TitleFinder: Extracting the Headline of News Web Pages based on Cosine Similarity and Overlap Scoring Similarity

Hadi Mohammadzadeh
Institute of Applied Information Processing
University of Ulm
D-89081 Ulm, Germany
hadi.mohammadzadeh@uni-ulm.de

Thomas Gottron
Institute for Web Science and Technologies
Universität Koblenz-Landau
D-56070 Koblenz, Germany
gottron@uni-koblenz.de

Franz Schweiggert
Institute of Applied Information Processing
University of Ulm
D-89081 Ulm, Germany
franz.schweiggert@uni-ulm.de

Gerhard Heyer
Institut für Informatik
Universität Leipzig
D-04009 Leipzig, Germany
heyer@informatik.uni-leipzig.de

ABSTRACT
Automatically extracting the headline of online web articles has many applications in web mining and information retrieval. In this paper, we developed a content-based and domain- and language-independent approach, TitleFinder, for unsupervised extraction of the headline of web articles. TitleFinder starts by using a heuristic to select a candidate headline. In a second step the contents of each text fragment in the HTML file are compared to the candidate headline. We implemented four types of similarity for this comparison: two variations of the cosine similarity based on tf and tf-idf weighting schemata, an overlap scoring similarity and an aggregated metric combining the scores of the previous three similarities. Our method achieves high performance in terms of effectiveness and efficiency and outperforms approaches operating on structural and visual features on a test set consisting of 11,218 news web pages from 15 different domains.

Categories and Subject Descriptors
H.3.1 [Information Storage and Retrieval]: Content Analysis and Indexing; H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval; D.2.8 [Software Engineering]: Metrics—complexity measures, performance measures

General Terms
Algorithms, Experimentations, Performance

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

WIDM ’12, November 2, 2012, Maui, Hawaii, USA.
Copyright 2012 ACM 978-1-4503-1720-7/12/11 ...$15.00.

Keywords
Vector Space Model, Cosine Similarity, Overlap Scoring Similarity, HTML Web Pages, Headline Extraction, Title Extraction, Information Retrieval

1. INTRODUCTION
A huge volume of online articles is published on the web every day. From political, sports, scientific or cultural news to personal blogs, web fora entries or user discussions. These articles are published as HTML documents on websites and weblogs and contribute a significant portion of the text-based information on the web. Thereby, they form a large share of the content on the web as HTML is still a predominant format on the web.

These online articles form the basis of various types of text based analytics. Information extraction, topic detection and tracking, sentiment detection and opinion mining are just some of the text mining applications leveraging data on the web.

But, while the sheer volume of the data makes it an interesting resource for text analytics, the format of the data poses a challenge. On the one hand, HTML documents are designed and interpreted in a presentation oriented fashion. Therefore, they imply at best a weak semantic of their content, which moreover is often neglected by web authors. Given, the human audience for HTML documents, web authors often do not see any need to use HTML tags according to their semantic intention, as long as the visual appeal of the documents is satisfactory. On the other hand, websites are complex structures of content and functionality. Thus, a document contains far more than the main article: navigation elements, interaction elements, advertisements and references to other information sources.

For this reason, approaches are being sought, that derive the semantics of the text fragments of an online document or that attempt to identify the main content. Main content identification or extraction has been in the focus of various recent research works [17, 12, 7, 14, 6, 5, 16]. The algorithms
proposed in these papers typically ignore the semantics of the HTML annotations and operate on features like tag density, character encoding, tree structures or text length. Accordingly also the extracted articles are lacking semantics about their structure: which part constitutes the headline (or title), the abstract (or teaser text), the main text body or provides examples, etc. Especially the headline of an article, needs to be highlighted here as on one hand it is important for tasks such as text summarization or topic detection and on the other hand many main content detection algorithms have a weak extraction performance regarding the headline, i.e. they erroneously miss the headline when extracting the article from a web document [7].

Motivated by these observations, we address in this paper the problem of identifying the headline of a web article. Little work has been done in this domain. Previous approaches mainly focus on structural and visual features of the headline [9, 18, 3, 4]. In this paper we propose a novel method based on the content of an article: TitleFinder. TitleFinder starts by using a heuristic to identify a candidate headline. This candidate headline is then used to query the web document for a text fragment that best describes the article. We implemented this method by using four different text similarity metrics. Given the design of the algorithm it has two main advantages over previous approaches. First, it is an unsupervised approach that does not require training data. Second, it is capable to operate on single documents and is independent of a template-driven layout of the documents.

In our evaluation the method shows a very accurate (F1-independent of a template-driven layout of the documents. Second, it is capable to operate on single documents and is an unsupervised approach that does not require training data. Especially the headline of an article, needs to be highlighted here as on one hand it is important for tasks such as text summarization or topic detection and on the other hand many main content detection algorithms have a weak extraction performance regarding the headline, i.e. they erroneously miss the headline when extracting the article from a web document [7].

In our evaluation the method shows a very accurate (F1-value of 0.969) identification of the headlines of 11,218 web documents and outperforms two baseline methods using structural and visual features.

The rest of the paper is organized as follows. In Section 2, we review related work and provide a survey and brief description of headline extraction algorithms applicable to HTML web documents. We proceed by explaining and analyzing the problem setting of our work. In Section 4, we develop the TitleFinder method. The data sets, the evaluation methodology, and the results of our approach are presented in 5. Finally, in Section 6 we discuss our conclusion and give some suggestions for future work.

2. RELATED WORK

Hu et al. [9] have claimed in 2005 that no special research has exactly concentrated on the problem of headline extraction from HTML files prior to their work. They have also stated that extraction of the headline from the body of HTML files is not that simple, since various web pages contain different contents and formats. The method proposed by Hu et al., supervised machine learning approach, has been used to identify and extract the headline. This method benefits from two main phases like many other machine learning methods, namely training phase and extraction phase. The experimental data will be prepared before first phase and during pre-processing. First, each web document is transformed into several text segments such that each text segment exactly corresponds to a line containing text in the web document. Afterwards, the text segments will be initialized based on specifications such as font size, font weight and position, which are very conspicuous for the headline. Meanwhile, headline is annotated among text segments of each web document. Now, a classification model is trained in the first phase according to the training data.

This model will determine whether a text segment is a headline or not. However, they have used the classification model as perceptron with uneven margins during extraction phase. In the second phase of algorithm, extraction phase, text segments of a web document enter the model, while the model allocates a score to every unit. Finally, text segments having the greatest score will be taken as the headline.

Hu et al. [8] in their research on automatic headline extraction used other models of machine learning, such as maximum entropy, maximum entropy marker model and voted perceptron. At last, they concluded that performance of the perceptron model was the best. Furthermore, the authors of this paper have also assessed the models developed on various domains and in different languages. They finally discovered that the accuracy is not decreased significantly.

Xue et al. [18] have employed supervised machine learning methods to extract the headline from web documents. The methods employed covered SVMs and Conditional Random Fields (CRF). As features in the machine learning models, formatting information and linguistic information have been utilized in this contribution. It is interesting to note that the CRF model is shown to be more successful than SVM in extraction of the headline.

Ibrahim et al. [10] developed an algorithm for automatic extraction of the headline and the main content of news web pages. For doing this, they proposed a supervised machine learning classification technique based on the use of a SVM classifier to extract the desired textual elements. The SVM classifier is trained strictly on structural features to identify the main content and its headline.

Chang et al. [1] proposed an automatic method for extracting the headline of HTML web documents based on supervised machine learning technique such as Decision Trees and Random Forests. The key point in their contribution is that they employ information in the header of the HTML file in order to obtain labelled training data for title extraction with limited human effort.

Zhang et al. [19] proposed a content-based and domain-independent method for extracting headlines from Chinese research papers using support vector machine classifier. They claimed their contribution achieved better results than rule based methods and attribute this to using apriori information about the headline’s words and the relation between headline and the body of the HTML web page.

All the approaches presented so far employ supervised machine learning techniques. Therefore they depend on training data from which they derive typical characteristics of the headline. Given the large variety of web document designs on the web, it is a challenging task to provide an un-biased training set in this setting. A similar problem has already been observed in the related field of main content extraction [2].

An alternative approach is to formalize human experience and domain knowledge into heuristics for solving the task of headline detection. Fan et al. [3, 4] have found that the headline is often annotated by special HTML tag (H1-H6) and given visual prominence. In this regard, they proposed a two-stage algorithm for characterizing the headline. At the first step, headline candidates are selected based on two following criteria: The first criterion states that just those candidates (lines containing text) can be selected as the headline whose horizontal starting position is not greater than that of main text region. Meanwhile, their top position must not
Figure 1: An example of Web pages with the selected headline.

be less than top quarter of the main text region. The second criteria imply that font size of candidates must not be smaller than that of the main article text. In the second step, a score is calculated for each candidate based on some rules including font size and position of the candidate. The candidate with the highest score will be selected as headline.

3. PROBLEM SETTING AND ANALYSIS

We already motivated the need for identifying and characterizing the headline in HTML web documents. To motivate a content based approach (rather than the structural and visual based approaches of related work), we now look at the problem setting in a more detailed way. Generally speaking, the structure of a web article is comprised of some text paragraphs, some figures or diagrams with their relevant captions, and of course a headline. Besides these main components of news, some additional items come along with the article, which are not related to the main content at all. Moreover, there are some elements which are not accounted for the main article but are somehow related to it. For example the news links located besides the main news directing the user to other pages.

We use the web document in Figure 1 to elucidate this issue in a bit more detail. The following parts can be identified in this article taken from the BBC News website:

- \( \alpha \) : Header and main menus of this news page from BBC website can be seen in this section which are known as additional items.

- \( \beta \) : This section includes main parts of the news, i.e. the headline, published date, constituent paragraphs, subtitles, and semi-related useful news \( \beta' \) as well as prominent news tips \( \beta'' \).

- \( \gamma \) : This section involves additional news items which are not closely related to the main news. The following sections are some of them:
  - top stories
  - features and analysis
  - most popular

It is obvious that most of the subsections contain a series of links which direct users to other pages for getting access to the text of news.

- \( \delta \) : The browser window’s menu bar indicates meta-information of the web document. It shows the text fragment which is encoded in HTML using the content of the title element \(<title>...</title>\). This fragment usually provides a good hint towards the headline, but typically contains other text as well. In this case, the website’s name is provided as well, other settings add a date or a copyright remark.

4. TITLEFINDER

We now develop our novel and content based method TitleFinder for identifying the headline of web articles. The algorithm proposed in this paper, benefits from one of the observations mentioned above: A great proportion of the word tokens inside the headline are similar to the title element. However the text content between the HTML tags \(<title>...</title>\) usually contains additional information, e.g. the name of the website hosting the article, the current date or a copyright notice.

This observation is utilized in TitleFinder to identify the headline of web articles. It interprets the text in the \(<title>\) element as a query to be applied on the text fragments of the article’s content. This allows for the identification of the most descriptive text fragment for the article. By considering all text nodes in the DOM tree as potential candidates for this text fragment and breaking them down into sentences, TitleFinder is independent of structural or visual information.

In more detail, the process behind TitleFinder passes through the following phases:

4.1 Pre-processing

First, since we do not make use of it, JavaScript code, CSS code, comments and meta tags are removed from the web document. Furthermore, we normalize the source code [13] for line breaks or excessive white space, which allows for easier identification of text fragments and word tokenization in the downstream process. For the same purpose, also other characters like single quote, double quote, comma and colon are deleted from the content of an HTML file.

4.2 Conversion of Text Fragments into Vector Space Representation

Our content based headline identification method is making use of a vector space representation [15] of text fragments. To arrive at this representation we take several steps. After the pre-processing, we extract the text fragments from the source code by considering the text nodes
of the DOM tree. For simplicity, we will use the expression sentence to distinguish a text fragment from the text of the complete web document. Assuming that $N$ text fragments of the HTML file denote its content, there would be $N$ sentences corresponding to the web document. We tokenize each sentence into words and transform it into a classical vector representation $\overrightarrow{s}_j$:

$$\overrightarrow{s}_j = \{w_{1,j}, w_{2,j}, ..., w_{|V|,j}\} \quad (1)$$

The entries $w_{i,j}$ in this vectors correspond to the weight of term $t_i \in V$ in sentence $j$, where $V = \{t_1, t_2, ..., t_{|V|}\}$ is the set of distinctive words of the content of HTML document. In our experiments we considered two very well established weighting schemes, namely term frequency (tf) weights and tf-idf weights.

### 4.3 Similarity Metrics

In order to identify the headline in a web document, we first obtain the text in the title element of an HTML file. Now, we consider this text as a query $q$. Then, the similarity between the query $q$ and each one of the $N$ sentences is assessed. Based on our observation, the hypothesis behind TitleFinder is, that the sentence which bears the highest similarity with query $q$ will be the headline. In the vector space framework of TitleFinder, we implemented the four following methods to measure the similarity between query $q$ and each of the $N$ sentences:

- **TF**: Cosine Similarity based on tf weighing Scheme
- **TF-IDF**: Cosine Similarity based on tf-idf weighing Scheme
- **OSM**: Overlap Scoring Measure
- **Aggregate** = TF + TF-IDF + OSM

The cosine similarity is well known in Information Retrieval. It corresponds to measuring the cosine of the angle $\theta$ between the query vector $\overrightarrow{q}$ and one of the $\overrightarrow{s}_j$ vectors ($1 \leq j \leq N$) with its value being in the range between 0 and 1. Higher values indicate a higher similarity of the vectors, which is interpreted as a high semantic similarity of the respective text fragments.

OSM, the third similarity metric [11], is a less commonly used similarity metric. It incorporates tf-idf weights for the query terms and tf weighting with Euclidean normalization for the sentences, which is shown by $w_{i,s}$. The OSM similarity is provided in the following Formula 2:

$$OSM(q, s_j) = \sum_{1 \leq i \leq q} w_{i,q} \times w_{i,s_j} \quad (2)$$

The fourth and last method aggregates the values of the previous three metrics and is shown in Formula 4. It combines values obtained from the three previous techniques for each of $N$ sentences to identify the headline. Thus, it show similarities to techniques for result list merging.

$$Aggregate(q, s_j) = \cos(q, s_j) + \cos_{t-idf}(q, s_j) + OSM(q, s_j) \quad (3)$$

$$Aggregate(q, s_j) = \frac{A}{A + C} \quad Precision = \frac{A}{A + B} \quad (4)$$

$$F1 - measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (5)$$

### 5. EXPERIMENTS

#### 5.1 Data sets

For evaluation purposes we use the datasets introduced in [7, 5] (cf. Table 1) and [12] (cf. Table 2), composed of 8,936 and 2,282, respectively, web pages from different websites. The first data set has been established for the evaluation of main content extraction algorithms on Western language documents and the second data set is a collection from web documents in Arabic, Farsi, Pashto, and Urdu.

### Table 1: Evaluation corpus of 8,936 web pages.

<table>
<thead>
<tr>
<th>Website</th>
<th>URL</th>
<th>Size</th>
<th>Lang.</th>
</tr>
</thead>
<tbody>
<tr>
<td>BBC</td>
<td><a href="http://www.bbc.co.uk/">www.bbc.co.uk/</a></td>
<td>900</td>
<td>English</td>
</tr>
<tr>
<td>Economist</td>
<td><a href="http://www.economist.com/">www.economist.com/</a></td>
<td>250</td>
<td>English</td>
</tr>
<tr>
<td>Golem</td>
<td>golem.de/</td>
<td>1,000</td>
<td>German</td>
</tr>
<tr>
<td>Heise</td>
<td><a href="http://www.heise.de/">www.heise.de/</a></td>
<td>1,000</td>
<td>German</td>
</tr>
<tr>
<td>Repubblica</td>
<td><a href="http://www.repubblica.it/">www.repubblica.it/</a></td>
<td>1,000</td>
<td>Italian</td>
</tr>
<tr>
<td>Slashdot</td>
<td>slashdot.org/</td>
<td>394</td>
<td>English</td>
</tr>
<tr>
<td>Spiegel</td>
<td><a href="http://www.spiegel.de/">www.spiegel.de/</a></td>
<td>1,000</td>
<td>German</td>
</tr>
<tr>
<td>Telepolis</td>
<td><a href="http://www.telepolis.de/">www.telepolis.de/</a></td>
<td>1,000</td>
<td>German</td>
</tr>
<tr>
<td>Wiki</td>
<td>en.wikipedia.org/</td>
<td>1,000</td>
<td>English</td>
</tr>
<tr>
<td>Yahoo</td>
<td>news.yahoo.com/</td>
<td>1,000</td>
<td>English</td>
</tr>
<tr>
<td>Zdnet</td>
<td><a href="http://www.zdnet.de/">www.zdnet.de/</a></td>
<td>422</td>
<td>German</td>
</tr>
</tbody>
</table>

### Table 2: Evaluation corpus of 2,282 Web pages.

<table>
<thead>
<tr>
<th>Website</th>
<th>URL</th>
<th>Size</th>
<th>Lang.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ahram</td>
<td><a href="http://www.ahramonline.org/">www.ahramonline.org/</a></td>
<td>188</td>
<td>Arabic</td>
</tr>
<tr>
<td>BBC</td>
<td><a href="http://www.bbc.co.uk/arabic/">www.bbc.co.uk/arabic/</a></td>
<td>492</td>
<td>Arabic</td>
</tr>
<tr>
<td>BBC</td>
<td><a href="http://www.bbc.co.uk/pashto/">www.bbc.co.uk/pashto/</a></td>
<td>368</td>
<td>Pashto</td>
</tr>
<tr>
<td>BBC</td>
<td><a href="http://www.bbc.co.uk/persian/">www.bbc.co.uk/persian/</a></td>
<td>598</td>
<td>Farsi</td>
</tr>
<tr>
<td>BBC</td>
<td><a href="http://www.bbc.co.uk/urdu/">www.bbc.co.uk/urdu/</a></td>
<td>213</td>
<td>Urdu</td>
</tr>
<tr>
<td>Hamshahr</td>
<td>hamshahrionline.ir/</td>
<td>355</td>
<td>Farsi</td>
</tr>
<tr>
<td>Jame Jam</td>
<td><a href="http://www.jamejamonline.ir/">www.jamejamonline.ir/</a></td>
<td>137</td>
<td>Farsi</td>
</tr>
<tr>
<td>Reuters</td>
<td>ara.reuters.com/</td>
<td>116</td>
<td>Arabic</td>
</tr>
<tr>
<td>Wiki</td>
<td>en.wikipedia.org/</td>
<td>35</td>
<td>Farsi</td>
</tr>
</tbody>
</table>

#### 5.2 Evaluation Methodology

In order to calculate the accuracy of any headline extraction method, it is necessary to provide a manually crafted gold standard for the headline of all HTML files. Both corpora mentioned in Section 5.1 provide such a gold standard. For the purpose of evaluation, the output of a headline extraction algorithm is checked for a precise match with the gold standard headline of the corresponding HTML document.

In our experiments, we follow other approaches and employ recall, precision and F1 as performance metrics [8]. The metrics are defined in Formula 4 and 5, where we define $A$ to be the number of headline correctly identified as headlines, $B$ as the number of other elements misclassified as headline and $C$ to be the number of headline erroneously non identified as such (cf. also Table 3).

$$Recall = \frac{A}{A + C} \quad Precision = \frac{A}{A + B} \quad (4)$$

$$F1 - measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (5)$$
Table 4: Evaluation results of data set in Tables 1 and 2 based on F1-measure and processing speed (KB/s).

<table>
<thead>
<tr>
<th>Website</th>
<th>TF</th>
<th>TF-IDF</th>
<th>TitleFinder</th>
<th>OSM</th>
<th>Aggr.</th>
<th>Largest Font</th>
<th>Baselines</th>
</tr>
</thead>
<tbody>
<tr>
<td>BBC</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Espn</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Hise</td>
<td>0.999</td>
<td>0.998</td>
<td>1.0</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Repubbica</td>
<td>0.999</td>
<td>0.999</td>
<td>0.999</td>
<td>0.999</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Slashdot</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Spiegel</td>
<td>0.977</td>
<td>0.974</td>
<td>0.976</td>
<td>0.976</td>
<td>0.976</td>
<td>0.976</td>
<td>0.976</td>
</tr>
<tr>
<td>Teletop</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Wiki</td>
<td>0.861</td>
<td>0.805</td>
<td>0.834</td>
<td>0.828</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Yahoo</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Zdf</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Abram</td>
<td>0.952</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>BBC Arabic</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>BBC Pashto</td>
<td>1.0</td>
<td>0.997</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>BBC Farsi</td>
<td>0.998</td>
<td>0.998</td>
<td>0.998</td>
<td>0.998</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>BBC Urdu</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Hamasahri</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Iame Jaine</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Reuter</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Wiki</td>
<td>1.0</td>
<td>0.706</td>
<td>0.599</td>
<td>0.735</td>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Macro-Avg.</td>
<td>0.989</td>
<td>0.975</td>
<td>0.969</td>
<td>0.978</td>
<td>0.6</td>
<td>0.1</td>
<td>0.1</td>
</tr>
</tbody>
</table>

Table 3: Contingence table with regard to headline extraction.

<table>
<thead>
<tr>
<th>Is headline</th>
<th>Is not headline</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>B</td>
</tr>
<tr>
<td>C</td>
<td>D</td>
</tr>
</tbody>
</table>

5.3 Baselines

As baseline approaches we consider the two methods. First of all, we evaluate the naive approach of assuming the content of the title element to be equivalent to the headline. As mentioned previously, extracting headline of the web document using this method is not reliable because there are some additional information in the title element of the web document not related to the headline.

Therefore, we employ a more sophisticated baseline. It follows the observation that the headline often corresponds to the first occurrence of the largest font size in the document [1.9]. Thus, we tested the accuracy of such a visual feature based baseline method. We implemented a method searching for the first encounter of the highest order headline element in web document.

5.4 Results

Table 4 presents the results of our experiments showing the average F1-measures and the processing speed (KB/s) of TitleFinder, the columns labelled TF, TF-IDF, OSM and Aggregate, and the baseline approaches, the columns labelled Largest Font and title, on our data sets. The last row in the table aggregates the values using a macro-average F1-value.

Regarding the baselines, it can be observed that they either always succeed or always fail on the documents taken from one website. This is not surprising as most modern websites are template driven and exhibit the same or at least a very similar structure. The naive approach of simply using the title element’s content fails in most cases (18 out of 20 websites).

Also the more sophisticated baseline of using the first occurrence of the largest font-size fails quite often (8 out of 20 websites). The two assumptions considered in it, namely “the headline has the largest font size” and “the headline is the first occurrence of the largest font size”, seem not to hold for extracting the headline from HTML documents. We observed that the headline quite often is not contained in the semantically correct elements h1 to h6, but rather in <div> and/or <span> tags. This technique is used, for instance, on the websites BBC, Jame Jam, and Abram. Furthermore, the documents from Slashdot, Spiegel, and ZDF websites have employed <h1> tag for displaying the website’s name which is not headline. An important point to bear in mind is, that the extraction of the headline assumes a certain consistency of the underlying document template. It always selects the same first largest headline element in a document. In Table 4, there are five entities, BBC, BBC Arabic, BBC Pashto, BBC Farsi, and BBC Urdu, employing the same website template. Looking for explanations of the failure of the baseline algorithm on the BBC Arabic web site we noticed that—unlike the other BBC websites—it did not encode the headline in the first <h1> tag. Hence, we can conclude trusting to website template for extracting the headline of a web page sometimes misleading us to incorrect results.
Considering the performance of TitleFinder, we can make the following observations in Table 4:

- In 14 out of the 20 considered websites, the TitleFinder algorithm shows an F1-measure value of 1.0 which is indicative of the high accuracy of this method. Furthermore, in 17 out of 20, our proposed approach demonstrates an F1-measure value greater than or equal to 0.997, which is a considerable accuracy.

- None of the four TitleFinder variations employing different similarity metrics has failed completely on our data sets as observed, for instance, for the two baseline approaches with a F1-measure of 0. This can be explained by the algorithm to be agnostic towards structural properties of the documents and therefore being more independent from template properties.

- The average F1-measure for all variants of the TitleFinder is no lower than 0.969. This demonstrates a general stability of the approach.

- Considering the Macro-Average F1-measure, it can be seen that although TitleFinder based on TF does not show F1-measure equal to 1.0 in all cases, but its reported Macro-Average F1-measure value, 0.989, is greater than Macro-Average F1-measure value of the other TitleFinder implementations. It seems that TitleFinder based on the TF weighting scheme can be an appropriate method for extraction of headlines from web pages, since its Macro-Average F1-measure value, 0.989, is relatively high, although the improvement of TF over the other TitleFinder methods is not statistically significant.

- For the websites Repubblica, Spiegel, Wiki, and BBC Farsi none of TitleFinder methods were able to find average F1-measure value of 1.0. This behaviour is mainly attributed to an error which occurs when the total number of headline tokens is smaller than that of tokens related to name and specifications of the website cited in the title element. In this case, the TitleFinder starts from a sub-optimal candidate headline to query the documents content elements and tends to extract name or specifications of the website instead of its headline.

When comparing the processing time of the approaches, TitleFinder is significantly slower than the structural approaches. This can be attributed to the overhead of analyzing all text elements and building their vector representations. In addition, identifying and extracting all content tokens of HTML document using an HTML parser are far more time-consuming. However, so far we did not tune our algorithm for fast processing and there is still potential for streamlining the programs written.

6. CONCLUSIONS AND FUTURE WORK

In this contribution we proposed TitleFinder, a content based method for headline extraction from web pages. The main idea is to use the content of the title element as a candidate headline and then compute the similarity between this candidate and all text fragments in the body of the HTML file. We implemented TitleFinder using four different similarity metrics. The results obtained from implementation of four variations of the algorithms on 11,218 web pages indicated a high accuracy of our method. In many cases we observed an perfect extraction-performance with an F1-value of 1.0.

In future work, we aim to overcome some minor weaknesses for particular settings observed in evaluation data. One issue is to improve the quality of the initial selection of the candidate headline. As one solution, name and specifications of a web page appearing in the title element as well as other tokens present in the title element can be searched in a main content. Since name and specifications related to a web page are not present in the main content related to a web page, it seems promising to remove tokens which are present in the title element but not in the main content of a web page. Also, we will combine our headline detection algorithms with content extraction algorithms to enhance accuracy of the content extraction methods.

7. REFERENCES


