

Webサイト 評価システムの構築

00H077 鳥阪武志

目次

1	はじめに	1
2	Web評価システムの概要	2
2.1	Webユーザビリティとは	2
2.2	Webアクセシビリティとは	3
2.3	CMSとは	4
2.4	実装	6
2.5	実行結果	7
3	考察	9
3.1	本システムの利点	9
3.2	本システムの問題点	9
4	まとめ	10
5	付録	11

1 はじめに

Webが登場した当初では、Webは研究者の間でのみ使われていたが、現在では多様なユーザに使われるようになり、インターネットが普及していった。これに伴い、Webは個人の情報発信やビジネスに活用される機会が増えていった。これによってWebサイト、Webページの数膨大な物になった。そして、今までになかったWebの使いやすさいわゆるユーザビリティや、多様な閲覧環境で閲覧可能かというアクセシビリティが重視されるようになってきた。

ユーザビリティやアクセシビリティの高いサイトの構築、管理を行うには人的コストがかかるため、解決法の一つとしてBlogやtDiary¹などのCMS(コンテンツマネジメントシステム)が登場した。しかし、CMS導入にはWebサーバ構築などの導入コストがかかることや、CMSに登録されたスタイルに固定化されてしまうなど、改善の余地がある。また、既存のサイトをCMS上に移行させるには多くの手間がかかることも問題である。

そこで既存のサイトのユーザビリティやアクセシビリティを評価し、問題点を指摘するツールの必要性が出てきた。

今回の研究では、指定されたWebサイトがどの程度多様な閲覧環境に対応しているかを評価する前段階として、Webの文法の正しさを点数化するツールを用いて、サイトを総合的に評価するシステムを構築した。

¹<http://www.tdiary.org/>

2 Web評価システムの概要

2.1 Webユーザビリティとは

Webユーザビリティ（Usability）とは、Webの使い勝手、使いやすさ、利用品質のことを意味する。現在、多くのWebサイトが存在するが、ユーザビリティに配慮して制作されたサイトは、まだまだ少ないのが現状である。

ユーザビリティの良いページは具体的にどう作ればいいのか様々な議論がある。また、ユーザビリティをよくすることは簡単ではない。ユーザビリティは様々な要素が合わさった産物の結果だからである。関連する分野は情報工学、人間工学、デザイン学、社会学、心理学、言語学などの幅広い分野の知識が必要になる。

ユーザビリティを向上させる配慮としては、ダウンロード時間の予測のための情報提供、リンク先への適切なナビゲーション、デザインの統一、サイト内の現在の表示位置の明示などが挙げられる。

1. インタフェース情報

機械と人間の接点部分である。ユーザーインタフェース、マン・マシン・インタフェース、マシン・マン・インタフェースなど呼び方も様々である。接面とも呼ばれている。コンピュータの場合、マウスやキーボードが接面です。画面上のアイコンやボタンリンクテキストもインタフェースの一部である。マウスやキーボードは第一接面、アイコンやボタンは第二接面と呼ばれる。Webユーザビリティを考えていくとき、「1つはインタフェース部分をどうするか」という問題がある。

2. 情報の構造化

情報を整理し情報をうまく伝えることもWebユーザビリティにおいて必要である。これはインフォメーションアーキテクトとよばれる。Webサイトのどの部分を現在閲覧しているかを示すこともインフォメーションアーキテクトのひとつである。

2.2 Web アクセシビリティとは

Web アクセシビリティ (Accessibility) とは、ハードウェア、ソフトウェア、サービスなどの情報技術を、複数の支援技術 (アシティブ・テクノロジー) を併用、または単独で使用するによって、ユーザの閲覧環境を問わず、Web を利用しての情報の送受信ができること、あるいは、その使いやすさを意味する。

たとえば視覚異常の人などにも正確に内容を伝えられるように、画像などに代換情報を付加することや文字を大きく表示させること、マウスを使わずに Web サイトを閲覧できることなどを指す。これは W3C² で標準化が進められている。特に、2001 年 6 月にアメリカ合衆国でリハビリテーション法第 508 条「電子・情報技術アクセシビリティ基準」が実施されてからは、Web 製作者の間でもアクセシビリティに対する関心が高まってきた。

これには、解像度による影響が少ないデザインや、一般的 (標準的) な技術の利用、フレームを使用しないことなどが求められる。

また、日本でも経済産業省が Web アクセシビリティガイドラインの JIS 規格化に取り組んでおり、官公庁などのサイトのアクセシビリティも見直しがかけられつつある。

²<http://www.w3.org/WAI/>

2.3 CMS とは

コンテンツ マネージメント システムは、プロダクトやドキュメント、Web 上の各コンテンツなど、あらゆるコンテンツを手軽に管理できるシステムのことである。

HTML などの技術的な知識がなくても、簡単に Web コンテンツを作成できるのが特徴である。

また、あらかじめ用意されたテンプレートを使用することにより、Web サイトのデザインを統一する事ができる。このテンプレートを変更することによりサイト全体のデザインを変更することも可能である。さらに、プラグインを用いることにより、機能を拡張することもできる。

しかし、CMS 導入には Web サーバ構築などの人的コストがかかることや、CMS に登録されたスタイルに固定化されてしまうなど、改善の余地がある。また、既存のサイトを CMS 上に移行させるには手間がかかることも問題である。

CMS の主なものとしては、Weblog(blog)があり、日記の機能に特化した tDiary、議論・共同作業に特化した Wiki などが挙げられる。

1. Weblog(blog)
2. tDiary
3. Wiki

文字ベースの画面に、コマンドと呼ばれる文字列を打ち込むことで動作するインターフェース環境で動作するシステムを構築した。

コーディングに関しては既存の Web の文法評価ツールとシェルスクリプトを組合せることで、開発の労力を軽減させると共に、他の評価ツールと連携がとれるよう、汎用性を持たせることに注力した。

完成した Web 評価システムの動作手順は以下の通りである。

1. 対象サイトをローカルに保存
2. ローカルに保存したサイトのサイトマップを作成
3. 作成したサイトマップを元にサイト構成を取得
4. サイト構成から個々のページを評価・記録
5. サイト評価結果の表示
6. サイト平均点の算出

対象サイトをローカルに保存することで、ネットワークの負荷を軽減すると共に、動作時間も節約することができる。

2.4 実装

システムの構築には GNU/Linux をインストールしたマシンを用いた。汎用性を重視して開発したため、Perl と一部の GNU によって配布されているツールが動作する環境があれば、OS に依存せずに動作させることができる。

本システムの最小構成は Another HTML-lint と各種シェルスクリプトの組み合わせである。

主な動作は以下のとおりである。

1. 対象サイトをローカルに保存

wget³ コマンドを用いローカルにサイトをコピーする。

コピーすることにより、HTML の文法チェッカーやその他のツールを使用するように拡張した場合でも、実際にサイトを参照するのは1度のみなのでネットワークにかかる負荷を軽減することができる。

2. 対象サイトのサイトマップを作成

tree.pl⁴ でローカルに保存したサイトのサイトマップを作成する。

引数として与えられた web ページのリンクをたどり XHTML 形式でサイトマップを作成する。

階層ディレクトリに保存されたサイトを評価するために用いている。

3. 作成したサイトマップを元にサイト構成を取得

サイトマップをローカルにある html ファイルの相対位置を取り出す。

tree.pl の出力結果には、本研究では必要のない情報も含まれるため、それらの中からファイルの相対位置のみを取り出す。

本研究では C 言語で getsite コマンドを作成し、サイト構成を取得した。

4. サイト構成から個々のページを評価し、記録する。

取得したサイト構成を1ファイルごとに Another HTML-lint⁵ を用いて点数をファイル化する。

デフォルトではページの減点対象部分を明示するため、点数が含まれる、最後の行のみを取り出し、そこから数値のみを取り出し、記録した。

本研究では不要な文字列を取り除くために、C 言語で getnum コマンドを作成し、点数のみを取得し、記録した。

5. サイト評価結果の表示

サイト構成と、点数を記録したファイルを paste コマンドを用いて1つのファイルにした。これにより、各ページの点数を確認することができる。

6. サイト平均点の算出

サイト全体の評価を出すために、平均点を算出した。

これには、awk スクリプト average.awk を作成し、平均を求めた。

以上のように実装を行ない、実際にシステムとして稼働することを確認した。

³主に UNIX 上で使えるダウンロード支援ツールであり、ホームページ自動巡回ツールでもある。

⁴Daniel Naber 氏が作成した perl プログラム

⁵石野 恵一郎氏が作成した HTML 文書の文法チェッカーである。Perl で書かれており、Unix 系 OS、MacOS、Windows 系 OS で使用できる。

2.5 実行結果

あるサイトに対して実行した動作結果は以下の通りである。図1ではファイルが階層化されていないサイトを、図2ではファイルが階層化されているサイトをそれぞれ評価している。

```
[tori@h-ps011 kenkyu]$ points.sh www2.odn.ne.jp/ikedatyuuou/  
サイトをローカルに保存しています...  
点数化をしています...  
http://www2.odn.ne.jp/ikedatyuuou/  
index.html -310  
ankeat.htm 0  
index2.htm -128  
map.htm -5  
newpage12.htm 59  
newpage14.htm -26  
rinku.htm -102  
sinema1.htm -17  
sinema2.htm -19  
sinema3.htm -14  
平均は -56.2 点です.
```

図 1: 階層化されていないサイトへの評価システム実行結果

```
[tori@h-ps011 kenkyu]$ ./points.sh www.109cinemas.net/minoh/  
サイトをローカルに保存しています...  
点数化をしています...  
http://www.109cinemas.net/minoh/  
index.html 18  
access.html 29  
establishment.html 56  
prices.html 8  
recruit.html 35  
schedule/index.html 29  
schedule/0131.html 94  
schedule/0207.html 94  
schedule/0208.html 94  
schedule/0214.html 94  
平均は 55.1 点です.
```

図 2: 階層化されているサイトへの評価システムの実行結果

3 考察

Web 評価システムを構築し、実際に動作させてみた。
本システムについて以下のような利点と問題点が挙げられる。

3.1 本システムの利点

サイトを評価する場合、ローカルに保存したファイルを点数化する事により、実行時間がかからない。

複数のツールを併用するよう拡張した場合でも、実際に対象サイトを参照するのは1度ですむ。

CUI で動作するので、他のツールやコマンドとの連携が取りやすい。

3.2 本システムの問題点

シェルアカウントがなければ利用できない。

今研究で構築した状態では web 上から利用できない。

サイトの平均点を算出するため、極端に偏差値の大きいページが存在した場合、平均点が大きく変わり、参考にならなくなってしまうことがある。

cgi や DHTML などを用いた動的なページなどには対応できていない。

4 まとめ

本研究では、Another HTML-lint とシェルスクリプトを組み合わせてサイト評価システムを構築した。

本システムではサイトの構造と、点数を記録したファイルを別のファイルに保存することによって拡張性を高めている。

このサイト構造を記録したファイルを他のツールで用いることで、アクセシビリティや、ユーザビリティなど、別の角度からサイトを評価することも可能である。

今後、このシステムを利用して Web 上のコンテンツの評価を行ない、Web サイトの運用上の問題点を明確にし、改良を加えていく一助になれば幸いである。

謝辞

本研究を進めていく上で、御指導及び御協力を頂いた大垣 齊講師・藤井 信夫教授・中村 孝講師・fken.a4w メールリストのメンバーの方々、そして情報システム工学専攻の大学院生である赤松 康介氏、中路 貴博氏に深く感謝の意を表します。

参考文献

- [1] Another HTML-lint, [<http://openlab.ring.gr.jp/k16/htmlhint/>]
- [2] tree.pl sitemap script, [<http://www.danielnaber.de/tree/>]
- [3] tDiary, [<http://www.tdiary.org/>]
- [4] W3C, [<http://www.w3.org/WAI/>]
- [5] ウェブ・ユーザビリティ&アクセシビリティ・ガイドライン, 石田優子, 毎日コミュニケーションズ, [ISBN 4-8399-1025-1]
- [6] ウェブ・アクセシビリティ, Michel G.Paciello, ASCII, [ISBN 4-7561-4099-8]
- [7] Wiki Way, Bo Leuf&Ward Cunningham, SOFTBANK, [ISBN 4-7973-1832-5]

※ URL はすべて 2004 年 2 月 6 日時点のもの

5 付録

Webサイト評価スクリプト

```
points.sh
```

```
#!/bin/sh
echo "サイトをローカルに保存しています..."
wget -r1=5 -np -L http://$1 2> /dev/null
cd $1

echo "点数化をしています..."
# サイトマップを作成し、URIから.htmを含むものを取り出す
~/kenkyu/tree/tree.pl ~/kenkyu/$1 | grep "href" | awk '{print $2}' | \
~/kenkyu/getsite | grep ".htm" > ~/kenkyu/site.txt 2> /dev/null

# 取り出したファイルパスをhtmlintに渡す
cat ~/kenkyu/site.txt | awk '{print "perl ~/htmlint/htmlint " $1 " | \
tail -1 >> ~/kenkyu/tmp.txt" }' | sh

cat ~/kenkyu/tmp.txt | awk '{print $2}' >> ~/kenkyu/tmp2.txt
rm ~/kenkyu/tmp.txt
echo "http://"$1
# 点数を取り出す
cat ~/kenkyu/tmp2.txt | ~/kenkyu/getnum > ~/kenkyu/tensu.txt

# URIと点数を関連付ける
paste ~/kenkyu/site.txt ~/kenkyu/tensu.txt > tmp3.txt
cat tmp3.txt | awk -f ~/kenkyu/average.awk > ~/kenkyu/site_points.txt
cat ~/kenkyu/site_points.txt

# 後処理
rm ~/kenkyu/site.txt
rm ~/kenkyu/tensu.txt
rm ~/kenkyu/tmp2.txt
rm tmp3.txt
rm -r ~/kenkyu/$1
```

```

getsite.c

#include <stdio.h>

/*
 * 入力された文字列からファイル名のみを取り出すプログラム
 */

int main(void)
{
int ch,flag=0;
while((ch=getchar())!=EOF)
{
    /*
     * はじめて「"」が出現した場合、フラグを立てる
     */
    if(ch=='"' && flag==0){
        putchar(ch);
    }
    /*
     * 2つめの「"」の出現でフラグをリセットし、改行する
     */
    else if(ch=='"' && flag==1){
        flag=0;
        putchar('\n');
    }else{;}
}
return 0;
}

```

getnum.c

```
#include <stdio.h>
#include <ctype.h>

/*
 * 入力された数値から始まる文字列から数値のみを取り出すプログラム
 */
int ch;
while((ch=getchar())!=EOF)
{
    if(isdigit(ch)){
        putchar(ch);
    }if(ch=='\n'){
        putchar('\n');
    }if(ch=='-'){
        putchar(ch);
    }
}
return 0;
}
```

average.awk

```
BEGIN {total=0}
total+=$2
END {print "平均は " total/NR " 点です."}
```