Considering Web Accessibility in Information Retrieval Systems

Myriam Arrue and Markel Vigo

University of the Basque Country, Informatika Fakultatea, Manuel Lardizabal 1, E-20018, Donostia, Spain
{myriam,markel}@si.ehu.es

Abstract. Search engines are the most common gateway for information searching in the WWW. Since Information Retrieval systems do not take web accessibility into account, results displayed are not useful for users with disabilities. We present a framework that includes the requirements to overcome this situation. It is composed of three modules: Content Analysis Module, Accessibility Analysis Module and Results Collector Module. This framework facilitates the implementation of search engines which return results ranked according to accessibility level as well as content relevance. Since criteria to sort results by their accessibility are necessary, we define accurate quantitative accessibility metrics which can be automatically calculated exploiting results yielded by any automatic evaluation tool. A prototype based on these requirements has been implemented to show the validity of the proposal.

1 Introduction

The WWW has a great potential to make life easier for disabled people and make them less dependant on their relatives or friends since users can perform tasks they hardly could accomplish by themselves (i.e.: do shopping, buy tickets, etc.). However, as most websites are not accessible, these users come up against design barriers which do not let them access the information. In order to tackle this situation Web Content Accessibility Guidelines [7] were proposed by the Web Accessibility Initiative[1] (WAI). WCAG 1.0 guidelines define specific testing techniques or checkpoints which refer to accessibility issues in a more accurate way. Depending on the way a checkpoint impacts on the accessibility of a web page, each checkpoint has a priority assigned (1, 2 or 3 from more to less impact). In addition, based on these priorities three conformance levels are defined:

- Conformance level A: all priority 1 checkpoints are satisfied.
- Conformance level AA: all priority 1 and 2 checkpoints are satisfied.
- Conformance level AAA: all priority 1, 2 and 3 checkpoints are satisfied.

New versions of these guidelines are being currently developed. The last draft of WCAG second version was released in April 2006 [6] and proposes a new guideline

---

[1] http://www.w3.org/wai

concept. This set of guidelines incorporate a new accessibility description as it defines the properties an accessible website has to accomplish. Similarly to the previous version of WCAG, each checkpoint defines three priority and analogous conformance levels. According to this description, an accessible website should fulfil these four guidelines:

- Make content PERCEIVABLE for any user.
- Make content and controls UNDERSTANDABLE to as many users as possible.
- Use ROBUST web technologies that maximize the ability of the content to work with current and future accessibility technologies and user agents.
- Ensure that interface elements in the content are OPERABLE by any user.

Searching is a significant activity when accessing the WWW. Kobayashi and Takeda [13] state that 85% of users use search engines when seeking for information on the Web. However, results are not tailored to the need of users with disabilities and they may find barriers when trying to access to the websites in results. According to a study carried out with visually impaired users by Andronico et al. [2] only 38% of them find search engines results useful while 90% of sighted users do not have any problem. This may be the reason why only 23% of visually impaired users versus the 70% of sighted users frequently use search engines. Ivory et al. [12] suggest that providing additional page features and re-ranking according to users visual abilities would be a way to improve their search experience. In this sense, this paper aims at exploring web accessibility issues on traditional Information Retrieval mechanisms such as search engines. It proposes a conceptual framework for including web accessibility measures in information retrieval processes. In addition, it presents a prototype implemented based on the proposed framework.

2   Web Accessibility as a Quality Measure

Some research aim at incorporating quality metrics in informational retrieval systems can be found in the literature such as the one presented in [20]. However, they do not consider web accessibility as a quality measure of web applications even if they take into account some usability related properties.

Many authors consider accessibility closely related to usability as they both enhance user satisfaction, effectiveness and efficiency. According to some of them, accessibility can be understood as a subset of usability. In fact, the concept of accessibility is related to the absence of physical or cognitive barriers to using the functionality implemented in a website, such as navigation, information searching, etc. Although diverse methods and tools for web usability evaluation exist [11], accessibility assessment has not been sufficiently developed, even though accessibility measurement, rating and assessment is essential in determining website quality. The lack in such accurate measures and tools for automatically calculate them may be the reason why accessibility is sometimes forgotten.

As far as standards related to quality are concerned, the ISO 9126-1 standard [10] defines six software product quality characteristics: functionality, reliability, efficiency, usability, maintainability and portability. For evaluation purposes, it also defines a quality model for software product quality and it should be used in conjunction with the ISO/IEC 14598-1 [9] providing methods for the measurement, assessment and evaluation of software product quality. When specifically refereeing to websites, specific models such as 2QCV3Q by Mich et al. [15] have been proposed. Even if they include several aspects related to both usability and accessibility, web accessibility is not considered as an important property of websites.

All the approaches for measuring the quality of software products coincide in the importance of creating adequate metrics in order to efficiently perform the quality evaluation process. The most accepted and used web accessibility metrics are qualitative ones proposed by the WAI in the WCAG 1.0 document. As previously mentioned, these metrics assign 0, A, AA or AAA value depending on the fulfilment of the WCAG 1.0 guidelines. They are not accurate enough in order to rate and classify websites according to their accessibility level. A website fulfilling only all priority 1 checkpoints would obtain the same accessibility value as another website fulfilling all priority 1 checkpoints and almost all priority 2 checkpoints: both of them would get the A level conformance. These criteria seem to be based in the assumption that if a webpage fails to accomplish one of the guidelines in a level, it is so un-accessible as if it fails to fulfil all of them. That is true for some users, but in general it is essential to have not only a reject/accept validation, but a more accurate graduation of the accessibility. Thus, as stated by Olsina and Rossi [16], defining quantitative accessibility metrics is necessary in order to overcome this situation. Moreover, they are essential in order to perform an adequate rating of websites and consequently for including web accessibility measures in informational retrieval systems. Development of adequate and accurate metrics will encourage developers to consider accessibility property as a website quality measure. Consequently, accessibility could be included in several processes such as information retrieval.

3   Related Work

There are two main components to be included in such a framework for including web accessibility metrics into informational retrieval systems. As stated in section 2, development of accurate accessibility metrics is essential as the websites should be rated and ranked based on them. These metrics will be calculated based on the results obtained after performing a comprehensive accessibility evaluation according to existing sets of guidelines. Both components should be automatized processes as the objective is to include them into other automatic processes which are the information retrieval systems. The following sections present some of the existing research works related to both components: web accessibility metrics and automatic accessibility evaluation.

3.1 Web Accessibility Metrics

Sullivan and Matson [18] evaluate eight checkpoints from WCAG 1.0. As a result, the so-called "failure-rate" is a proportion between potential errors and real errors. Therefore, the result range goes from 0 to 1. It is a naive approximation since other factors such as error impact, error nature (whether checkpoints are *errors*, *warnings* or *general warnings*) and other requirements explained in the following section are not taken into account.

$$failure\_rate = \frac{real\_errors}{potential\_errors}$$

Hackett et al. [8] proposed the WAB formula (Web Accessibility Barrier). This formula uses as input parameters the total pages of a website, total accessibility errors as well as potential errors in a web page and error priority. However, the returned marks are not restricted to a limited range of values. Therefore, it can be useful only for ranking web pages according to their accessibility level. The drawback of this metric is that considering the result for a unique web page, it is not possible to have an accessibility reference since there are no boundaries for good or bad accessibility levels. The formula for a single web page is calculated for all WCAG checkpoints found in the page:

$$WAB\_score = \sum \frac{real\_errors}{potential\_errors \times priority}$$

Bühler et al. [5] propose aggregation models order to adapt measurement to different disabilities groups. A simplification of that model is the following:

$$A(u) = 1 - \prod \left(1 - R_b S_{ub}\right)$$

Where R is the evaluation report and S is a severity value from 0 to 1 (for each barrier type b and user group u). However, these metrics are still in a developing stage until better results are obtained. These metrics are supposed to be integrated in the web accessibility benchmarking framework defined in [17]. To our best knowledge, there is not any implemented process for automatically calculating web accessibility metrics even if there are several proposed in the literature.

3.2 Automatic Accessibility Evaluation

In recent years, a great deal of tools for automatic accessibility evaluation has been developed[2]. Even if most of them evaluate predefined sets of general purpose accessibility guidelines such as WCAG 1.0 or Section 508, they vary in the number and type of test cases implemented. As stated in [4] evaluation results returned by different tools for the same website may vary.

In 2004, Abascal et al. [1] proposed the novel approach for automatic accessibility evaluation: separation of guidelines from the evaluation engine. The usefulness of this

---

[2] http://www.w3.org/WAI/ER/tools/

approach relies on its flexibility and updating efficiency. Adaptation to new guideline versions does not imply re-designing the evaluation engine but guidelines editing. The guidelines specification language is based on XML. Following this approach, in 2005, Vanderdonckt and Bereikdar proposed the Guidelines Definition Language, GDL [19] and recently Leporini et al. the Guidelines Abstraction Language, GAL [14].

4   Proposed Framework for Information Retrieval Systems

One of the objectives of this paper is to present an architecture proposal where information retrieval systems are enriched with web accessibility analysis. We propose a framework which produces results with the most suitable websites according to their content and end-user specific characteristics. In this sense, not only will be relevant the suitability of the web content returned but its accessibility level also will be taken into account. The implementation should adequately combine ranking regarding website relevance accessibility analysis. Research has been carried out on content analysis in order to rate websites but web accessibility has not been much studied in this sense. Thus, the proposed framework is as modular as possible in order to guarantee that further add-ons can be easily integrated. The architecture can be observed in Figure 1.
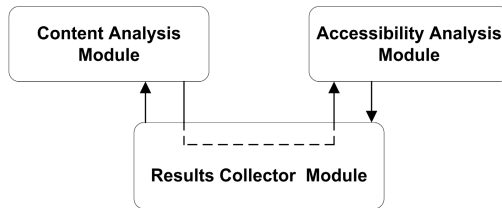

Fig. 1. Model of the proposed architecture

As can be appreciated, the architecture is composed of three independent modules:

Content Analysis Module (CAM) performs the content analysis based on information retrieval methods, techniques. It produces a list of websites rated according to their suitability for a specific query.

Accessibility Analysis Module (AAM) performs web accessibility evaluation. The previously mentioned automatic accessibility evaluation tools could be adequate for implementing this module. Integrating quantitative accessibility metrics into this module is essential in order to obtain accurately ordered lists of websites.

Results Collector Module (RCM) ensures that the information provided by the other two modules is adequately combined. Then, results according to their content and accessibility level will be produced and returned to the end-user.

Due to the modularity of the proposed framework, existing tools such as automatic accessibility tools, search engines, etc. can easily interoperate. In addition, this modular architecture will guarantee a correct independent testing for each module in order to obtain reliable systems.

5   Implementation of a Prototype

A prototype system has been implemented based on the proposed framework in order to test its usefulness. In this sense, three main tasks have been performed:

- Define accurate quantitative metrics.
- Automatic calculation of metrics.
- Integration of accessibility evaluation, metrics calculation and content analysis.

The following sections describe these tasks.

5.1   Quantitative Metrics for Web Accessibility

Accessibility problems in evaluation reports are classified by evaluation tools are classified in three main groups:

- Automatic tests (errors): these problems should not require human judgment to check their validity.
- Manual or semi-automatic tests (warnings): human judgment is necessary to check potential problems associated to particular fragments of code implementing the page.
- Generic problems: human judgment is necessary to check potential problems that cannot be associated to any code fragments; these problems arise in every web page. E.g. WCAG 1.0 14.1 checkpoint: "Use the clearest and simplest language".

The principal objectives when designing the metric have been the following:

- The value obtained by the metric should be meaningful in terms of accessibility level prediction.
- The metric should be useful for ranking web pages according to their accessibility level.

5.1.1   Requirements, Assumptions and Facts
This section defines the requirements, assumptions and facts considered when developing accessibility metrics.

Requirement 1: *The result of the metric should be normalized.*
In order to classify websites according to their accessibility a limited ratio scale from 0 to 100 is chosen so that results of the final quantitative accessibility value are expressed in a percentage scale. The closer the result of the metric is to 0 the less accessible the website is and the closer to 100 is the more accessible it is. This leads us to classify web pages according to their accessibility guidelines conformance percentage.

Requirement 2: *The metric should give one value for each accessibility attribute, as well as an overall value for each page.*

Although automatic accessibility evaluation reports returned by EvalAccess refer to WCAG 1.0 guidelines, these are mapped into WCAG 2.0 guidelines: Perceivable, Operable, Understandable and Robust[3].

Apart from an accessibility quantitative value for each guideline, an overall accessibility value based on POUR guidelines is also calculated. However, metric calculation according to these guidelines is useful to get a general idea of how accessible a page is.

Assumption 1: *Besides total number of errors for each checkpoint in the web page, the metric should also take into account the total number of times each checkpoint has been tested.*

The metric should not be based on the absolute number of found errors but in the relative number of found errors in relation to the number of tested cases [18]. That is, the ratio of errors and number of tested cases. For instance, if we analyze a web page that contains 5 images without text equivalent and another one containing 10 where 5 of them have a text equivalent, the second web page should obtain better accessibility score, since the failure percentage is 100% (5 of 5) and 50% (5 of 10) respectively.

Assumption 2: *The priority of an unfulfilled checkpoint should be reflected in the final result* [8].

Priority is an ordinal-scale qualitative variable of three levels: priority 1, priority 2 and priority 3. It is stated by the WAI that priority 1 checkpoints have more impact on the accessibility level of a web page than priority 2 checkpoints and so on. Consequently, their weight in the value obtained by the metric should be different. In order to empirically tune the weights, different values are assigned to the weights in some test files with different accessibility level. The unique restriction when selecting these weights is that $1 > priority1\_weight > priority2\_weight > priority3\_weight > 0$.

These test files have a determined failure rate. In addition, they are simple enough to manually calculate a quantitative metric. Different values were given to weights to calculate the quantitative accessibility value of each file using the metric defined next. The criterion for selecting the most appropriate weights was the similarity of the accessibility value to the failure rate on test files. The test files used are the following:

- **Low Accessibility level web page (LA):** This test file contains images without text equivalent, tables without summary, some links which open pop-up windows, auto-refreshing and wrong document language definition.
- **Accessible web page (A):** This test file contains the same potential errors but such that they do not cause any accessibility error: images have text equivalent, tables have summary, links do not open new windows, there is no auto-refresh and language is well defined.
- **Medium Accessibility level web page (MA):** Elements in this test file are the same than in Low Accessibility file but half of potential errors are actual errors.

---

[3] http://www.w3.org/TR/WCAG20/appendixD.html

- **Worse than MA:** 3/4 of the previously mentioned potential errors are actual errors.
- **Better than MA:** This test file is composed of the same elements but 1/4 of them have an actual error.
- **Empty web page (E):** This test file only contains the necessary structural HTML tags without any content element.

Assumption 3: *Generic problems should not have influence on the final metric.*
When performing an automatic evaluation, all web pages get the same report of *generic problems* in order to manually check the referred checkpoints. Thus, a metric based on automatic evaluation should not take into account these checkpoints.

Fact 1: *The interval where the metric results for lowest ratios of errors and tested cases are situated has to be spread.*
We have empirically tested that in each POUR guideline, the ratio of errors over potential errors, the failure rate, tends to be very low. Thus, it is difficult to discriminate among different pages since they all get similar accessibility values. The function in Figure 3 would be an approach to the ideal hyperbole in Figure 2. In this hyperbole, the closer to 0 it is the error and tested cases ratio (*E/T*), the higher it will be discriminated. The advantage of this approach is that the value of x' can be empirically assigned, in order to easily control the height allocated to the failure rate *E/T*. This feature makes possible to increase or decrease the variability in any interval depending on the experimental results obtained modifying *a* and *b* variables. For this paper, we used *a*=20 and *b*=0.3 following an empirical approach similar to the one carried out in *Assumption 2*.
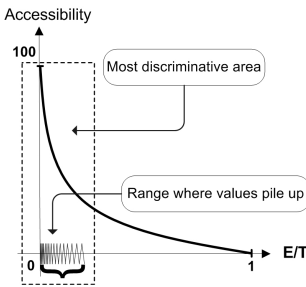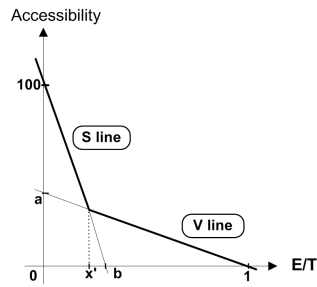

Fig. 2. Ideal hyperbole

Fig. 3. An approach to the hyperbole

According to the hyperbole approach, if *E/T* ratio is less than the intersection point x' the accessibility will be calculated using *S* line. Otherwise, *V* line is used. x' value depends on variables *a*, *b* and tested cases.

$$x' = \frac{a-100}{\dfrac{a}{T} - \dfrac{100}{b}}$$

x' point calculation

$$A = E \times \left(\frac{-100}{b} \right) + 100$$

S line formula

$$A = \left(\frac{-a}{T} \times E \right) + a$$

V line formula

Fact 2: *Manual tests (warnings) should be taken into account in the same way than errors.*

Our research concluded that the failure rate is highly correlated for errors and warnings when checkpoints were grouped by their guideline (POUR) and by their priority (1, 2, 3). Therefore, tested cases in warning checkpoints will fulfil the accessibility guidelines with the same ratio than their equivalent errors subgroup.

5.1.2   Variables, Constant Values and Final Metric

Table 1 contains a description of variables, constants and final values of the metric according to the requirements, assumptions and facts. Some constants are tool dependent while others are guideline-set dependent.

This metric proved to correlate positively with a research carried out by experts on Spanish universities' websites classification according to their accessibility level as presented in [3].

Table 1. Variables, constants and metric for accessibility quantitative measurement

Variables	Description	range
E	number of accessibility **e**rrors in each checkpoint	0-∞
T	number of **t**ested cases in each checkpoint	0-∞
A	variable for hyperbole approach customization (y axis)	0-100
B	variable for hyperbole approach customization (x axis)	0-1
Constants		**value**
N	Total number of checkpoints (EvalAccess)	44
N_{xy}	Number of checkpoints in guideline $x \in \{P,O,U,R\}$, where P stands for Perceivable, O for Operable, U for Understandable and R for Robust, and type $y \in \{error, warning\}$	
N_x	number of checkpoints in guideline $x \in \{P,O,U,R\}$	
$N_{x,error}$	total number of automatic tests	18
$N_{x,warning}$	total number of manual tests	25
$N_{P,error}$	*error* checkpoints in Perceivable	4
$N_{O,error}$	*error* checkpoints in Operable	3
$N_{U,error}$	*error* checkpoints in Understandable	3
$N_{R,error}$	*error* checkpoints in Robust	7
$N_{P,warning}$	*warning* checkpoints in Perceivable	11
$N_{O,warning}$	*warning* checkpoints in Operable	1
$N_{U,warning}$	*warning* checkpoints in Understandable	6
$N_{R,warning}$	*warning* checkpoints in Robust	7
Weights		**value**
k_1	*priority 1* items	0.80
k_2	*priority 2* items	0.16
k_3	*priority 3* items	0.04
Metric		**range**
A_{xyz}	Accessibility of priority $z \in \{1,2,3\}$ in $x \in \{P,O,U,R\}$ guidelines and in $y \in \{error, warning\}$ type of checkpoints.	0-100
A_{xy}	Accessibility of $x \in \{P,O,U,R\}$ guidelines in $y \in \{error, warning\}$ type of checkpoints.	0-100
A_x	Accessibility of $x \in \{P,O,U,R\}$ guidelines	0-100
A	Mean accessibility value	0-100

5.2 Automatic Metric Calculation

We selected EvalAccess accessibility evaluation tool for integrating the automatic metric calculation due to its flexible architecture. In addition, the evaluation reports returned by EvalAccess are formatted based on a specific XML-Schema. Consequently, gathering of all the necessary data for metric calculation such as checkpoint type (*error* or *warning*), the times a checkpoint is tested (*T* variable), the times each test fails to be conformant with the guidelines definition (*E* variable), and its priority is straightforward. All these parameters are grouped in 2 groups (errors and warnings). Each group contains 12 subgroups classified by their priority in WCAG 1.0 (3 priorities) and their membership in the WCAG 2.0 four POUR guidelines according to the previously mentioned mapping.

Therefore, the quantitative accessibility metric takes into account the previously mentioned facts, assumptions and requirements. The quantitative accessibility metric is calculated by the following algorithm:

*for **x** in each checkpoint in a guideline {P,O,U,R} loop*
    *for **y** in each type of checkpoint {error, warning} loop*
        *for **z** in each priority{1,2,3} loop*
            x'=calculate_x'_point(a,b,T)

$$\text{if} \left(\frac{E}{T} < x' \right) \text{then}$$

                A_{xyz}=calculate_S_line(b, E)
            else
                A_{xyz}=calculate_V_line(a, E, T)
            end if
        end loop

$$A_{xy} = \sum_{z=1}^{3} k_z \times A_{xyz} \quad \Leftarrow \textbf{ Step a}$$

    end loop

$$A_x = \frac{\sum\limits_{y} N_{xy} \times A_{xy}}{N_x} \quad \Leftarrow \textbf{ Step b}$$

end loop

$$A = \frac{\sum\limits_{x} N_x \times A_x}{N} \quad \Leftarrow \textbf{ Step c}$$

In **Step a** we get all the A_{xy} values such as $A_{P,error}$. This means that we get values for *error* checkpoints in Perceivable guideline. In **Step b** an average value for each POUR guideline is calculated by weighting A_{xy} value with the number of *errors* and *warnings* in *x* guideline. Finally, we get an overall accessibility value in **Step c** weighting each POUR guideline with the number of checkpoints they contain. The last two steps take into account the number of guidelines in each category (guidelines and type) in order to distribute the weights in a well-balanced way.

5.3 Integrating Web Accessibility Evaluation and Content Relevance Analysis

In order to verify that the proposed architecture fits in a real world, a customized search engine has been developed following the described framework. In the

implemented prototype, the previously mentioned automatic evaluation tool, EvalAccess, and a conventional search engine interoperate in order to return the results of the searching tasks ranked by accessibility level and content relevance. Both services can be easily accessed and used since they are implemented as Web Services (WS). Therefore, the implemented prototype is based on the interoperation of different Web Services. The development of each module in this framework is explained below:

Content Analysis Module: Nowadays, it is unmanageable for a small company to create an outstanding information retrieval system. Large amounts of infrastructure (hardware), reliable operative systems, applications to run over servers and prepared staff is required, which implies a significant economical investment. Thus, we take advantage of the services offered by one of the most used and known search engines, Google. Google provides developers with an API[4] which facilitates making requests to its search engine. Therefore, the function of this module is fulfilled by the techniques and methods implemented in Google since it generates an ordered list of websites based on the specified query. Thanks to the abstraction layer provided by this API, the interaction with Google Web Service is performed in a transparent way. For this reason, there is no need to directly use the SOAP protocol.

Accessibility Analysis Module: EvalAccess [1], the previously mentioned automatic accessibility evaluation tool has been integrated into the framework. It evaluates web pages according to the WCAG 1.0 guidelines and returns a machine understandable accessibility errors report formatted in XML. Web accessibility quantitative metrics can be easily applied to the information returned in this report. Since it is implemented as a Web Service, clients using SOAP protocol could get the XML report and exploit its results.

Results Collector Module: This module coordinates user requests with the different Web Services. This means coding and decoding data gathered from results of Google WS query and the XML accessibility evaluation report returned by EvalAccess WS. Then, each accessibility report is exploited in order to get all the necessary information and calculate the accessibility value based on the quantitative metrics explained in the previous section. Finally, it returns the results ordered by its accessibility.

A user web interface has been integrated into the Results Collector Module. The system shows a common search engine web interface which communicates with the core of the modules. This interface provides different options related to accessibility guidelines defined in WCAG 2.0: results re-ranking according to a value for a guideline in POUR or their average value. The results page is a list of items ordered by its accessibility as can be see in the Figure 4. Each item contains a title, a URL, its accessibility value as well as a snippet where the search keyword is contextualized.

The overall latency when evaluating web accessibility and calculating the metric is higher than in traditional search. However, the proposed model is still valid since it is useful in order to remove existing barriers.
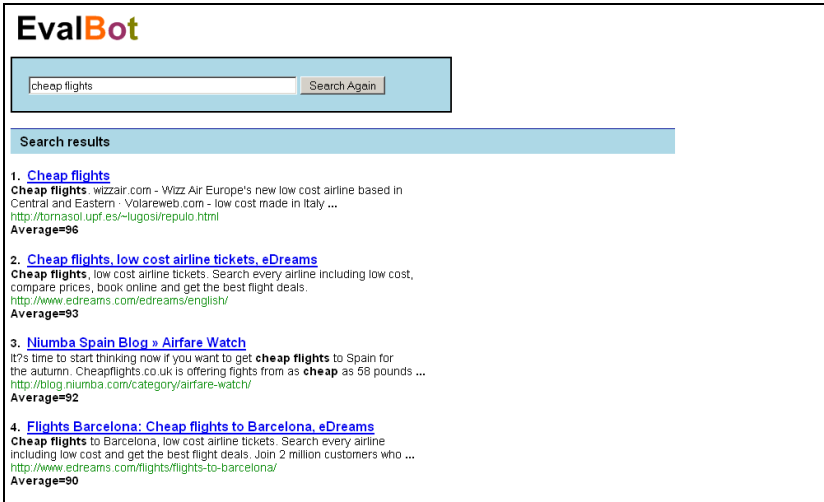
---

[4] http://www.google.com/apis/

Fig. 4. Search result page

6   Results and Discussion

Several tests have been carried out in order to evaluate the performance of this framework. One testing case is presented in this section for discussing the results obtained by the implemented system. Firstly, a searching query was defined. In this case, this query was: "cheap flights". Then, this query has been introduced to Google search engine as well as to our Customized Search Engine (CSE). Table 2 shows the first 10 results returned by the system and their order in the results list. In addition, errors yielded by automatic evaluation are grouped by their priority.

Table 2. Results obtained by Google and CSE. Automatic evaluation results of EvalAccess tool grouped by their priorities (P1- Priority 1 errors, P2 - Priority 1 errors, P3 - Priority 3 errors) are also provided.

URL	Google	CSE	P1	P2	P3
www.cheapflights.co.uk	1	5	0	15	265
www.easyjet.com	2	2	0	29	91
www.ryanair.com/site/EN	3	6	45	230	995
www.skyscanner.net	4	8	0	9	69
www.cheapflights.com	5	3	0	7	186
www.flightline.co.uk	6	4	0	6	128
www.flybmi.com/bmi	7	1	0	2	60
www.bmibaby.com	8	10	2	75	169
www.aerlingus.com	9	-	0	5	100
www.openjet.com	10	-	0	1	3
www.bargainholidays.com	-	7	29	67	262
www.cheapestflights.co.uk	-	9	8	143	198

It can be appreciated in Table 2 that the results obtained by both search engines differ, as two websites included in Google result (www.aerlingus.com and www.openjet.com) are not in CSE results. Other two websites have been included in CSE results instead of these two websites (www.aerlingus.com and www.openjet.com). There are discrepancies between Google API and Google.com results due to the fact that apparently queries are made against different indexes. It could be discussed whether the trade-off of content ranking versus the accessibility ranking is really worthy. This is an approach to the conceptual model explained in Section 4 and the evidence that this model works. However, query results could be listed according to Google's results and each result could be marked with its accessibility value without re-ranking it so the user would decide to browse it or not as proposed in [12].

Web accessibility analysis of these websites has been performed by using the service provided by one of the automatic accessibility evaluation tool, EvalAccess. Columns P1, P2 and P3 show the number of automatically detected accessibility errors in the returned websites. The first website returned by Google (www.cheapflights.co.uk/) is also included in CSE result but in the fifth position. Evalbot returns as the first result the website of www.flybmi.com/bmi/. According to the data, the result obtained by the CSE has less accessibility problems than the one obtained by Google, since www.cheapflights.co.uk website errors are P1-0, P2–15, P3–265 whereas www.flybmi.com/bmi website errors are P1-0, P2-2, P3-60. Then, end-users would easier access to the result obtained by the CSE. However, some contradictory data can be found in these results. For example, www.cheapflights.com website accessibility errors are P1-0, P2-7, P3-186 and www.flightline.co.uk website accessibility errors are P1-0, P2-6, P3-128. According to this data, the www.flightline.co.uk website would have less accessibility barriers than the other one. But the CSE results show that it is placed behind in the results list. As explained in **Assumption 1**, this is due to the accessibility quantitative metric is based in the percentage of found errors divided by the number of tested cases (potential errors).

7   Conclusions and Future Work

This paper proposes a framework for incorporating web accessibility information into information retrieval systems. This significantly improves user's satisfaction when searching for information in the WWW, as they could obtain search results ordered by content relevance as well as accessibility level.

The necessity of a quantitative metric for web accessibility assessment has been demonstrated in this paper. If accurate discrimination among web pages wants to be done, these measures are key factors. The proposed metric aims at being a general approach to accessibility awareness in information retrieval processes. The metric does not take into account specific users grouped by disabilities (hearing, visually or physically impaired). However, besides an average value, POUR values are given because the system performs a mapping between WCAG 1.0 and WCAG 2.0 draft.

A prototype based on the proposed framework has also been developed. It enables users with disabilities making search queries which return websites listed according to both Google information retrieval criteria and accessibility criteria. The measure of

web accessibility is based on EvalAccess, an automatic accessibility evaluation Web Service which returns accessibility reports formatted in XML. This feature facilitates applying quantitative accessibility metrics to the evaluation results.

The most significant disadvantage when using the prototype is the increase in the response time comparing with the original search engine. This latency could discourage users to performing search tasks with this prototype. We are currently working on the next version of the framework which will improve the response time.

Acknowledgements

References

1. Abascal, J., Arrue, M., Fajardo, I., Garay, N., Tomás, J.: Use of Guidelines to automatically verify web accessibility. International Journal of Universal Access in the Information Society 3(1), 71–79 (2004)
2. Andronico, P., Buzzi, M., Castillo, C., Leporini, B.: Improving search engine interfaces for blind users: a case study. International Journal of Universal Access in the Information Society 5(1), 23–40 (2006)
3. Arrue, M., Vigo, M., Abascal, J.: Quantitative Metrics for Web Accessibility Evaluation. In: Lowe, D.G., Gaedke, M. (eds.) ICWE 2005. LNCS, vol. 3579, Springer, Heidelberg (2005)
4. Brajnik, G.: Comparing accessibility evaluation tools: a method for tool effectiveness. International Journal of Universal Access in the Information Society 3(3-4), 252–263 (2004)
5. Bühler, C., Heck, H., Perlick, O., Nietzio, A., Ullveit-Moe, N.: Interpreting Results from Large Scale Automatic Evaluation of Web Accessibility. In: Miesenberger, K., Klaus, J., Zagler, W., Karshmer, A.I. (eds.) ICCHP 2006. LNCS, vol. 4061, pp. 184–191. Springer, Heidelberg (2006)
6. Caldwell, B., Chisholm, W., Slatin, J., Vanderheiden, G. (eds.). (2006, April 27) Web Content Accessibility Guidelines 2.0. (Working Draft). http://www.w3.org/TR/WCAG20/
7. Chisholm, W., Vanderheiden, G., Jacobs, I. (eds.) Web Content Accessibility Guidelines 1.0. (May 5, 1999) http://www.w3.org/TR/WAI-WEBCONTENT/
8. Hackett, S., Parmanto, B., Zeng, X.: Accessibility of Internet websites through time. In: Proceedings of 6th International ACM SIGACCESS Conference on Computers and Accessibility, pp. 32–39 (2004)
9. International Organization of Standardization (ISO), Information Technology - Software Product Evaluation (ISO 14598). Geneva, Switzerland (1999)
10. International Organization of Standardization (ISO), Software Engineering - Product Quality - Part1: Quality Model (ISO 9126-1). Geneva, Switzerland (2001)
11. Ivory, M.Y., Hearst, M.A.: The state of art in automating usability evaluations of user interfaces. ACM Computing Surveys 33(4), 470–516 (2001)
12. Ivory, M.Y., Yu, S., Gronemyer, K.: Search result exploration: a preliminary study of blind and sighted users' decision making and performance. In: CHI Extended Abstracts, pp. 1453–1456 (2004)

13. Kobayashi, M., Takeda, K.: Information Retrieval on the Web. ACM Computing Surveys 32(2), 144–173 (2000)
14. Leporini, B., Paternò, F., Scorcia, A.: Flexible tool support for accessibility evaluation. Interacting with Computers 18(5), 869–890 (2006)
15. Mich, L., Franch, M., Gaio, L.: Evaluating and Designing Web Site Quality. IEEE Multimedia 10(1), 34–43 (2003)
16. Olsina, L., Rossi, G.: Measuring Web Application Quality with WebQEM. IEEE Multimedia 9(4), 20–29 (2002)
17. Snaprud, M.H, Ulltveit-Moe, N., Pillai, A.B., Olsen, M.G.: A Proposed Architecture for Large Scale Web Accessibility Assessment. In: Miesenberger, K., Klaus, J., Zagler, W., Karshmer, A.I. (eds.) ICCHP 2006. LNCS, vol. 4061, pp. 234–241. Springer, Heidelberg (2006)
18. Sullivan, T., Matson, R.: Barriers to use: usability and content accessibility on the Web's most popular sites. In: Proceedings of the ACM Conference on Universal Usability 2000, pp. 139–144 (2000)
19. Vanderdonckt, J., Bereikdar, A.: Automated Web Evaluation by Guideline Review. Journal of Web Engineering 4(2), 102–117 (2005)
20. Zhu, X., Gauch, S.: Incorporating quality metrics in centralized/distributed information retrieval on the World Wide Web. In: Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 288–295 (2000)