Language and Age Filter Progress

1 Age Filter

One important heuristic used in ranking search results is the age of the webpages linked to in each result. This age can be calculated by subtracting the webpage's date of creation from the present date. The date of creation is often embedded in the webpage itself, in the form of a particular HTML field called a time tag. The time tag can be used to specify the date the webpage was last modified. A few examples of the tag's syntax and meaning is shown in figure 1 below:

```
Dates:
<time datetime="1914">  <!-- means the year 1914 -->
<time datetime="1914-12">  <!-- means December 1914 -->
<time datetime="1914-12-20">  <!-- means 20 December 1914 -->
<time datetime="12-20">  <!-- means 20 December any year -->
<time datetime="1914-W15">  <!-- means week 15 of year 1914 -->

Date and Times:
<time datetime="1914-12-20T08:00">  <!-- means 20 December 1914 at 8am -->
<time datetime="1914-12-20 08:00">  <!-- also means 20 December 1914 at 8am -->
<time datetime="1914-12-20 08:30:45">  <!-- with minutes and seconds -->
<time datetime="1914-12-20 08:30:45.687">  <!-- with minutes, seconds, and milliseconds -->
```

Figure 1 Timestamp examples

Once the timestamp has been extracted, its age can be calculating the duration between it and the present time. For ranking purposes, this age is mapped into a domain of 0 – 1 with is said to be the page's 'significance'. Young webpage, i.e. those that have been created or modified only a little while before being crawled and indexed, are likely to be more up-to-date and informative to a user than old ones. Additionally, a difference in age of one day for two webpages only a week old is much more significant that the same difference for webpages a year old. To model this phenomenon, a nonlinear mapping such as the one shown in figure 2 below has been developed. Here, a 1 corresponds to highly significant webpages while a 0 corresponds to less significant webpages.

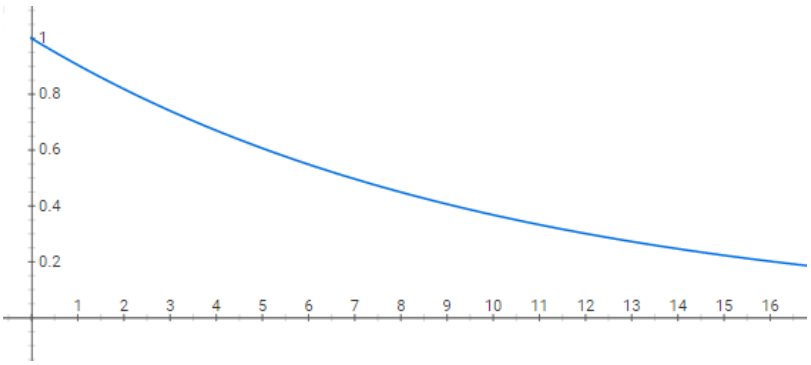

Figure 2 Webpage age-significance mapping

The overall model of the age filter incorporating the above components is shown in figure 3 below.

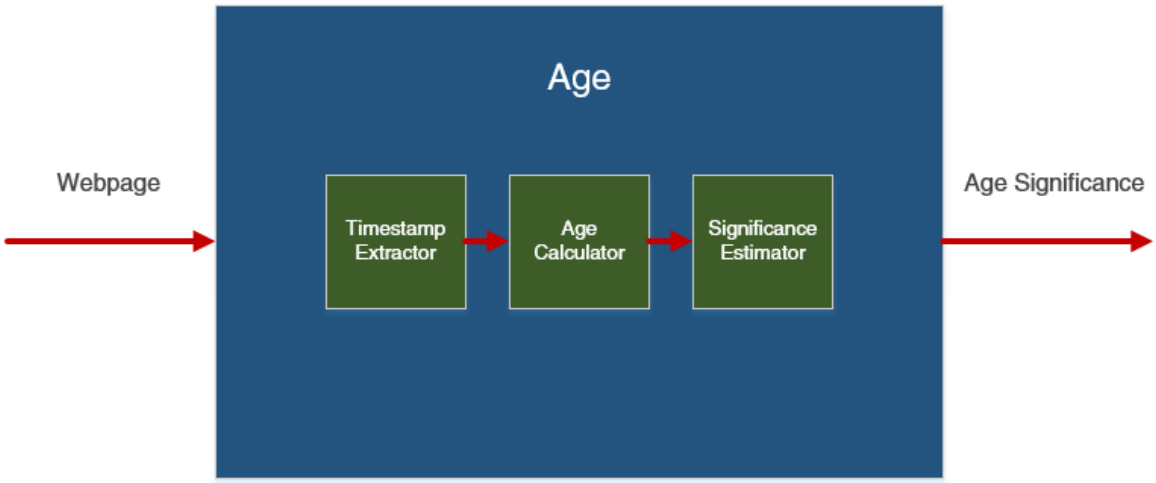

Figure 3 Age filter structure

2 Language Filter

The language filter is responsible for identifying the presence of Urdu in a webpage and subsequently estimating its proportion in it. Language identification is the task of automatically detecting the language(s) present in a document. While originally focused on assigning a language label to a document as a whole, a more fine-grained extension of this task is the estimation of the relative proportions of languages present in multilingual documents. This will allow the Urdu Search Engine conduct open-ended crawling while retaining only webpages that have a significant amount of Urdu in them. Webpages which pass the threshold set by the policy will then be forwarding for indexing, while those that fail will be discarded. Additionally, the amount of Urdu in the webpage may also be used to rank search results by placing more focus on webpages that are wholly in Urdu. These use cases of the language filter will help save on storage space while also improving the search experience.

Language identification is generally treated as a text classification problem, typically over the length of a whole document. Estimating the proportion of a language in a multilingual document, however, requires classification of words or short runs of text, which is made harder by code-switching or shared vocabulary among languages. We focus on Urdu identification and proportion estimation in an open, real-world environment without prior knowledge of other languages that may feature in a document, and without any assumptions that a document must be written in any one of a closed set of languages.

An extensive literature survey has been conducted on language identification techniques, with a focus on estimating the proportion of languages in multilingual documents. Knowledge from this

survey was used to develop a technique for the identification and proportion estimation of Urdu in webpages.

Multiple datasets of both monolingual and multilingual webpages were collected for the development and testing of the proposed technique. A dataset of 2000 monolingual BBC News stories in Urdu, Arabic, and Persian was built. The dataset containing over 1,000,000 tokens of these languages with an average wordcount of 500 words per document. A second dataset of 216 multilingual webpage samples containing Urdu alongside Arabic and English has been manually collected from the Internet, distributed by total word count and proportion of Urdu. There are six different wordcount levels: 25, 50, 75, 150, 300, 500, and six different Urdu proportion levels: 0, 0.2, 0.4, 0.6, 0.8, 1.0, with six examples for each combination of the two. A third dataset of 80 real-world webpages crawled from the Internet has been created for testing the accuracy of our technique.

The overall model of the language filter is shown in figure 4 below.

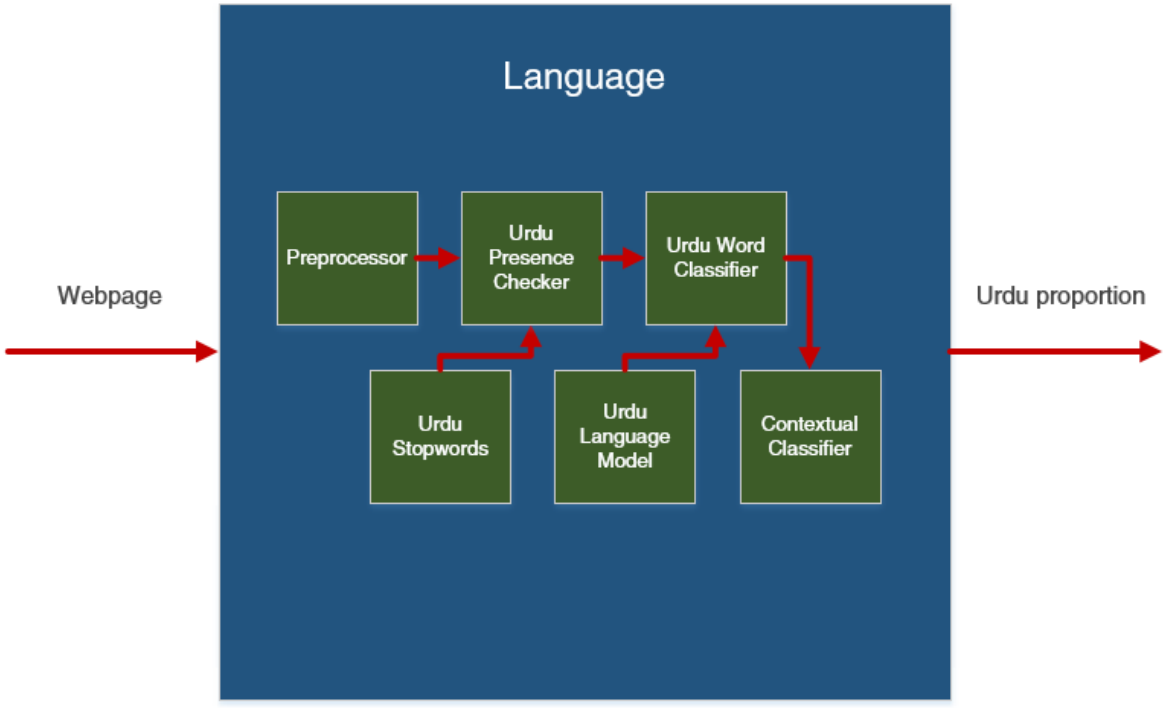

Figure 4 Language filter structure

During preprocessing, the webpage is first checked for the existence of Arabic characters. These are Unicode characters with codepoint values in the range 0600-06FF, which is the designated Unicode Arabic block. If any are found, the webpage content is normalized to remove any non-letter characters and tokenized on whitespace. For identifying the possible presence of Urdu in the content, a dictionary of Urdu stop words is used to find matches in the tokens. If any are found, Urdu proportion estimation is performed on the webpage. In this stage, an extensive language model built from the Center for Language Engineering's own 35-million-word Urdu corpus is used to classify successive words as Urdu or not depending upon whether they occur in the language model or not.

Once this preliminary classification has been performed, the result may contain some false positives due to a shared vocabulary with other languages like Arabic and Persian. To cater to this, contextual smoothing is performed over the document which classifies bigrams based on whether a majority of its neighbours were found in the language model or not.

We evaluate our approach on both monolingual and multilingual documents. On the dataset of 2000 monolingual BBC articles in Urdu, Persian, and Arabic, we achieve a mean absolute error (MAE) of 3.6%. On the dataset of 80 real-world multilingual webpages, we achieve a 5.6% MAE and a Pearson correlation of 0.95.

An alpha version of the language filter based on this technique has been built and successfully integrated into the Urdu Search Engine backend.