Log Analysis and Document Classification Toolkit (First Version)

Deliverable number
D1.3

Dissemination level
Public

Delivery date
2 November 2015

Status
Final

Author(s)
Ljiljana Dolamic, Sokratis Metallidis, João Palotti, Célia Boyer, Allan Hanbury

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 644753 (KConnect)
Executive Summary

This deliverable reports on the progress made by HON and TUW for the creation of the trustability and readability APIs, as well as the visualization tool for the query logs. Both readability and trustability APIs developed by HON can be found at: http://ec2-52-17-209-151.eu-west-1.compute.amazonaws.com/~kconnect/cgi-bin/.

Trustability estimation is important when dealing with medical information from the Internet. The purpose of the HONcode is to guide internet users and patients towards trustworthy medical information by certifying health-related websites that offer content that respects the HONcode criteria.

Understanding medical information can be challenging, especially for people without medical experience or education. A difficult to read document can lead to misunderstanding and potentially damaging actions. With the readability and classification tool provided by HON, the end user can access a public health related document and adapt the content to his/her own health literacy level. This tool is exposed to the API mentioned above. The algorithm provided by this API can classify the content of a document in three different levels, namely easy, average and difficult.

TUW has created a log analysis and visualization tool. The tool currently analyses search logs provided by the TRIP Database. Currently the data is annotated using GATE annotation tool with the Khresmoi knowledge database. Furthermore by using the Elastic Logstash Kibana (ELK) stack we are able to visualize these search logs. The end result is better understanding of data by the search engine provider, with the potential to use classification to adapt search results to the end users based on their interactions with the system. Access to the visualization tool is available upon request to Allan Hanbury (allan.hanbury@tuwien.ac.at).
Table of Contents

1 Introduction ............................................................................................................................................ 4

2 Trustability and Readability Classification .......................................................................................... 4
   2.1 Readability ....................................................................................................................................... 4
       2.1.1 Readability API ...................................................................................................................... 5
   2.2 Trustability ...................................................................................................................................... 6
       2.2.1 Date detection ......................................................................................................................... 6
       2.2.2 Three Trustability APIs ......................................................................................................... 8

3 Log Analysis ......................................................................................................................................... 12
   3.1 TRIP Database Field and Annotations .......................................................................................... 12
   3.2 Query Visualisation ......................................................................................................................... 13
       3.2.1 Line Chart ............................................................................................................................... 13
       3.2.2 Pie Chart .................................................................................................................................. 14
       3.2.3 Bar Chart .................................................................................................................................. 15

4 Conclusion ........................................................................................................................................... 15

5 References ............................................................................................................................................ 16

List of Abbreviations

ELK     Elastic Logstash Kibana
API     Application Programming Interface
K4E     Khresmoi for Everyone
HON     Health on the Net
HONcode HON Code of Conduct for medical and health Web sites
HONsearch HON search engine
TUW     Vienna University of Technology
NER     Named Entity Recognition
1 Introduction

This document has the purpose of describing the advances made in making the services developed by HON and TUW available. The first part of this report describes the readability and trustability estimation Application Programming Interfaces (APIs) developed by HON and some basic examples. The second part presents the first steps of the search log analysis tool. The search logs are annotated with the concepts found in the Khresmoi Knowledge Base using the Khresmoi GATE annotation pipeline. Examples of visualisations obtained from analysing the TRIP Database search logs are presented.

2 Trustability and Readability Classification

This section describe the APIs and methods for estimating the trustability and readability of medical documents. Such services are particularly important when searching for medical information on the internet, which contains medical information at a wide range of readability levels, and ranging in trustability levels from completely trustable information provided by a professional medical institution to invented information with no scientific evidence.

2.1 Readability

A difficult-to-read text could lead people to give up reading it or even to misunderstand the content. The goal of determining the readability level of the public health related document is to enable the end user the choice of the reading material adapted to his/her own health literacy level. In this direction, HON has developed the readability classification tool. This algorithm, based on machine learning for French and a lexical approach for English is described in detail in [1]. It classifies the content of the document according to three different levels, namely easy, average and difficult. This tool is already implemented into the Khresmoi for Everyone (K4E) (http://everyone.khresmoi.eu/hon-search/) search engine. Figure 1 illustrates this implementation. The readability level of the webpage is indicated in the form of a pictogram next to the result returned by the search engine.

![Fig. 1. K4E Readability shown to the left of each result](http://example.com/k4e_readability.png)
2.1.1 Readability API

The readability API was developed by HON in order to simplify the integration of the above-described tool into vertical search solutions. Figure 2 illustrates the API workflow.

This API has one parameter, data, and accepts two types of input, namely text or URL. The data parameter value has to be in JSON format as illustrated below:

- `{“text”:”text to be estimated”}`
- `{“url”:”url to be extracted and estimated”}`

In the case of the URL being passed as the parameter, the API first crawls and extracts the content of the webpage. The webpage text content undergoes automatic language detection. If the detected language is amongst the supported languages, for the moment English and French, the readability level is estimated, and returned to user in the JSON format. If the detected language is not supported, the appropriate error message is returned.

This API is available from the following address:

http://ec2-52-17-209-151.eu-west-1.compute.amazonaws.com/~kconnect/cgi-bin/readability.cgi

![Figure 2. Readability API](image-url)
2.2 Trustability

The Health on the Net Foundation established the HON Code of Conduct in 1996 [5] with a consensus of health information editors in order to have common criteria of good practices for on-line health information. The aim of the HONcode is to guide internet users and patients towards trustworthy medical information by certifying that health-related websites offer content that respects the HONcode criteria. The HONcode certification process is a voluntary process, which is performed upon request. Thus, a website can be completely reliable and respect the HONcode criteria without being certified. HON is pursuing research on how the ethical principles within a health website can be identified automatically. The automated system for the detection of HONcode, illustrated in Figure 3, is described in detail in [2]. This system consists of 9 distinct classifiers based on a machine-learning framework [3] for each of the HONcode criteria. The Attribution criterion is divided into two distinct parts: namely Date Attribution and References. The excerpts extracted by HONcode experts as a justification of the website’s compliance to a given criteria are used as a learning/test collection in this system. After extensive testing of this system using the test collection, the next logical step was to verify its performance on real life examples, thus comparing it to the current manual HONcode certification process. This process illustrated in Figure 4 was described in [4].

The comparison performed in the scope of this research has revealed two main problems of the developed trustability detection algorithm. Concerning the Date Attribution, the system was unable to detect this information if it was displayed through numbers only without any accompanying text, e.g. 24/08/2015. Keeping the number in the tokenization process would not be a solution, as it would require that all dates be listed in the training set. Forcing the feature selection in this case might result in serious system over-fitting as all the dates could be wrongly recognized as information linked to the last update. Another problem we identified was that the content relevant to the Complementarity criteria was not detected because it was submerged by the large amount of information concerning other HONcode criteria, present on the page parsed by the classifier. This issue was acknowledged as the main reason for the low recall of the Complementarity criterion.

2.2.1 Date detection

In the light of the specificity of the given vocabulary for the Date Attribution criterion, we opted to replace the machine learning classifier for this criterion by the Named Entity Recognition (NER) tool from the OpenNLP toolkit [6]. The previously mentioned corpus of 2794 excerpts was used to train the NER model (Table 1) so it would correspond to the required specificity of the date detection.

<table>
<thead>
<tr>
<th>NER model training</th>
</tr>
</thead>
</table>
| This notice of privacy practices is `<START:date>` effective February 1, 2010 `<END>`.
| `<START:date>` revised 5/11/06 12/21/01 `<END>` |
| This information was `<START:date>` last updated on july 17, 2007 `<END>` |
| This page was `<START:date>` last modified on friday may 22, 2009 `<END>` 02:13pm |
| Content last updated dynamically at `<START:date>` last updated sun, 28 nov 2010 `<END>` 13:04:08 -0600 |
| Date of first authorisation/renewal of the authorisation `<START:date>` 12th april 2003 `<END>` |

Table 1. Date NER training model
Figure 3. Automated HONcode detection
2.2.2 Three Trustability APIs

Three APIs were developed by HON to simplify the integration of the trustability tool described above into the vertical search solutions. Figure 5 illustrates the flow chart of these three APIs. The APIs “Is_trustable” and “Trustability” are directly related to the trustability classification algorithm. On the other hand, the “Similar_pages” API does not use this algorithm. Its link to the trustability lies in the type of the results returned, e.g. pages from trustable sources.
2.2.2.1 Trustability API

This API available from:
http://ec2-52-17-209-151.eu-west-1.compute.amazonaws.com/~kconnect/cgi-bin/trustability.cgi

enables direct estimation of the HONcode criteria justification present in the submitted text or within submitted url.

This API takes one parameter “data” with the values of text or URL being passed in the JSON format as follows:

- {“url”:”https:…”}
- {“text”: “text to be analyzed”}

The trustability criteria detection is available in following languages: English, French, Spanish, German, Italian and Dutch. The language of the text submitted for analysis or of the extracted content of the submitted webpage is automatically detected. If the language is not in the in the supported language set the appropriate error message is returned.

This API returns results in JSON format that contains the following information:

- trustability – list of detected criteria with the score (between 0 and 1) representing the system confidence in the result. For the Date attribution, on the other hand, the value represents the detected date.
- Language – the language detected by the system
- Text – the text on which the criteria detection is performed. This represents either the text submitted by the user or the text extracted by the system from the webpage in case of the url being passed as the API’s parameter.
- url – this optional value is returned only in the case of the URL being passed as the parameter.
Figure 6 gives an illustration of the result returned by this API in case of the text being passed as the parameter.

```json
{
   "trustability" : {
      "Advertising policy" : 0.999007228844789
   },

   "language" : "en",

   "text" : "Advertising policy this website may display advertise constitute an endorsement, guarantee, warranty, or recommendation labeling. links to other websites this website may contain by the authors of this website or an indication of any affiliate not responsible for the content of linked third party website"
}
```

Figure 6. Trustability API result (text parameter)

### 2.2.2.2 Is_trustable API

This API is available from the following address:

http://ec2-52-17-209-151.eu-west-1.compute.amazonaws.com/~kconnect/cgi-bin/is-trustable.cgi

It takes as the parameter a web domain (e.g. webmd.com), verifies whether any of the pages from the given domain are indexed and what is the overall status of the automated detection based on those pages.

Within the K4E search engine pipeline each page crawled and indexed of each domain passes through the automated HONcode detection tool. The information about the detected criteria and their scores is then stored into the NoSQL database, in addition to this information being indexed into the search engine. The “Is_trustable” API uses the information stored in this database for the whole body of pages from the given domain, to present the list of automatically detected criteria. Figure 7 illustrates the results of this API for the domain “webmd.com”.

```json
{
   "domain" : "webmd.com",
   "trustability" : {
      "score" : 88,
      "principles" : {
         "Advertising policy",
         "Attribution",
         "Authoritative",
         "Data",
         "Financial disclosure",
         "Justifiability",
         "Privacy",
         "Transparency"
      }
   }
}
```

Figure 7. Is_trustable API results for webmd.com domain

It can be noticed that for this domain the system was unable to detect the Complementarity criterion on any of the crawled pages (e.g. score = 88%).

The main restriction of this API was the presence of the pages from the requested domain in the database. It is important to mention that the K4E search system contains pages from certified or websites estimated
to be trustworthy even though they are not HONcode certified. This largely limits the coverage of this API. Two possible solutions appeared during the API development. Broaden the coverage by adding additional domains to the system, manually or by users’ request or create a separate database, which would contain the pages from the domains not already present in the K4E database. We have opted for the second solution, creating an additional database. This enables us to keep the content of the search engine controlled, while being able to give the trustability estimation for a wider range of domains. The API pulls the information from both databases in order to give the estimation for the requested domain, and only in the case of the domain not being present in either of them, is the error message returned. However, the new content is added to the newly created database that should additionally broaden this API’s coverage.

2.2.2.3 Similar_pages API

The Similar_pages API is available from the following address:

http://ec2-52-17-209-151.eu-west-1.compute.amazonaws.com/~kconnect/cgi-bin/similar-pages.cgi

For a URL passed as a parameter, this API returns the list of pages found in the search engine index that cover the same thematic material as the parameter URL.
3 Log Analysis

TUW is developing a search log analysis and classification service, which will allow medical search engine providers to get insight into what their users are querying about and how they are querying. It will also allow the search engine providers to classify their individual users into categories (e.g. medical professional or layperson) based on their querying style and type of vocabulary used in their queries, using techniques recently published in [7].

In this first prototype, the search log analysis is done with the search logs from TRIP Database. By looking deeper into these search logs it is possible to understand better the queries of the end user and improve the search results.

3.1 TRIP Database Field and Annotations

The source of analysed logs is the TRIP Database. The logs represent queries of people searching for medical information. These queries were annotated using GATE and the Khresmoi medical annotation pipeline. Search logs are indexed, analysed and visualized further using the Elastic Logstash Kibana (ELK) stack.

- User queries inside these logs without annotation look like the following example:


- Fields analysed gives us the following information:
  1. Hashcode of the session
  2. Date the query occurred
  3. Query
  4. User Number
  5. Webpage chosen by the user
  6. Title of the webpage the user visited

- The fields that were annotated are the query field and the title field. Therefore two additional fields were added to the previous information which are the actual annotations. The previous query is changed as follows:

  "0002kcz0fphmx355f24y1t55","2014-10-28 19:56:28.050","(chiropractic or mobilization) AND osteoarthritis AND pain","4862030","http://www.ncbi.nlm.nih.gov/pubmed/22923762","A systematic review of evidence for the effectiveness of practitioner-based complementary and alternative therapies in the management of rheumatic diseases: osteoarthritis.","osteoarthritis|Disease;C1970083;T033$pain|Disease;C2675800;T033$","therapies|Disease;C1363945;T033$osteoarthritis|Disease;C1970083;T033$srheumatic diseases|Disease;C0035435;T047$"

The addition of the concept information allows the query logs to be analysed by the medical concept queried instead of by the word queried.
3.2 Query Visualisation

In this section, we show three examples of visualisations of information extracted from the TRIP Database query log. All of the visualizations shown can be found and processed further by using the ELK stack demo available on our server. For access to this demo, please contact Allan Hanbury (allan.hanbury@tuwien.ac.at).

3.2.1 Line Chart

Figure 8 shows how many times a query was issued containing the keyword “influenza” in the time period covered by the TRIP Database search logs. The interval of the count of the occurrences is monthly. The decreasing slope at the beginning could correspond to the tail end of the H7N9 avian influenza scare that started in 2013. The second peak is likely the annual increase in searching for reports on the type of influenza that is like to occur in an upcoming winter in the northern hemisphere.

![Influenza search occurrences line chart](image)
3.2.2 Pie Chart

This example illustrates the use of the annotation classes in the analysis. Each query and document title is annotated with types that indicate whether it refers to drug, disease, investigation or anatomy, where Annotation1 contains the annotation from the query field and Annotation2 contains the annotation of the title field.

Figure 9 is a pie chart that illustrates an analysis on how often the query and document title contain the same annotation classes. The inner pie chart analyses the Annotation1 field and shows the occurrences of each type of annotation found in the query (Drug, Disease, Anatomy, Investigation). The outer level further analyses the queries in the inner pie chart – for all queries in each inner slice, it presents the distribution of each annotation type in Annotation2 (the title of the webpage the user has clicked on after issuing the query in Annotation1).

In this double pie chart, the inner level shows that the majority of queries have the annotation class of “disease”. It can be seen that the largest segment in the outer level always corresponds to the annotation class of the query in the inner level. However, for the three non-disease query classes, documents with a title having annotation class “disease” are always in second position in terms of the proportion of times they are clicked.

![Annotation analysis](image)

Figure 9. Annotation analysis using a pie chart
3.2.3 Bar Chart

Figure 10 shows a bar chart illustrating the number of visits to the top five most-visited URLs per month. It shows that the top five most-visited are stable over time. As can be expected, the most visited website is always the National Library of Medicine (NLM) National Center for Biotechnology Information (NCBI). The second most visited site is always the Centre for Reviews and Dissemination at the University of York. The frequent accesses to dx.doi.org indicates the method by which the search engine retrieves some scientific articles. The final two most-visited sites are a collection of information for physicians and health professionals (Medscape), and the National Guideline Clearinghouse.

Figure 10. URL bar chart depicting the top 5 websites visited per month

4 Conclusion

Among the requirements from KConnect deliverable D3.1, the API for readability and trustability are implemented and presented in this report. Following in the log analysis section, we annotate the search logs from the TRIP Database using the GATE annotation tool and the Khresmoi Knowledge Base. The data from search logs can be useful for the classification of users, allowing them to receive better targeted search results – implementing such a user classification is the next step in the development of the search log analysis tool. We also provide the insights of our first version of the visualization tool for the analysis of annotated search logs from medical search engines, based on ELK stack framework. An API for accessing the search log analysis functions and visualizations is planned for the next release.
5 References


