Statistical model - Course matching

Guðmundur F Hallgrímsson

24. maí 2011

The module has two main interfaces. The first is for matching existing courses to target institutions, in which case the python script receieves the corresponding course and institution ids. The other interface accepts an implicitly specified course to be matched to a target institution. In this case the script receives a description of the original course, as well as its name and ects workload. The python script then connects to MySQL and digs up all the relevant information (ids, descriptions, ects credits) both for the course to be matched (if it is passed an id) and for a list of all courses in the target institution. Currently there are two factors governing the similarities between courses: difference in ECTS and text similarities. The program uses the text similarity as a base and penalises that score if the ECTS workloads are different. This is done according to a globally defined constant, ectsPenalty.

Everything text-related here is evaluated using the Python NLTK. The Python NLTK is an open source toolkit for python that has extensive capabilities for natural language processing. In this project it is mainly used for processing a block of text into tokens, where each token stands for a single word in the text. The toolkit also has a built in method for calculating TF-IDF, which is used for estimating the text similarities. More info on this toolkit can be viewed on their website: http://www.nltk.org

The text similarity is calculated using TF-IDF (Text Frequency - Inverse Document Frequency). This means that the texts are compared using the relative frequency of the words in the original course, which is given more importance if it is more rare in the descriptions from the target institution. This number is calculated thus:

$$\mathrm{tf}_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

$$\mathrm{idf}_i = \log \frac{|D|}{|\{j : t_i \in d_j\}|}$$

$$\text{tf-idf}_{i,j} = \mathrm{tf}_{i,j} \times \mathrm{idf}_i$$

Where we have $|D|$ = the number of texts in the corpus, $|\{j : t_i \in d_j\}|$ = the number of texts where the current word appears (t_i = text i in the corpus, $n_{i,j}$ = word j in t_i). We calculate the tf-idf for every word in the original description against each of the courses in the target institution. The similarity between the

texts is then calculated using a cosine-similarity, where each text is given an array of its tf-idf values. The similarity is then given as:

$$s_t = \frac{(\text{tf-idf})_i^T \cdot (\text{tf-idf})_o^T}{\left\|(\text{tf-idf})_i^T\right\| \left\|(\text{tf-idf})_o^T\right\|}.$$

The ECTS similarity is calculated using a simple formula:

$$s_e = |\text{ECTS}_t - \text{ECTS}_o|.$$

Finally, the total similarity is then calculated and the top 10 results are returned:

$$S = s_t - s_e \cdot \text{ectsPenalty}.$$