

Web ブラウジング支援システム Zero-Click

2J-6

南野 朋之† 齋藤 豪†† 奥村 学††

† 東京工業大学大学院 総合理工学研究所 †† 東京工業大学 精密工学研究所

1 はじめに

World Wide Web の普及は、世界中に分散された文書をハイパーリンクで結びつけ、情報の共有を実現した。そして、情報交換が活発に行われることによって、Internet 上に膨大な情報が存在するようになった。しかし、情報が膨大になるにつれ、ユーザが求める情報を探し出すのは困難となってきている。現状では、検索エンジンなどの検索サービスは存在するが、それぞれの情報の価値や内容に対する評価が十分なされていないと言えない。

よって、自分の求める情報の存在するページを探すためには、実際にリンクされているページを訪れ、内容を評価しなければならない。そのため、ユーザの作業として以下の様な作業を想定することができる。

1. Web ページ上のリンクのうち一つを選び、そのリンクをクリックしてリンク先の Web ページを訪れる
2. 訪れた Web ページを吟味する
3. もし、自分の求める情報が含まれなければ、ブラウザの「戻る」ボタンをクリックし、前のページに戻る

以上のような操作を減らすことにより、情報閲覧の効率を上げる Web ブラウジング支援システムが求められている。

解決手段の一つとして、検索エンジンの高性能化 [1] があげられるが、このような問題は検索時のみに起きる問題ではない。また、ユーザの履歴などを用いて、ユーザがたどるべきリンクを推奨するシステム [2] も研究されているが、その推奨がユーザの期待したもので無かった場合、ユーザの望んでいるページにたどり着けないという問題がある。

そこで、本研究ではブラウザに表示されているページにハイパーリンクが存在する場合に、リンク先のページにどのような内容が含まれているか、ユーザが簡単に参照することができるシステムの構築を目指す。

本研究ではプロキシサーバを用いてユーザが Web ページ上のリンクの上にマウスカーソルを置くと、同じウィンドウ上にポップアップウィンドウが表示され、そこにリンク先の情報が表示されるようにすることで、クリック操作なしでリンク先の情報を得ることのできるブラウジング支援システムの構築を提案する。また、誰もが容易に導入、利用する事のできるように、プロキシサーバ形式でシステムの実装を行う。

2 ブラウジング支援システム

2.1 システムのインターフェース

リンク先の情報をユーザに示す方法は、図 1 に示したようなポップアップを用いる。図 1 では、ベースペー

Zero-Click - a system to support Web-browsing

† Tomoyuki NANNO (nanno@r.pi.titech.ac.jp)

†† Suguru SAITO (suguru@pi.titech.ac.jp)

†† Manabu OKUMURA (oku@pi.titech.ac.jp)

Interdisciplinary Graduate School of Science and Engineering,
Tokyo Institute of Technology(†)

Precision and Intelligence Laboratory, Tokyo Institute of
Technology(††)

ジ上のハイパーリンク「ニュース」上にマウスポインタが置かれ、ポップアップウィンドウ内にリンク先のページが表示されている。ポップアップの利点はクリック操作すること無く、次ページの情報を同じウィンドウ内で見ることが出来る事である。

よって、「戻る」ボタンのクリックやウィンドウを閉じるなどのブラウザの操作が減り、情報閲覧操作の効率の向上が期待できるとともに、非常に簡単に操作できるという点で、アクセシビリティの向上にもつながると考える。

本システムは、リンクをクリックして次ページに飛ぶような、今まで可能だった操作が失われない。また見た目も変わらない、という点がこのインターフェースの利点である。

ユーザの操作は次の通りである。

1. ユーザはブラウザに対してプロキシサーバを使う設定をする
2. Web ページにアクセスすると、見た目にはプロキシを経由しない場合と何も変わらない Web ページが表示される
3. 表示された Web ページ中に存在するリンクの上にマウスが置かれると、ポップアップウィンドウが開き、その中にリンク先の情報が示される
4. マウスをリンクの上からはずすと、ポップアップウィンドウが閉じる

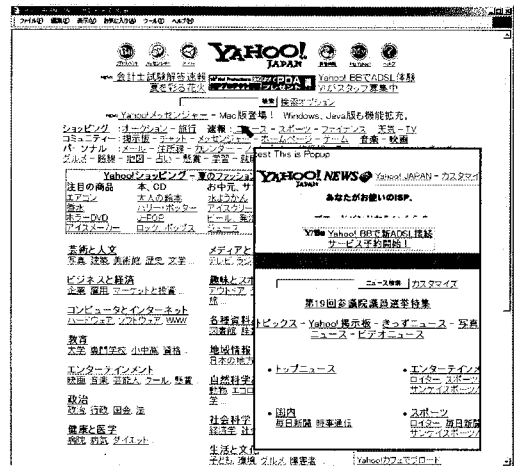


図 1: システムの表示例 (太枠内がポップアップ)

2.2 システム構成

図 2 は本研究で構築したプロキシサーバのシステム図である。図で、黒の矢印で表示されている経路はベースページを処理する場合の経路で、グレーの矢印で示される経路はポップアップページを処理する場合の経路である。

このシステムでは、プロキシサーバを通過する HTML 文書に加工を行うことによりポップアップ機能

を実現している。しかし、通過する全ての HTML 文書に対して同じ加工を行うわけではない。よって、クライアントからの要求が、ポップアップページとベースページのどちらからの要求であるか区別する必要がある。

本システムでは、ポップアップとして表示すべき HTML 文書の URL に、プロキシサーバの IP アドレスを自動的に付加することで区別を行っている。

以下にプロキシサーバでの基本的な処理を示す。

ベースページの場合

1. ユーザがアドレス欄に URL を指定するか、もしくはリンクがクリックされると、クライアントからプロキシサーバに対して要求が出される。
2. その場合、接続先 URL にプロキシサーバのアドレスが含まれていないので、ベースページと判断する。
3. Web サーバから、要求のあったデータを取得する。
4. 取得したデータが HTML 文書であれば、Javascript の付加と、マウスが乗った事を判断するためのリンクの書き換えを行う。
5. クライアントに送信する。

ポップアップページの場合

1. リンクの上にマウスポインタが置かれると、リンク先の URL にプロキシサーバの IP が付加された URL に対して接続要求が出される。
2. プロキシサーバでは、接続先 URL にプロキシサーバの IP が含まれているため、ポップアップページであると判断する。
3. 接続先 URL からプロキシサーバの IP を取り除き、要求のあったデータを取得する。
4. ポップアップに表示されるデータは、以下の方針で加工を行う。

- 取得したデータが HTML 文書の場合は、ブラウザから見た際、画像などの URL 指定が相対指定のままであると、ベースページとして表示したときと URL が異なり、ブラウザのキャッシュが使用できないため、相対指定の URL を絶対指定に書き換える。
- 自動的に他のページにジャンプするページに対しては、ジャンプ先に本来のリンク先情報があると考えられるため、ジャンプ先のページを再帰的に取得する。
- 取得したデータがブラウザで表示できないデータ (ps ファイル等) であった場合、リンク上にマウスが乗っただけで、ダウンロードが始まってしまうため、表示できない旨を示す HTML 文書に差し替える。

5. クライアントに送信する。

3 おわりに

本研究では、プロキシサーバを用い既存の HTML 文書を書き換えることにより、リンク先の情報をポップアップによりユーザに提示する Web ブラウジング支援システムを提案し、実装を行った。

本研究で実装したシステムはプロキシサーバを用いているので、新たにクライアントにソフトウェアを導入しない形での実装が可能である。また、リンクをクリックして次ページに飛ぶような今まで可能だった操作方法も残されており、ページの表示にも変化がないため、Web ページ制作者の意図を害すこともユーザ

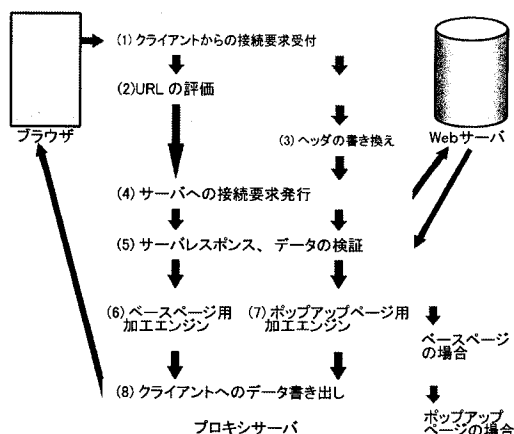


図 2: システム構成

の利用に関する抵抗も無く使用することができると考えられる。

このシステムは、ベースページを作る際に、リンク先全てを先読みする必要が無く、リンク上にマウスが乗った時点で必要なもののみサーバから取得するため、このシステムを使用しなかった場合と比べて、トラフィックは付加した Javascript の分しか増加しない。したがって、ページの表示もこれまで同様高速に行われる。

また、すでに Javascript が記述されているページに対しても、その機能を失うことなく、共存することが出来るように、実装されている。

本研究で実装したシステムを用いたとき、特に情報閲覧の効率が向上すると考えられるケースとして、以下のものが考えられる。

- 画像がサムネール形式で一覧されている Web ページにおいて、拡大表示を参照するようなケース
- 検索エンジンの出力結果やリンク集、もしくはニュースのタイトルのように、多数のリンクが存在し、その中から自分の意図にあったページを見つけようとするケース
- 文章中にリンクが存在し、リンク先の情報を参照しながら本文を閲覧するようなケース

今後の課題としては、ブラウザや OS の違いによる挙動の違いへの対応を厳密に行うことが挙げられる。Internet Explorer と Netscape Navigator では Javascript の仕様が全く異なることや、同じ種類のブラウザであっても OS やバージョンが変わることで動作が変わるため、クライアントの環境により使用する Javascript が変わるような実装が必要になると考えられる。

参考文献

- [1] 石濱雄志, 伊藤吾一, 阿部倫之, 服部進実, “ユーザ・アダプティブ協調 WWW 検索システム”, 情報処理学会第 62 回全国大会, 8W-06, 2001.
- [2] 中嶋敏行, 水原徳洋, 太田 学, 石川 博 “Web アクセスログを利用したユーザモデルに基づく推薦システム”, 情報処理学会第 62 回全国大会, 4W-04, 2001.