Detecting Information Waste on Websites

Sjoerd van der Meer
University of Twente
P.O. Box 217, 7500AE Enschede
The Netherlands
s.vandermeer-1@student.utwente.nl

ABSTRACT
With the rapid growth of the internet, the amount of information waste is growing as well. However, it is not clear how much of this information waste is caused by websites. This paper describes a model using known and validated metrics. To make an estimate on the amount of information waste on the internet, this paper describes a proof of concept to show how data mining techniques can be used to estimate the amount of information waste on the internet caused by websites.

Keywords
information waste, data mining, website quality

1. INTRODUCTION
This paper is part of an ongoing research on the effects of information waste on the global carbon emission footprint of the internet. Dietz developed an automatic information waste detector which is able to detect, with an accuracy rate of at most 80%, whether a file is waste or not [4]. This tool is however developed for the Windows operating system and validated on a small group of test users.

The next step in this research is a study on the effect of information waste on websites. With the rapid growth of the internet nowadays, IDC expects the digital universe to expand from 0.8 ZB (10^{21}B) in 2009 to 35 ZB in 2020 [14], it is likely that the amount of information waste on websites grows as well.

Information waste is files which are unnecessary and are the consequence of human limitations in knowing what files are useless and could thus be removed[4]. Therefore, information waste is a subjective metric. It needs user input to detect whether or not a file (or website) is information waste. In respect to websites we would replace file(s) with information on websites. Therefore we define: Information waste on websites is information which is unnecessary and is the consequence of human limitations in knowing what information is useless and could thus be removed.

While search engines are a good way to find relevant websites, they do not check whether the information on these websites is correct. In fact there is currently no publicly known implemented method that can define a website as being information waste without user input. That is, given a website, define without user input whether the information on this website is information waste or not. With regard to the growth of the internet, we feel that this is a problem. The goal of this research is to provide such a model. To this end, this research provides known subjective and objective metrics that identify a website as being information waste or not. To automatically detect information waste on a website however, some model is needed to find the connection between the subjective and objective measures. For this, a proof of concept is provided for the development of a information waste model using data mining techniques. Finally the model is tested on 605 websites.

2. RESEARCH QUESTIONS
The goal is to provide the means to estimate the amount of information waste on websites on the internet. Therefore our first research question is:
How much of the current websites on the internet can be defined as being information waste?

Before this question can be answered, a model needs to be developed that can estimate whether a website is information waste. This model should work using only objective attributes available with site analyzers. Since currently there are no models that allow the detection of information waste on websites a second research question is formed: Is it possible to develop software that can automatically identify information waste on the internet?

To be able to develop this software, we first need to identify which methods can be used to classify the information quality on a website. These methods should give a reliable classification. Henceforth, a final research question is formed: What reliable methods can be used to classify the value of a website?

3. THEORY
A lot of research has been done in the area of website quality. Most notably by Palmer [11], Yang et al.[16] and Eppler et al. [5]. They collected various metrics, both objective and subjective, to classify the value of websites. Although these researches are relatively old for the fast changing internet, they are still being used in relative new researches, for example [13] [8] [12]. They are considered well validated and reliable. At the time these researches were performed most of the current content of websites was already present.
3.1 Subjective Value

According to McKinney et al.\cite{10} the subjective value of websites is the ‘users’ preception of the quality of information presented on a website’. For this Eppler et al. highlighted information comprehensiveness and quality/correctness of information\cite{5}. Similarly, Yang et al. emphasized the relevance of information and information comprehensiveness\cite{16}. Palmer noted the quality/correctness of information\cite{11}. Summarized these give:

- **Content Quality/Correctness** (from Palmer\cite{11} and Eppler et al. \cite{5})
  
  Content quality is used to determine whether the information on a site is correct. This is necessary to determine the information quality of a website since a low content quality means a low information quality and vice versa.

- **Relevant Information** (from Yang et al.\cite{16})
  
  The next metric is Relevant Information. It is necessary to determine whether a site delivers relevant information or not to determine its quality.

- **Information comprehensiveness** (from Yang et al.\cite{16} and Eppler et al. \cite{5})
  
  The last subjective metric is information comprehensiveness. For a website to have a high information quality, it’s information has to be comprehensive.

3.2 Objective Value

To get objective analytics from websites, many methods exist. For the most complete analytics a tool should be installed on every web server which measures all the metrics you need. These so called site analysers are mostly used by web administrators to monitor their websites. Some software packages allow this data to be sent to analyse servers like Google Analytics\footnote{1}. This has advantages for both Google (in this instance) and the web administrator. Google can use the data to optimize their search engine, optimize their advertisements and much more. Web administrators can more easily target specific customers using the integration with, for example, Google Adwords. An alternative to Google Analytics is W3counter\footnote{2}. Piwik\footnote{3} works the same as, for instance, Google Analytics except that you host the collected data on your own server.

Another option is to scrape the entire internet and generate the analytics from the results. This is mostly used in search engines like for instance Google\cite{3}. In Googles case it scrapes a webpage for links and follows them. They used a system called PageRank which is a probability distribution used to represent the likelihood that a person randomly clicking on links will arrive at any particular page.

A third option is to measure where the users browse to by installing a tool on the clients’ computers. This is, for example, used by Alexa\footnote{4}. It collects to which websites a user browses, how long he stays on that site, how he got there, etc. Because many users install the client, it gives a good indication of the general internet usage.

For this paper the third option will be used. The first option requires that every webserver has the same tool, which is infeasible. The second option requires hardware

\footnote{1\url{http://www.google.com/intl/en/analytics/}}\footnote{2\url{http://www.w3counter.com/}}\footnote{3\url{http://piwik.org/}}\footnote{4\url{http://www.alexa.com/}}

which goes beyond the scope of this research. The third however has data which can be freely searched by any user and gives a good representation of the internet. To get this information a small scraper will be written.

Similarly to the subjective metrics, a lot of objective metrics are provided by Palmer \cite{11}, Yang et al.\cite{16} and Eppler et al. \cite{5}.

For the objective metrics of websites, the following measures are collected using Alexa:

- **Access Speed** (from Palmer\cite{11}, Yang et al.\cite{16} and Eppler et al. \cite{5})
  
  Gives an indication whether a site feels ‘fast’. If a site feels slow it is expected that users might continue to a next site.

- **Links** (from Palmer\cite{11} and Yang et al.\cite{16})
  
  One of the most important objective metric is the number of incoming links. If a website has a lot of incoming links, it is expected to contain good information.

- **Frequency of Access** (from Dietz \cite{4} and Wijhoven et al. \cite{15})
  
  Defined as the number of unique monthly visitors to a site. If a site has a lot of visitors it most likely is valuable information.

The next metrics are metrics which are available at Alexa and might be useful. Because we will use data mining to create a model, extra objective metrics can be inserted during the process. This is further explained in section 4.4.

- **Time on site**
  
  If a user stays at a site for a long time, it most likely is good information. Precautions need to be taken with this metric because if a user keeps his browser open at a certain page while he is away or watching a youtube clip, it will give a false positive.

- **Bounce percentage**
  
  Bounce percentage gives the percentage of unique users which visited only one page on a certain website. Therefore this might give a good indication of the information quality. This one might also give false positives because it could also be a bad search by the user.

- **Global pageview percentage**
  
  Global pageview percentage gives the percentage of pages viewed on this website compared to the estimated total number of pageviews.

- **Global user percentage**
  
  Global user percentage gives an estimation of the percentage of global internet users who visit a specific site.
4. MODEL DEVELOPMENT

4.1 Sample set

Because the internet consists of at least 50 billion websites (according to worldwidewebsize\(^5\)) it is infeasible to test the entire internet. Therefore a subdomain was chosen for the sample set.

Since most of the data is collected from site and traffic analyzers a subdomain which they index was selected. Furthermore, access to the data is needed. While Google indexes a lot of sites (currently 50 billion), they are not likely to give this information. One site that allows and does give this information has been found. This is Alexa. Alexa does not give a complete coverage of the internet but we feel that it is sufficient for this proof of concept.

Because an user survey was needed, only English sites were chosen. Although this compromises the results, we feel that it gives a good enough sample to test. To prevent compromising the results further, a different source is chosen for the sample websites. Quantcast publishes a top million list everyday\(^6\). This gives a very large set that does not compromise the data gathered from Alexa. From this set a hundred sites, with the above mentioned restrictions, were randomly chosen as the sample pool. For this the complete set was first divided in 100 equal size sub sets. From each sub set one website was chosen randomly to be tested. This allows a fair representation of the internet.

4.2 Subjective rating

With the sample set defined, a tool is developed to create the sample data. The tool is written in Java. Java is chosen because it allows for easy cross platform programming using good supported stable libraries. It also allows for an easy creation of custom libraries. This is useful for the scraper of the objective metrics, which is used in multiple tools. Another reason is that the tool can easily be updated with extra scrapers when we use custom libraries.

The feedback tool allows a user to easily rank a website (see Figure 1). It consist of a link to open the current webpage and the following three metrics for the user to rank a website:

- Content Quality/Correctness
- Relevant Information
- Information comprehensiveness

The user of the tool was given an explanation of these metrics according to the description given in section 3.2. As soon as these metrics are filled in the user can press the add button to continue to the next website. The results are saved in a separate comma separated file which allows easy manipulation of the data.

4.2.1 Inter-rater reliability

All the websites were rated by one person. This gives a total of 100 samples. To show that this person has a good average view on information quality on websites a kappa inter-rater reliability was performed. The kappa inter-rater reliability shows the general agreement among two or more people. For this 10 websites were randomly chosen and rated by two other experts.

Most kappa inter-rater reliability tests are performed between just two persons (or raters). Therefore Fleiss’ kappa is used\([6]\). Fleiss’ kappa is able to simultaneously rate the reliability of agreement on multiple raters. For this, it will create a table to which all the results of all raters are added. That is, if two raters rated a websites content quality as 4 there will be a 2 corresponding to that metric and score. The results are shown in Table 3 in the Appendix. \(P_i\) is the extent to which the raters agree on that metric for that website. \(P_i\) is calculated using the following equation:

\[
P_i = \frac{1}{N} \left( \frac{1}{N} \sum_{i=1}^{k} n_{ij}^2 - n \right)
\]

Where \(N = \text{number of tests}, n = \text{number of raters}, k = \text{number of outcome}, i = \text{websites} \) and \(j = \text{ratings}. Since there are 10 websites rated and for each website 3 metrics are rated \(N = 30\). \(n = 3\) since three raters rated the websites and \(k = 5\) because there is a five-point scale for all the answers. With these values the general agreement \(P\) can be calculated using the following equation:

\[
P = \frac{1}{N} \left( \frac{1}{N} \sum_{i=1}^{N} P_i \right)
\]

Landis and Koch\([9]\) gave Table 1 for interpreting kappa values. This is however no generally accepted interpretation. The number of categories and subjects will affect the magnitude of the value. The kappa will be higher when there are fewer categories. For simplicity we will accept Landis and Koch’s conclusions but further research is needed.

The general agreement is 0.433 which means a moderate agreement. Because of this result 10 more random sites were rated by the same raters. This time however the results (shown in Table 4) were even worse. The general agreement was 0.289. Since the actual rater seemed to be inbetween the other two raters the research was continued.

---

\(^5\)http://worldwidewebsize.com/
\(^6\)http://www.quantcast.com/top-sites
but for a further research the data needs to be collected by multiple raters.

4.3 Objective rating
Since our goal is to develop a model that is able to determine the subjective value based upon objective metrics, these were limited to metrics that were available on the internet. These are the following.

- Links (Incoming)
- Frequency of Access (Pagerank)
- Time on site
- Bounce percentage
- Global pageview percentage
- Global user percentage
- Access Speed

As mentioned in section 4.2, a library was built to scrape the results of these metrics from Alexa. This library was implemented in the feedback tool developed for the subjective rating. It scrapes the websites’ objective metrics after the user clicked on the add button.

4.4 Classifier training
In order to classify websites, we will need the best fitting model. In order to determine this model data mining techniques are useful. Data mining is used to describe the relationship between the subjective and objective metrics. Data mining tries to discover hidden patterns within large sets of data with the use of algorithms. Since a tree is easier to implement in an other program, only algorithms that build tree models were considered. Simply put it works as follows. Consider a set consisting of 2 records and with 3 attributes say a, b and result. One result is ‘yes’ and the other ‘no’ and furthermore only a is different, say 3 for the first record and 1 for the other. The algorithms would then find a value for a which splits the largest part of the result set. In this case that would be 2. If it is lower then 2 result would be ‘yes’, else it would be ‘no’. Data mining algorithms try to find these sort of splits on large sets.

Before training the classifiers with the gathered data, the gathered data was processed to give the classifier as little confusion as possible. It was noted that a lot of data was incomplete. A closer look learned that most records only missed one or two measures with Access speed being the one missing in most. Since this metric applies more to the ‘feel’ of a website than the actual information waste, this metric was removed from the date. Furthermore another 50 sites were rated by the same person resulting in a total of 87 complete records. This is not a lot but enough to train a classifier in WEKA, the data mining application used for the training[1][7].

During training it was noted that the best results came from classifiers trying to classify a dual result outcome. Therefore the subjective data was combined into one value for the site being information waste or not. This was accomplished by adding up all the subjective values. If this value was smaller then 10, it was considered information waste and received the value ‘yes’, else ‘no’.

WEKA uses a ten fold validation technique which means that the data set is split in ten sections. The classifier is then trained with nine of these sections and tested with the tenth. Once this is done, the training is repeated until each section has been used for the validation.

The first algorithm that was used was the J48 classifier. It had a classifier rating of 77% but it’s tree consisted of one node; Waste = yes. This was probably the highest rating the J48 classifier could manage but it is not wanted. It would classify the entire internet as being information waste. Another reason why this tree was generated might be the relative small result set. Only 20 of 87 sites are ranked as not being information waste. This gives the classifier the option to create this tree with just one node.

As a solution the Random Forest algorithm[2] was considered. It builds not one tree but a whole forest of trees. Although this one gave the best results, it could not be used because it was not possible to get the resulting trees from the application. Therefore the best of both was found in the Random Tree implementation of WEKA. The implementation is also known as a Random Forest Tree. It uses part of the Random Forest algorithm but it’s result is a single tree. Therefore this solution is chosen. The results of the trained classifiers are summarized in Table 2 and the resulting decision tree is shown in Appendix B.

4.5 Implementation
To illustrate the use of the classifier, a tool has been developed that uses the top million list (see section 4.1) and tries to classify all the websites on this list. Since this takes a long time it has a simple User Interface which shows the user which website it is processing, what the current ratio is and how many websites were already processed. This tool is just for illustrating the working of the classifier implementation. The source code for the classifier implementation is shown in Appendix A.

A first test with the 150 websites which were originally tested showed that 4 sites (of the 87) no longer had objective values. Furthermore of the 83 that had objective values, 67 were still considered information waste giving an information waste rate of 80.7%.

The second test used the entire list containing a million sites. This test showed the limitation of this approach to scraping. During the test the ip address of the test computer was blocked by the firewall of Alexa. This resulted in an incomplete test in which only 605 websites were successfully processed (see Figure 2). The tool concluded that of these 605 websites 468 (77.4%) should be considered

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Correctly classified instances</th>
<th>Incorrectly classified instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>J48</td>
<td>67 (77.0%)</td>
<td>20 (23.0%)</td>
</tr>
<tr>
<td>RandomForest</td>
<td>63 (72.4%)</td>
<td>24 (27.6%)</td>
</tr>
<tr>
<td>Random/Tree</td>
<td>59 (67.8%)</td>
<td>28 (32.2%)</td>
</tr>
</tbody>
</table>

Figure 2. Classifier implementation
information waste.

4.6 Discussion

4.6.1 Resulted model
The accuracy of the classifier is lower then expected. Since this is just a proof of concept it is considered a good first step. The RandomForest classifier showed the potential of data mining but this will need a different implementation then WEKA can supply at the moment.

When looking at the resulted model (see Appendix B) it is clear that the most important objective metric according to the RandomTree classifier is the traffic rank (Frequency of Access). In the theory this was also noted. Furthermore the classifier has a lot more nodes then expected. A total of 49.

4.6.2 Implications
What are the practical implications of this model? Once further researched this model could be used as a companion to search engines. This makes it possible for search engines to not only provide results which match the search best, but also provide results that contain the highest information value.

Furthermore this model can be used as a feedback to site administrators. Many sites on the internet are no longer maintained but are still there. This model should be able to identify these sites and notify the owners. In the same way it can be used to give administrators feedback on their sites. Before this is possible however, the model needs to be expanded to allow for single pages to be rated.

There are also scientific implications to this model. At the moment we were unable to find a simular model for the detection of information waste on websites. This is a good start to attract further research to this area since we showed that it is possible to develop such a model.

5. CONCLUSION

5.1 Results
The third question of this research stated:

What reliable methods can be used to classify the value of a website?

To answer this question, this paper provided a number of subjective metrics able to measure the information value of a website. Furthermore it provided a number of objective metrics, which could easily be scraped of a website.

The objective metrics do have its limitations. The metrics were selected based upon what the scraper could provide. For further research this should be resolved.

Using these metrics the second question of this research can be answered:

Is it possible to develop a tool that can identify information waste on the internet?

This paper provided a proof of concept model able to identify information waste on websites with an accuracy of 67%. It also showed the possibility to improve this to 73%. This is not good enough for a positive yes, but it shows that with further research this is possible.

This model however is not implemented in a tool which could identify a specific site as being information waste. This however can be done quite easily using the source code of the java implementation given in Appendix A.

Using this model this paper attempted to identify:

How much of the current websites on the internet can be defined as being information waste?

To answer the question, a tool was written that checked 605 sites and, based upon the model, concluded that 77.4% of these websites was information waste. Since all of these websites were in the top million English sites, this is reason for further research.

It should be noted that during the test, the computer running this test was blocked by Alexa. This was due to an abnormal amount of requests. For further research this need to resolved (see section 5.2.3).

5.2 Further Research

5.2.1 Larger sample set
In order to develop an improved information waste detector a larger sample set should be created. This would allow better training for the classifier. As noted in section 4.4 the small data set was probably the cause of the J48 classifier not working. Also noted was the high difference between sites being rated as information waste or not. Although this might be the current situation on the internet, a larger sample set should prevent this from being an issue for the classifier training.

5.2.2 Other classification variables
The classification model can be extended with other variables. This current model was limited due to its objective metrics source. A good alternative might be Urlspy\textsuperscript{7}. Urlspy provides amongst other the number of pages on a website and the amount of external links (outgoing).

5.2.3 Other objective value source
As mentioned, the computer running this model was blocked by Alexa. Therefore, for further research a different source, which allows more data records to be collected, should be used. The previously mentioned Urlspy might be a good source. Also the option to create and run a scraper locally should be considered. This allows for much better control of the data. This does however imply the use of a dedicated server.

Finally, direct access to a analytics database should be considered as well. This would allow for very fast data collection and processing.

6. REFERENCES


\textsuperscript{7}http://www.urlspy.co.uk


APPENDIX

A. DECISION TREE SOURCE CODE

This appendix contains the code implementing the decision tree. A website's use value (information waste or not) is calculated using this method.

```java
public class DecisionTree {
    /**
     * Checks whether an objective contains a website which is ranked as waste.
     * Decision Tree generated using WEKA's RandomTree classifier
     * @param obj, The objective values of the website to be ranked
     * @return true if this objective is information waste, false if not
     * @requires !obj.toLine().contains("NULL")
     */
    public static boolean isWaste(Objective obj) {
        if (Double.parseDouble(obj.getValue("pageviewsPerc")) < 0) {
            return false; // Website is rated as no information waste
        } else {
            if (Double.parseDouble(obj.getValue("trafficRank")) < 2547.5) {
                return false; // Website is rated as no information waste
            } else {
                if (Double.parseDouble(obj.getValue("globalUsersPerc")) < 0) {
                    if (Double.parseDouble(obj.getValue("trafficRank")) < 3569135) {
                        return false; // Website is rated as no information waste
                    } else {
                        if (Double.parseDouble(obj.getValue("trafficRank")) < 3891262) {
                            return true; // Return true; // Website is rated as information waste
                        } else {
                            return false; // Website is rated as no information waste
                        }
                    }
                }
            }
        }
    }
}
```
< 2.2) {
  return true; // Website is rated as information waste
} else {
  if (Double.parseDouble(obj.getValue("bouncePerc")) < 41) {
    return false; // Website is rated as no information waste
  } else {
    if (Double.parseDouble(obj.getValue("uniquePageViewDaily")) < 2.85) {
      if (Double.parseDouble(obj.getValue("timeOnSite")) < 109.5) {
        return false; // Website is rated as no information waste
      } else {
        if (Double.parseDouble(obj.getValue("bouncePerc")) < 51.8) {
          return true; // Website is rated as information waste
        } else {
          if (Double.parseDouble(obj.getValue("pageviewsPerc")) < 0) {
            return true; // Website is rated as information waste
          } else {
            return false; // Website is no information waste
          }
        }
      }
    } else {
      return true; // Website is rated as information waste
    }
  }
} else {
  if (Double.parseDouble(obj.getValue("uniquePageViewDaily")) < 2.5) {
    return true; // Website is rated as information waste
  } else {
    return false; // Website is rated as no information waste
  }
} else {
  return false; // Website is rated as no information waste
}
} else {
  return true; // Website is rated as information waste
}
B. DECISION TREE

Because of its size it was not possible to put a visual model on one paper. Therefore a textual implementation is given.
Each level contains 2 options for that level. For the first that is: if pageviewsPerc smaller then 0 then this is no information waste, else if pageviewsPerc greater then or equal to 0 then continue with next level. The : means the tree stops at that end node and the site is rated as the value given there.

pageviewsPerc < 0 : no
pageviewsPerc >= 0
  | trafficRank < 2547.5 : no
  | trafficRank >= 2547.5
    | globalUsersPerc < 0
    | | trafficRank < 3569135 : no
    | | trafficRank >= 3569135
    | | | trafficRank < 3891262 : yes
    | | | trafficRank >= 3891262 : no
    | globalUsersPerc >= 0
    | | timeOnSite < 69.5 : yes
    | | timeOnSite >= 69.5
      | bouncePerc < 35.1
      | | timeOnSite < 98.5 : no
      | | timeOnSite >= 98.5 : yes
      | | bouncePerc >= 35.1
      | | | trafficRank < 1253069.5 : no
      | | | trafficRank >= 1253069.5
      | | | globalUsersPerc < 0 : no
      | | | globalUsersPerc >= 0 : yes
      | | | bouncePerc >= 36.8
      | | | | trafficRank < 2164361.5
      | | | | trafficRank < 1967745.5
      | | | | | bouncePerc < 40.25 : yes
      | | | | | bouncePerc >= 40.25
        | | | | uniquePageViewDaily < 1.35
        | | | | | bouncePerc < 58.65 : yes
        | | | | | bouncePerc >= 58.65 : no
        | | | | | uniquePageViewDaily >= 1.35
        | | | | | | uniquePageViewDaily < 2.2 : yes
        | | | | | | uniquePageViewDaily >= 2.2
        | | | | | | | bouncePerc < 41 : no
        | | | | | | | bouncePerc >= 41
        | | | | | | | | uniquePageViewDaily < 2.85
        | | | | | | | | timeOnSite < 109.5 : no
        | | | | | | | | timeOnSite >= 109.5
        | | | | | | | | | bouncePerc < 51.8 : yes
        | | | | | | | | | bouncePerc >= 51.8
        | | | | | | | | | | pageviewsPerc < 0 : yes
        | | | | | | | | | | pageviewsPerc >= 0
        | | | | | | | | | | uniquePageViewDaily < 2.5 : yes
        | | | | | | | | | | uniquePageViewDaily >= 2.5 : no
        | | | | | | | | | | | trafficRank >= 1967745.5 : no
        | | | | | | | | | | | trafficRank >= 2164361.5 : yes

Table 3. Fleiss’ Kappa results first 10 sites

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>( P_i )</th>
</tr>
</thead>
<tbody>
<tr>
<td>getcnajobs.com</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>0.33</td>
</tr>
<tr>
<td>ginnel.com</td>
<td>2</td>
<td>1</td>
<td>0.33</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>0.33</td>
</tr>
<tr>
<td>crushproof-hose.com</td>
<td>1</td>
<td>2</td>
<td>0.33</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>0.33</td>
</tr>
<tr>
<td>kettlebell-training.com</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td>0.33</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>0.33</td>
</tr>
<tr>
<td>beasleydeals.com</td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>affordcustom.com</td>
<td>2</td>
<td>1</td>
<td>0.33</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>0.33</td>
</tr>
<tr>
<td>android-sale.com</td>
<td>2</td>
<td>1</td>
<td>0.33</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>0.33</td>
</tr>
<tr>
<td>katkincaid.com</td>
<td>1</td>
<td>1</td>
<td>0.33</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>0.33</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>0.33</td>
</tr>
<tr>
<td>berrytots.com</td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>searchengineacademywest.com</td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4. Fleiss’ Kappa results second 10 sites

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>( P_i )</th>
</tr>
</thead>
<tbody>
<tr>
<td>nyhealth.gov</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>0.33</td>
</tr>
<tr>
<td>usherbrooke.ca</td>
<td>2</td>
<td>1</td>
<td>0.33</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>0.33</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>0.33</td>
</tr>
<tr>
<td>geekphilosopher.com</td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
<td>0.33</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>inetcheck.net</td>
<td>1</td>
<td>2</td>
<td>0.33</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>0.33</td>
</tr>
<tr>
<td>spannews.com</td>
<td>1</td>
<td>1</td>
<td>0.33</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>0.33</td>
</tr>
<tr>
<td>mernards.com</td>
<td>2</td>
<td>1</td>
<td>0.33</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>0.33</td>
</tr>
<tr>
<td>emerchantview.com</td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>newhopemeadowview.com</td>
<td>2</td>
<td>1</td>
<td>0.33</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td>0.33</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>0.33</td>
</tr>
<tr>
<td>beingjrridinger.com</td>
<td>1</td>
<td>2</td>
<td>0.33</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td>0.33</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>2</td>
<td>0.33</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>autumnsedgeretreat.com</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td>0.33</td>
</tr>
</tbody>
</table>