

WebDigest: 視覚補助のためのウェブページ要約技術

前田 潤治 小林 真
日本アイ・ビー・エム (株) 東京基礎研究所
〒242-8502 神奈川県大和市下鶴間1623-14
e-mail: {maeda, mkobaya}@jp.ibm.com

あらまし

ウェブによるインターネットへのアクセスが日常化するにつれて、特に高齢者や障害者のようないわゆる情報弱者に対する配慮がより重要となる。ここでは、特に高齢者にとってわかりやすいようなウェブブラウザの視覚的な表現の工夫に関して述べる。この手法は、ブラウザにおけるページレイアウトを保存しながら、ページ内容のダイジェストをおこない簡潔な表現を実現する。システム評価により、既存の手法に比べてこの方式が有効であることがわかった。

キーワード:

アクセシビリティ、高齢者向けユーザインターフェイス、ユーザ適合化、ウェブダイジェスト、ページレイアウト保存

WebDigest: Visually Enhanced Web Page Presentation through Page Digesting

Junji Maeda and Makoto Kobayashi
IBM Research, Tokyo Research Laboratory
1623-14, Shimotsuruma, Yamato, Kanagawa 242-8502, Japan
e-mail: {maeda, mkobaya}@jp.ibm.com

Abstract

As the Web becomes a platform for social activities, it is increasingly important for people unfamiliar with information technologies, such as the elderly and those with disabilities, to have easy access to the Information Superhighway.

In this paper, we describe a mechanism for modifying the visual formats of existing Web pages, so that they become more comprehensible, especially for elderly people. The mechanism digests and simplifies the Web contents while preserving their layout. Evaluation of the method with senior users showed that this approach is more effective than conventional ones.

Keywords

accessibility, user interface for aged people, user adaptation, Web digest, layout preservation

1. はじめに

ウェブによるインターネットへのアクセスが重要度を増すにつれ、高齢者や障害者などのいわゆる情報弱者が締め出される危険性が指摘されている。このような問題に対処すべく、情報アクセシビリティという視点から、W3CのWAI[1]に代表されるようなさまざまなアクションがなされてきているが、まだ十分とはいえない。

情報弱者の中でも、特に高齢者に関しては、これから急速に数が増えていき、対応がより緊急な課題となる。加齢には知覚的・物理的・認知的なさまざまな機能低下を伴うが、この中でも視覚機能の低下が、ウェブからの情報獲得にとって直接的な問題となる。

ウェブ情報の加齢に対する視認性向上に関しては、主にフォントサイズや行間の調節(増大)が有効であることが、さまざまな研究から報告されている[2][3]。フォントの大きさの調節に関しては、通常のブラウザの基本機能として提供されている。

また、別の方法としては、Windows98のオプション[4]、IBM ThinkPadのトラックポイントの拡張機能[5]、あるいはZoomText[6]などのアプリケーションによって実現される、虫眼鏡を模した部分的画面拡大機能を用いることもできる。

しかし、フォントサイズを大きくした場合には、ページが大きくなって一覧できる内容が少なくなり、全体的レイアウトも崩れて一覧性が著しく悪くなる。部分的な画面拡大の手法も、画面の一部だけが見える状態になるために、やはり全体像の把握が困難である。

高齢者は短期記憶が衰えやすく[7]、一覧性が損なわれることによる悪影響が大きい。

本論文では、これらの問題を解決するために、必要なフォントサイズを用いつつ一覧性を確保する手法を提案し、実験によってその有用性を確かめる。

2. 従来手法との比較

本手法を詳しく述べる前に、まず従来の手法をもう少し詳しくみていく。

2.1 ブラウザ機能によるフォント拡大

一般的に用いられている、Internet ExplorerおよびNetscape Navigatorといったブラウザにも、フォントのサイズ変更機能が用意されている。いずれの場合でも、文字列に関しては特に指定がない限りはウィンドウの幅に応じて折り返しが発生する。そのため、フォントサイズが大きくなっ

た場合には単に各行の高さが広がるのみならず、折り返しの回数も増加してページが縦方向に広がり、これによってスクロール回数が増加してしまう。また、ページ作成者が意図していたようなレイアウト構成を損なうことにもなる。

2.2 部分的な画面拡大機能

虫眼鏡を模した部分的画面拡大機能としては、画面が固定的に分割(もとの表示領域と、その一部を拡大する表示領域)される場合と、拡大表示領域がマウスカーソルの動きにつれて動く場合とがある。

Windows98には、前者の固定的拡大表示機能が用意されている。また、IBM ThinkPadには、TrackPointのオプション機能として拡大鏡表示モードが選択でき、マウスカーソルの動きとは独立に第3のマウスボタンを用いてドラッグ操作が行えるようになっている。独立のアプリケーションとして提供されている製品では、ZoomTextのようにさまざまな表示モードを選択できるものが多い。しかし、画面拡大機能においては、拡大領域が一般に狭い部分に限定されてしまうために、ページ全体の把握に関して高齢者のように認知的な機能が低下している場合には困難が生じる。特に、拡大表示領域が固定されている場合には、カーソルの指す位置と対応する拡大領域の2箇所を交互に見る不便が生じる。

3. ウェブページ要約手法

上述のような問題を解決するために、本稿ではウェブページ要約手法を提案する。本手法の目的は、ウェブページ閲覧に際してユーザが大きなフォントサイズや広い行間を要求したときに、もとのページレイアウトをできる限り損なわずにそのページの内容を提示することである。レイアウトを保存することにより、一覧性を維持してユーザのウェブページに対する理解を容易にし、かつページ制作者側の意図も尊重することができる。

レイアウトを保存してもとのページ面積を維持しつつ表示を拡大するため、もとのコンテンツすべてを一度に表示することはできない。そこで重要な情報が残るようにダイジェストを作成し、ページの詳細な内容にはダイジェストを一種の目次のように扱うことによってアクセスすることになる。

3.1 処理の概要

レイアウトを保存するために「割付枠」という

概念を導入する。割付枠とは、もとのウェブページを矩形によって分割したもので、矩形をオリジナルと同じように配置し、矩形ごとに重要な情報を取捨選択する。この割付枠の配置は、ウェブページ制作者が考えている意味的な切れ目ともできるだけ一致していなければならない。図1が処理全体の概要である。

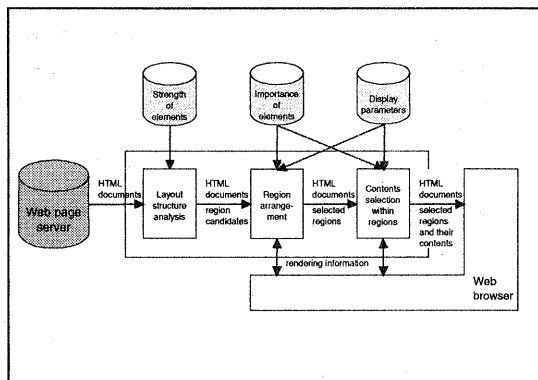


図1 処理全体図

処理は大きく分けて「レイアウト構造解析」、「割付枠設定」、「割付枠内表示内容決定」の3つのステップから成る。また、ここで用いられる情報として、「HTML タグの強さ」、「HTML タグの重要度」、「表示条件（最低限必要なフォントサイズ、行間等）」がある。

3.2 HTML タグの重み付け

HTML 文書で用いられるタグには、画面上で矩形の領域を定めるために使われるブロックレベルエレメントと呼ばれる種類と、矩形領域とは関係なくコンテンツの修飾に用いられるインラインエレメントと呼ばれる種類がある。

制作者が意図する HTML 文書の構造を推測するためにブロックレベルエレメントに対して文章の区切りの強さを表す「タグの強さ」を導入する。例えば、<H1>というタグは主に章見出し等に用いられ、段落に用いられる<P>と比較すれば区切りの強さが強い。

また、制作者が重要と考えている内容を推測するために、表示に用いられるすべての HTML タグに「タグの重要度」を導入する。例えば、何も修飾されていない文章よりはタグで修飾された文章の方が重要度が高く、さらにそれよりは<H1>で表される見出しの方が重要度が高いと考えられる。

3.3 レイアウト構造解析

レイアウト構造解析部では、ブロックレベルエレメントをノードと見なした木構造を構成する。もとのページに現れる順にブロックレベルエレメントを並べ、これをリーフノードとする。

- 最も弱い「タグの強さ」を持つノードに着目し、
- (1) その強さのノードがいくつか隣接していたらそれらの子として持つような新たなノードを作り、子と同じ強さを持つものとする。
 - (2) その強さのノードが文頭方向により強いノードと隣接していたらそれらの子として持つような新たなノードを作り、強い方のノードと同じ強さを持つものとする。

この(1),(2)の処理を「タグの強さ」の弱い順に順次繰り返すことによって、最終的には一つの連結な木を作る。この例が図2である。この木の各ノードは画面上の矩形と同一視でき、上位ノードは下位ノードを包含するような矩形を表す。ルートノードはページ全体を表すことになる。

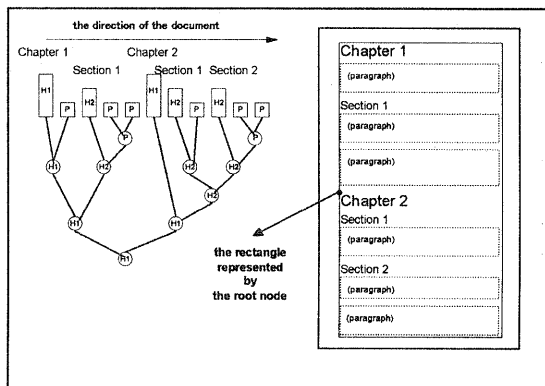


図2 レイアウト構造解析結果

3.4 割付枠配置

割付枠配置部では、前ステップで作成された木をトップダウンにたどる。これは、画面上の矩形を分割していくことに相当する。この分割を、各矩形が分割条件を満たしている範囲でできるだけ進めていく。ここで分割条件とは各矩形内で最も重要なコンテンツをユーザの求めるフォントサイズ、行間でその矩形内に表示できるということである。分割が終了した時点における各矩形が割付枠として確定する。

図3は、図2の木構造を用いた例である。木の右半分における2段階目の分割が失敗している。

この例では、木の左半分は2段階、右半分は1段階の分割が行われることになる。最終結果が図4である。3つの割付枠が設定されている。

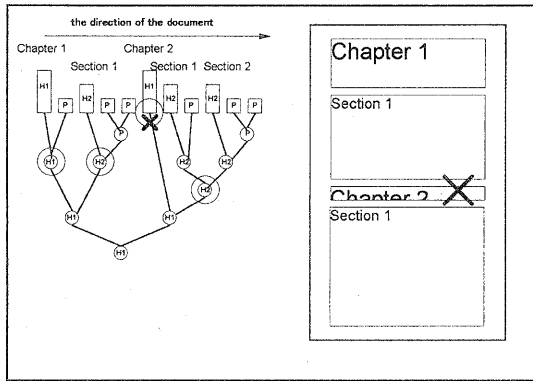


図3 割付枠配置過程

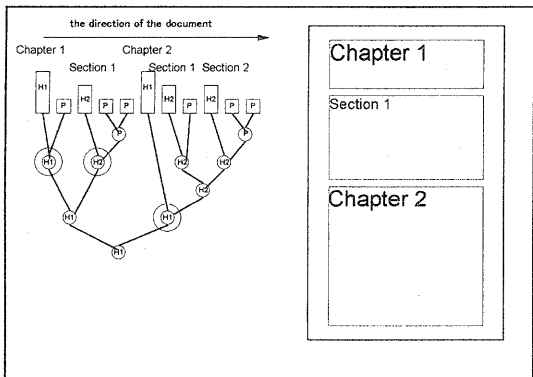


図4 割付枠配置結果

3.5 割付枠内表示内容決定

割付枠内表示内容決定部では、各割付枠ごとに「タグの重要度」の高い順にコンテンツを埋められるだけ埋めていく。表示可能な領域の大きさはもとのページと同じであるのに、個々のコンテンツはユーザの要求に応じて拡大表示されるので、重要度の低いコンテンツは表示しきれずに捨てられることになる。

3.6 詳細内容表示

ウェブページを要約すると、重要性が低いと思われる情報は失われてしまう。もとのページの内容を細部まで正確に知るには、さらに付加的

な仕組みが必要である。例えば、ダイジェストによってページの概略を把握し、興味を持ち細部を知りたいと思う部分の割付枠内をクリックするとその枠内の全内容が拡大表示されるようなシステムが考えられる。

図5から図7に本手法の実行例を示す。図5がオリジナルページで、6つの見出しとそれぞれの内容(リスト項目)がある。

図6はそれに本手法を適用して作られたダイジェストである。見出しがもとのレイアウトを保って配置されていることがわかる。見出し以下の内容は、もとの面積に応じて表示できるだけ表示されている。

図7はポップアップ画面によって詳細内容を表示する例である。ここでは、左辺中央の割付枠の内容全体が拡大表示されている。

TRL 情報 <ul style="list-style-type: none"> TRL 情報 & ガイドライン TRL カレンダー TRL の紹介 所長からのあいさつ 	リンク集 <ul style="list-style-type: none"> サーチエンジン ニュース 問い合わせ お問い合わせ お問い合わせ
IBM 情報 <ul style="list-style-type: none"> 日本IBM IBM の製品とサービス IBM Corporation 	プロジェクト <ul style="list-style-type: none"> TRL プロジェクト TRL プロジェクト TRL プロジェクト
インターネット <ul style="list-style-type: none"> インターネット接続 インターネット インターネット インターネット 	TRL の紹介 <ul style="list-style-type: none"> 所長からのあいさつ メディアにおける TRL メディアにおける TRL メディアにおける TRL

図5 実行例 (オリジナルページ)

TRL 情報 <ul style="list-style-type: none"> TRL 情報 & ガイドライン TRL カレンダー 	リンク集 <ul style="list-style-type: none"> サーチエンジン ニュース
IBM 情報 <ul style="list-style-type: none"> 日本IBM 	プロジェクト <ul style="list-style-type: none"> TRL プロジェクト
インターネット <ul style="list-style-type: none"> インターネット接続 	TRL の紹介 <ul style="list-style-type: none"> 所長からのあいさつ メディアにおける TRL

図6 実行例 (ダイジェスト)

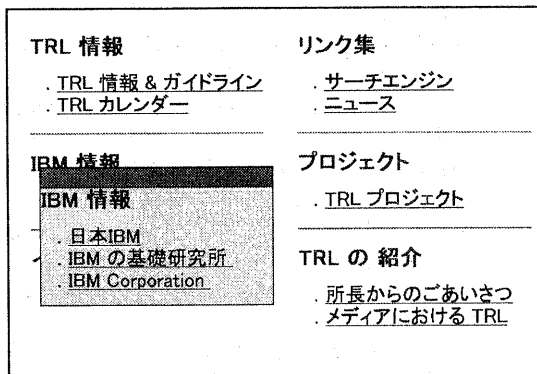


図7 実行例 (詳細内容表示)

4. 実験と主観評価

本方式の有用性を確認するために以下の手順で実験を行った。

4.1 テスト方式

シンボルによる抽象的な表現パターンに基づくページと具体的な文章 (新聞記事の見出し) によるページの 2 種類のテストセットを用意した。

(1) 抽象的ページ

1 ページを「ひらがな」、「カタカナ」、「英字」、「数字・記号」の 4 つのカテゴリに分け、2 行 2 列の計 4 つの領域で表示する。各領域にそれぞれのカテゴリに属する 36 個の文字を不規則な順に配置する。

(2) 具体的ページ

1 ページを「政治」、「経済」、「国際」、「スポーツ」の 4 つのカテゴリに分けて一列に表示する。各カテゴリには 6 つずつ記事見出しがある。すべての見出しは 1 行である。

図 8 に、実験に用いたページの例を示す。

開始	ひらがな あけふひるくれ にのきんどうあす ようちせわあじあ せりしげえだるるま	カタカナ クワキキキキキ *ズウヒツゼンツ リスエラフメセラ ノカウコソソソ
8		
次のテスト	英字 *Xg445*RE NmZMa0v*c LbtWsyD8C pOrVf*Gtu	数字・記号 ~[123*~1 050%)*~9: :7x2*...1> <1886[.6

図 8 実験に用いられたページの例

ウィンドウは 2 つのフレームから成り、左側のフレームにはテストの指示内容、右側には作業ページが表示される。指示フレームの「開始」をクリックすることにより、そのページでの計時が開始され、作業フレームから見つけ出してクリックすべき対象となる文字あるいは記事のキーワードが表示される。図 8 は、右フレームから「8」という文字を見つけ出してクリックせよ、という指示を表している。

正しくクリックが行われた場合に、次のクリック対象が表示される。1 ページについて、5 個所のクリックを行い、各クリックに要した時間が求められる。

比較対照のために、以下の 3 つの表示方式をあわせて試験を行った。

(1) 単純なフォント拡大

ウェブブラウザの機能を用いて、単純に全フォントを拡大した。拡大することによって、行の折り返しが生じ、スクロール操作が必要になった。

(2) 部分的な画面拡大表示 (拡大鏡方式)

Windows98 の部分的画面拡大機能を用いて、マウスカーソル付近の内容が固定拡大領域に表示されるように設定した。

(3) ダイジェスト

本稿で提案する手法によりダイジェストを作成した。その結果、抽象的ページ、具体的ページともカテゴリごとに一つの割付枠が設定された。ダイジェスト画面において行の折り返しが起こらないように、各行からはみ出す部分は表示しないようにした。図 7 と同様に、割付枠内をクリックすれば、そのカテゴリの詳細がポップアップ画面によって表示される。

抽象的ページ、具体的ページのそれぞれに 4 種類のパターンを用い、3 つの方式との組み合わせによって計 12 通りのテストを行った。

被験者は 3 人 (70 歳女性、65 歳女性、75 歳男性) である。3 人とも、コンピュータの操作経験はほとんどない。

時間間隔データの取得のほかに、以下の項目に関して主観評価も行った。

- 操作が理解しやすいか
- 操作がわずらわしくないか
- 操作が疲れにくい
- 目標はみつかりやすいか
- 画面は読みやすいか

4.2 結果

抽象的ページ、具体的ページの時間測定で得られた結果をそれぞれ表 1と表 2に示した。被験者ごとの行において、上の行は各ページ（5 回クリック）完了までの所要時間（ミリ秒）の平均を、下の行は、本方式（ダイジェスト）による時間を 1としたときの比率を示す。

表1 抽象的ページの測定結果

	所要時間（ミリ秒）		
	時間比		
被験者	フォント拡大	拡大鏡	ダイジェスト
A (女 70)	82518	140648	63810
	1.29	2.20	1.00
B (女 65)	73793	116248	91203
	0.81	1.27	1.00
C (男 75)	135470	164183	112033
	1.21	1.47	1.00

表2 具体的ページの測定結果

	所要時間（ミリ秒）		
	時間比		
被験者	フォント拡大	拡大鏡	ダイジェスト
A (女 70)	94483	109798	88235
	1.07	1.24	1.00
B (女 65)	65308	70015	60570
	1.08	1.16	1.00
C (男 75)	181500	164183	112033
	1.46	1.58	1.00

また、被験者による主観評価の結果を表 3に示す。各数字は、5 段階評価で 1 が否定的、5 が肯定的としたときの 3 人の被験者による回答の平均を示している。

表3 主観評価結果

質問事項	フォント拡大	拡大鏡	ダイジェスト
操作が理解しやすいか	3.3	2.0	4.0
操作がわずらわしくないか	2.7	2.0	3.7
操作が疲れにくい	3.0	2.0	3.7
目標はみつけやすいか	3.7	2.0	3.7
画面は読みやすいか	3.3	2.0	4.0

その他に、実験中の観察から以下のような知見が得られた。

スクロールバーの操作について、実験の期間中では完全に把握しきれず、操作をためらうことがたびたびあった。また、スクロールの

位置決め操作に時間を取られていた。

- 部分的画面拡大方式については、もとの画面と拡大領域との対応を見失ってしまうという感想や、拡大領域内ではカーソルの移動距離も大きくなってしまいうので、位置決めがより困難になるという感想が得られた。
- ダイジェストに関しては、ポップアップを表示させるためにどこをクリックすればよいかで迷うことがあった。

5. おわりに

ダイジェストは他の方式に比べほとんどの場合で一番所要時間が短い。主観評価や実験の観察中の知見も含め、以下のように考えることができる。(1)部分的拡大方式は全方式の中で最悪の結果であったが、これは常に一部しか拡大表示されず、一覧性がないことに起因すると思われる。(2)高齢者にとってはフォント拡大に伴うスクロールの発生が、内容理解、操作の両面で重大な困難をもたらす。これらを解決するダイジェストのメリットが、詳細内容表示を別に必要とするデメリットを上回ったと考えられる。今後の発展方向として、本方式を音声合成と統合することが挙げられる。音声による出力でランダムアクセスできないことの欠点を、この要約方式で補うことができると思われる。また、本稿で提案した範囲では、HTML タグのみを手がかりとして、ページの意味的な内容にまで踏み込んではいなかった。これに自然言語処理による要約の手法を組み合わせ、質の向上を図ることも可能であると考えられる。

参考文献

- [1] World Wide Web Consortium, Web Accessibility Initiative, <http://www.w3.org/WAI/>
- [2] Morris, J. Morgan, "User Interface Design for Older Adults," *Interacting with Computers*, Vol. 6, No. 4 (1994)
- [3] "Older Adults and the World Wide Web," SPRY Foundation (1999)
- [4] Microsoft Windows 98 magnifier, <http://microsoft.com/enable/training/windows98/magnifier.htm>
- [5] ThinkPad TrackPoint magnifier, <http://www.pc.ibm.com/us/thinkpad/>
- [6] ZoomText Xtra, <http://www.zoomtext.com/>
- [7] R. Morrell and K. Echt, "Instructional Design for Older Computer Users: The Influence of Cognitive Factors," in "Aging and Skilled Performance," edited by W. A. Rogers, Lawrence Erlbaum Assoc., Inc. (1996)