

JavaScript を使用した HTML のアクセシビリティ評価支援

大原圭木[†] 岩田一[‡] 白銀純子^{*} 深澤良彰[†]
 早稲田大学[†] 神奈川工科大学[‡] 東京女子大学^{*}

1. はじめに

近年、Web 上で RIA(Rich Internet Application)と呼ばれるコンテンツを載せるページが増えている。JavaScript や Flash などを使用した操作性や表現性高い動的なコンテンツである。RIA を使用すれば、Web 上の文字や画像、ページに様々な動きを付加することができる。これにより、表現の可能性が大きく向上し、多機能な RIA を開発することが可能となる。

しかし、RIA を使用したページにはアクセシビリティの問題が生じる可能性がある。ページが動的に更新されるので、スクリーンリーダーや拡大鏡などの支援技術を使用してページを閲覧する人は更新されたことに気付かず、正しい情報が得られない恐れがあり、深刻な問題である[1]。

こういった問題を解決するために、W3C は WAI-ARIA[2]という、RIA のアクセシビリティに関するガイドラインを策定している。WAI-ARIA は、動的な Web コンテンツを障害者ユーザにとってよりアクセシブルにする方法を定義するものである。特に Ajax, HTML, JavaScript などの技術を用いて開発される動的なコンテンツを対象としている。

しかし、このガイドラインは膨大な量があり、作成したコンテンツがガイドラインに対応しているのかを確認するには大きな手間とコストがかかる。そこで本研究では、作成した HTML がガイドラインに対応しているかどうかを自動評価し、テストや修正を容易にする手法について提案する。

2. 研究の利点

本研究では、作成した HTML がアクセシビリティのガイドラインに対応しているかを自動で評価する手法について提案する。

JavaScript を使用した動的なページは、表示が次々と変化していくので、アクセシビリティ

のチェックが難しい。そのため、JavaScript を使用したページのアクセシビリティを高めることが急務である。

アクセシビリティのガイドラインの量は膨大である。作成したコンテンツがガイドラインに沿うかを自動で評価することにより、手間とコストを削減することができる。

結果としてアクセシブルな RIA を提供することが可能となる。

3. ガイドラインの例

WAI-ARIA の例を挙げる。

図 1 は、図 2 のチェックボックスを表示するソースコードである。図 1 のソースコードにガイドライン項目を適用させたものが図 3 である。

チェックボックスの例では、1 行目の「input type="checkbox"」の記述からこのタグによってチェックボックスが作成されていることが分かる。この要素がチェックボックスを表していることをユーザエージェントに知らせるために「role="checkbox"」と記述しなければならない。また、チェックボックスに対しては「aria-checked」属性を指定しなければならない。「aria-checked」属性を指定することによって、そのチェックボックスが現在チェックされているかをユーザに知らせることができる。

また、図 4 は図 2 のツリー構造を示すソースコードである。図 5 は図 4 のソースコードにガイドライン項目を適用した例である。HTML にはツリー構造の展開や折りたたみをサポートするネイティブの要素がない。そこでツリーの要素には treeitem のロール、その親要素には tree のロールを設定しなければならない。これらのロールを設定することにより HTML のツリー構造を支援技術に伝えることができる。

```
1 <input type="checkbox">
2 チェックボックスのラベル
```

図 1 チェックボックスのソースコード

Support for the evaluation of the accessibility on HTML using JavaScript

[†] Tamaki Ohara • Waseda University

[‡] Hajime Iwata • Kanagawa Institute of Technology

^{*} Junko Shirogane • Tokyo Woman's Christian University

[†] Yoshiaki Fukazawa • Waseda University



図2 ブラウザでの表示画面

```

1 <input type="checkbox"
2   value="check"
3   role="checkbox"
4   aria-checked="true">
5   チェックボックスのラベル

```

図3 ガイドライン項目の適用例1

```

1 <ul>
2   <li class="section">
3     <h2>果物</h2>
4     <ul>
5       <li>りんご</li>
6       <li>みかん</li>
7     </ul>
8   </li>
9   <li class="section">
10    <h2>野菜</h2>
11    <ul>
12      <li>トマト</li>
13      <li>キャベツ</li>
14    </ul>
15  </li>
16 </ul>

```

図4 ツリー構造のソースコード

```

1 <ul role="tree">
2   <li class="section" role="treeitem">
3     <h2>果物</h2>
4     <ul role="group">
5       <li role="treeitem">りんご</li>
6       <li role="treeitem">みかん</li>
7     </ul>
8   </li>
9   <li class="section" role="treeitem">
10    <h2>野菜</h2>
11    <ul role="group">
12      <li role="treeitem">トマト</li>
13      <li role="treeitem">キャベツ</li>
14    </ul>
15  </li>
16 </ul>

```

図5 ガイドラインの適用例2

4. 本手法の手順

本手法の概略図は図6のようになる。本手法は、HTMLの読み込みからガイドラインの適用の判定結果の出力まで、大きく以下の3つのステップで構成される

1. 作成されたHTMLの読み込み
2. ガイドラインの適用の判定
3. 判定結果の出力

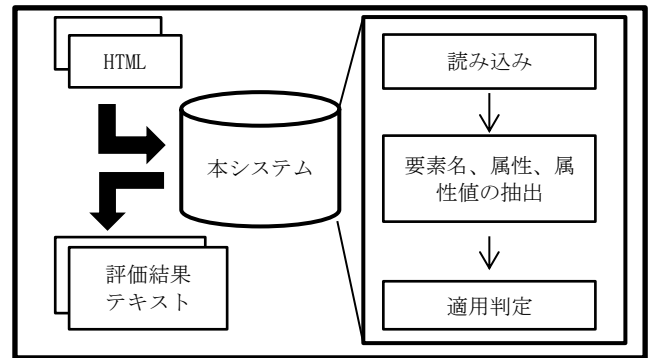


図6 本システムの概略図

4.1. HTMLの読み込み

作成されたHTMLを読み込み、要素名、属性、属性の状態を得る。図1では、チェックボックスというオブジェクトについて記述するタグの中に書かれた属性を得る。

4.2. ガイドラインの適用の判定

次に解析した情報をガイドラインと照らし合わせ、ガイドラインを適用できるかどうかを判断する。要素名や属性からその要素が何を表しているかを判断し、それに対してどのガイドライン項目を適用させればよいかを判定する。

4.3. 判定結果の出力

ガイドラインの適用の判定結果について、以下の3つのレベルに分けて出力する。

- ・必ず直さなければならないもの
- ・直した方がよいもの
- ・どちらでもよいもの

ガイドラインの各項目は、必須で適用すべきものからオプションで適用を判断してもよいものまで適用に関する強制度は様々である。3つのレベルはこの強制度を表していて、WAI-ARIAに記述されている「なければならない」「てはならない」「すること」「しなこと」「べきである」「推奨される」「てもよい」「オプション」というキーワードから判断する。

5. おわりに

本手法では、JavaScriptを使用した動的なWebページのHTMLファイルを入力し、ガイドラインの適応状況を自動で判定する手法を提案した。今後の課題はシステムの実装と評価である。

・参考文献

- [2] IBM AJAXのアクセシビリティについて
<http://www-06.ibm.com/jp/accessibility/news/ajaxaccessibility.html>
- [1] Accessible Rich Internet Applications (WAI-ARIA) 1.0
<http://www.w3.org/TR/wai-aria/>