別の文書セグメントにアクセスすることができる。文書を表示する際、文書全体ではなく文書セグメント毎に表示できるので、表示端末では大きな記憶容量を必要としない。文書を携帯端末に表示する際、文書セグメントごとに転送できるので、パケットサイズなどの通信上の制限やハードウェアの制限を考慮して転送することができる。検索結果を文書セグメント単位で提示することによって、ユーザは直ちに必要な文書部分を読むことができる。
【００４４】
自動抽出された文書セグメントは意味的なまとまりを表しているので、文献（亀田雅之 1997.“段落間及び文間関連度を利用した段落シフト法に基づく重要文抽出” 情報処理学会自然言語処理研究会報告，119-126．121-17.）等の方法を用いて各セグメントに対して重要語、重要文抽出、もしくは文献（仲尾由雄 1998.“文書の意味的階層構造の自動認定に基づく要約作成” 言語処理学会第4回年次大会併設ワークショップ「テキ スト要約の現状と将来」論文集，72-79.）等を用いて要約文生成を各セグメントに対して行い、それらを提示することによって、ユーザが容易にその文書の概略の理解、斜め読みができる。
【００４５】
以上にこの発明を具体的な実施例について説明したが、この発明はこのような実施例に限定されるものではない。
【図面の簡単な説明】
【図１】　この発明の一実施例の文書分割装置の全体的なブロック図。
【図２】　文書分割アルゴリズムの前半部を示すフローチャート。
【図３】　文書分割アルゴリズムの後半部を示すフローチャート。
【図４】　文書セグメント間の関連付けを行うアルゴリズムのフローチャート。
【図５】　窓幅を４８０単語にしたときの類似度曲線を示す図。
【図６】　窓幅を２４０単語にしたときの類似度曲線を示す図。
【図７】　窓幅を１２０単語にしたときの類似度曲線を示す図。
【図８】　窓幅を６０単語にしたときの類似度曲線を示す図。
【図９】　分割点尤度曲線を示す図。
【符号の説明】

2　形態素解析部
3　窓サイズ設定部
4　類似度測定部
5　分割点尤度計算部
6　分割点決定部
１１　サイズ比較部
7　文書セグメント生成部
8　関連度計算部
１２　リンク生成部

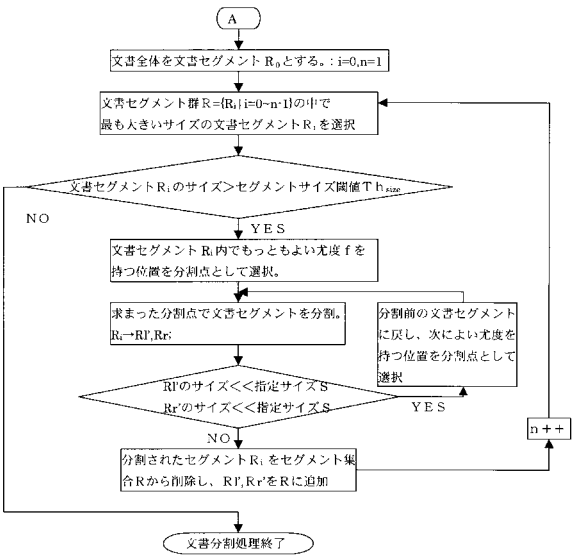
【図１】



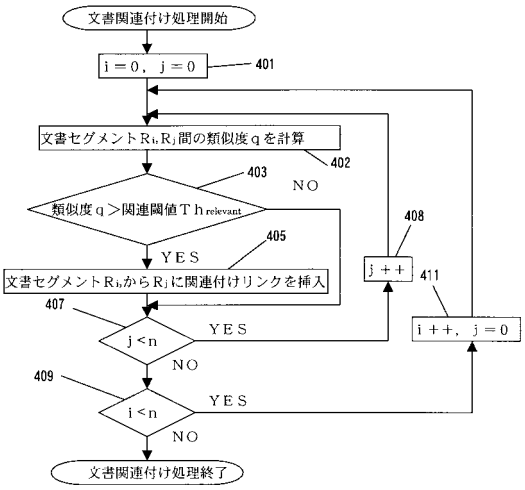
【図２】



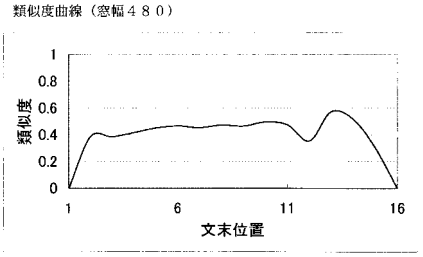
【図３】



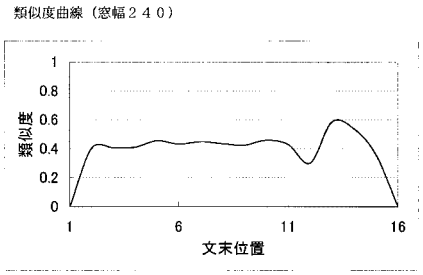
【図４】



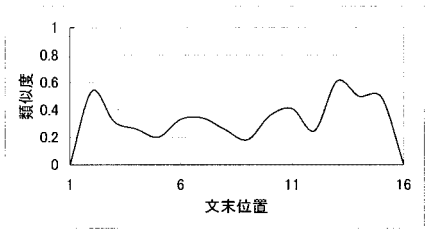
【図５】



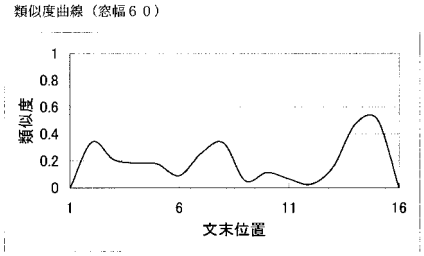
【図６】



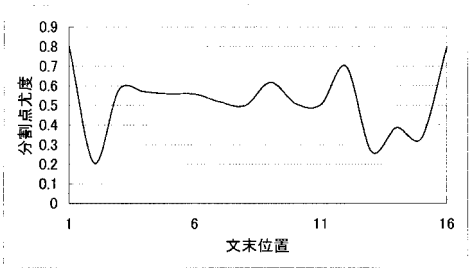
【図７】



【図８】



【図９】



フロントページの続き

(72)発明者　中川　真也
東京都杉並区高井戸東３丁目２９番２１号　日本ヒューレット・パッカード株式会社内

審査官　成瀬　博之

(56)参考文献　特開平１１－１８４８６６（ＪＰ，Ａ）
特開平０６－２０８５８２（ＪＰ，Ａ）
山田拓也 他，ユーザの意図を反映したシナリオに基づいたプレゼンテーション・スライド構成支援，情報処理学会研究報告，日本，社団法人情報処理学会，１９９９年１０月１５日，Vol99，No85，４７－５４頁

(58)調査した分野(Int.Cl.，ＤＢ名)
G06F 17/21-17/30