FIRE Programming Project Tasks

Further Information Retrieval information of the projects: Kalervo.Jarvelin@uta.fi

These projects are meant to utilize the Hadoop “Big Data” environment where appropriate in the implementation.

Contact person for doing the programming project: Jyrki.Nummenmaa@uta.fi

Programming Project A/B – Language-specific word crawler / word list manager

The task is to program an efficient language-specific word crawler. It may be based on a language-specific web page crawler, which is described below (Talvensaari 2008). The web pages may then be split into words and organized into a data structure giving words and their frequencies. Such a word list is needed for a cross-language information retrieval system, which uses a large word list as a translation aid.

Sheridan and Ballerini (1996) developed a comparable corpus based CLIR method. German and Italian documents were aligned and merged using language-independent features, i.e., dates and descriptors assigned to the documents. In this way documents discussing the same topic at the same time were identified. These documents are likely to contain similar words in German and Italian. Based on word co-occurrence statistics across the documents the researchers constructed a similarity thesaurus where query language words were related to the document language words. Queries were translated and expanded using the similarity thesaurus.

The general applicability of this approach is not clear though it seems that domain independent comparable document collections can be constructed more readily than parallel corpora. An advantage of this method is that it is multi-directional allowing querying and retrieving in any of the languages occurring in the collection.

Talvensaari (2008) developed a method for acquiring domain-specific comparable corpora from the Web. Their point of departure was the observation that CLIR resources, such as dictionaries and parallel corpora, are scarce for special domains. Obtaining comparable corpora automatically for such domains could be an answer to this problem. The Web, with its vast volumes of data, was seen to offer a natural source for this. The authors made an experiment with focused crawling as a means to acquire comparable corpora in the genomics domain. In this case, focusing crawling means collecting texts in the Web so that links are followed onwards from a found Web page only if the page fulfills two criteria: it has to be written in the desired language (one of ENG, GER, or SPA) and belong to the desired domain such as genomic. These criteria were implemented as two filters. The web pages passing the filters are aligned across the language boundaries at the paragraph level.

The acquired corpora were used to statistically translate domain-specific words. The same words were also translated using a high-quality, but non-genomics-related parallel corpus, which fared considerably worse. Talvensaari evaluated the aligned corpus translation system (called CoCot) with standard IR experiments, combining statistical translation with dictionary-based translation. The results showed improvement over pure dictionary-based translation. Therefore, mining the Web for comparable corpora was seen promising.
An alternative project is based on using a monolingual corpus for building the word-frequency list. Corpora in various languages are available. This project avoids the web crawling and language recognition phase but retains the challenge of building an efficient word list manager.


Programming Project C – Approximate String Matching through FITE-TRT

The task is to program an efficient new edition of FITE-TRT transliteration-based translation system. The system is described below.

The TRT technique was shown to yield good results in several experiments. However, in TRT, many target language candidate word forms will be generated for a single source word, and the technique does not identify which of the word forms is a correct translation. To identify the correct one among the several word forms produced by TRT, the statistical technique FITE was developed. FITE stands for frequency-based identification of translation equivalents (Loponen et al., 2008; Pirkola et al., 2006). The effectiveness of FITE-TRT has been tested between several language pairs. The tests have indicated that FITE-TRT achieved high translation recall and high translation precision, and outperformed n-grams in cross-lingual spelling variant translation. The UTACLIR cross-lingual retrieval system augmented with FITE-TRT performed substantially better than baseline dictionary-based CLIR where OOV words were kept intact.

FITE is based on the frequency pattern of translation candidates in the target collection or in the Web. For example, the frequency pattern associated with the English target word candidates for the Spanish word biosintesis in a given target collection were as follows:

<table>
<thead>
<tr>
<th>Target word candidate</th>
<th>Document Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>– biosynthesis 2 230 000</td>
<td>909</td>
</tr>
<tr>
<td>– biosintesis</td>
<td>634</td>
</tr>
<tr>
<td>– biosyntesis</td>
<td>255</td>
</tr>
<tr>
<td>– biosynthesis</td>
<td>3</td>
</tr>
<tr>
<td>– biosintessis</td>
<td>0</td>
</tr>
<tr>
<td>– biosinthesis</td>
<td>0</td>
</tr>
<tr>
<td>– biosynthesiss</td>
<td>0</td>
</tr>
</tbody>
</table>

The correct candidate clearly has a distinctive frequency.

The FITE phase of FITE-TRT scans through a list of candidates generated by TRT and gives either exactly one translation or an empty output (no translation cases) in contrast to approximate string matching where the source word always matches some target words (thus there are no ‘no translation’ cases). The FITE phase has three conditions and if those are fulfilled then a translation can be given. The first one is the beta condition which checks that the frequency of the candidate with the highest frequency value in a target language is more than a predefined beta-value (β) times the frequency of the second best candidate. If the first and second best candidates do not fulfil the beta-condition requirements, the second best candidate is compared with the third best. If the comparison meets the beta condition, then the first candidate is selected.

---

1 http://www.sis.uta.fi/infrm/julkaisut/fire/2008/TRT-ECIR%2708.pdf

2 http://www.sis.uta.fi/infrm/julkaisut/fire/2006/pirkola-FITE-TRT-SAC.pdf
because the most common among similar candidates is the most probable candidate for translation. An example of beta condition is presented below with following words and their frequency values:

<table>
<thead>
<tr>
<th>Word</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>lucille</td>
<td>20,000</td>
</tr>
<tr>
<td>lucile</td>
<td>5000</td>
</tr>
<tr>
<td>lusille</td>
<td>200</td>
</tr>
</tbody>
</table>

If \( \beta=2 \), then the first comparison between the words ‘lucille’ and ‘lucile’ satisfies the condition \((20,000 > 2*5000)\) and thus the ‘lucile’ is qualified. If \( \beta=10 \), then the comparison between first two words fails \((20,000 \leq 10*5000)\). Next the frequency values of words ‘lucile’ and ‘lusille’ are compared and this time the condition is satisfied \((5000 > 10*200)\) and again the ‘lucille’ can be qualified. Both stages fail if \( \beta=25 \).

The second condition checks that the relation between the frequency of a candidate in a target language and the frequency of a source word in a source language is valid. The frequencies are normalized using predefined parameter alpha, thus the condition is called alpha-condition. FITE takes the frequency information from word frequency lists specifically constructed for this purpose. The third condition (the length factor) checks that the length difference between the source word and the target candidate is reasonable.

**Figure 5.2.** The two-step translation process using TRT with FITE

Table 5.2. gives some indications of FITE-TRT effectiveness. While Spanish-English FITE-TRT achieves higher recall precision is approximately the same and remarkably high for both language pairs, i.e., 97.0%-98.8%. FITE-TRT was 100% correct in predicting which words are not translatable using the TRT technique, i.e., words that are native words. More results are in the reports (Loponen et al., 2008; Pirkola et al., 2006).

**Table 5.2.** FITE-TRT effectiveness. Translation recall and translation precision for *bio-terms* into English
<table>
<thead>
<tr>
<th>Source language</th>
<th>Translation Recall</th>
<th>Translation Recall %</th>
<th>Translation precision</th>
<th>Translation precision %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spanish</td>
<td>81/89</td>
<td>91.0</td>
<td>81/82</td>
<td>98.8</td>
</tr>
<tr>
<td>Finnish</td>
<td>64/89</td>
<td>71.9</td>
<td>64/66</td>
<td>97.0</td>
</tr>
</tbody>
</table>


The task is to implement an efficient fuzzy matching to search top-N most similar candidates from a large word list.

The target word list can be collected from the basic file of a HIST test database. The test database consists of three parts: (1) Basic file contains 180,468 old Finnish language newspaper articles (text recognized via optical character recognition, often from Fraktur typeface) newspapers, from years 1829-90. The basic data is copyrighted by Kansalliskirjasto and may not be copied or distributed. For details, see [1], [3], and README files at shell.sis.uta.fi (2) Topics file defines 56 search topics designed by FIRE/SIS/University of Tampere (3) Relevance judgments file expresses which documents are non-relevant (denoted by value 0); marginally relevant (value 1); fairly relevant (regular threshold) (value 2); or highly relevant (value 3) with respect to any given topic (graded judgments).

From computer science point of view, there is an efficiency challenge. Typical tokenization recognizes about 7 million unique strings (“words”) from the basic file (= index size). This high number is partially due to the level of noisiness (OCR). Task suggested: implement fast pure \textit{fuzzy matching} such as skip-grams with CCI = \{0\},\{1,2\} (see [1]), and/or some other methods, to rapidly retrieve a set of top-N (e.g., N=100) most similar strings from the index, given the user keyword. Focus is on minimizing the response time (end user needs the search word suggestions instantly).

\textbf{Alternative project:} implement a \textit{generative} method (targeted QE) so that a user given search key, such as “verotus”, is expanded with its variants to cover intended references to the search concept (for details, see [2]).

From IR point of view, the data contains special search effectiveness challenges related to (1) historical word forms, and (2) OCR errors, plus the common IR problems related to (3) word inflection, (4) compound words, and (5) derivatives. As the relevance file provides “correct answers” for topical queries, the test database can be used for testing, e.g., how effective are any of the matching methods developed (cf. [1]).

\textbf{RIGHT TO USE THE MATERIALS}

HIST copyright (c) by Kansalliskirjasto, all rights reserved. A written permission to use the data is needed. It is \textbf{FORBIDDEN TO COPY} or distribute any parts of the HIST database without prior written permission from Kansalliskirjasto. For topics and relevance judgments – please request availability from FIRE/UTA.

\textbf{REFERENCES}

