D5: Databases and Information Systems
Advanced Topics in IR, SS 2016
Dr. Vinay Setty and Dr. Jannik Strötgen
Assignment 5a, due: **30 Jun 2016**

max planck institut informatik

You don't have to submit anything, but you'll need the output for assignment 5B. If you run into trouble, please let us know.

Semantic Search (Programming Assignment) (25 Points)

Problem 1.

In the previous assignments, you already indexed our sample corpus using elasticsearch with named entities. Next week, we want to add the possibility to search for time. For this, we have to process the document collection with a temporal tagger. The idea is as follows:

- go to https://github.com/HeidelTime/heideltime/ and get HeidelTime (either the standalone or the UIMA version).
- please see heideltime's readme and in addition (for the standalone) also the Manual.pdf in the standalone package so that you know how to set up heideltime.
- NOTE: we will cheat a bit to make processing faster: we don't use a part-of-speech tagger, so you do not have to set up the TreeTagger. Instead, you can use the "all languages tokenizer" in the uima version or set the POS tagger to "NO" in the standalone version. This produces worse results but we don't care. You want to try with TreeTagger, do so!
- process the documents of your document collection with HeidelTime, set the language to "English", the domain to "news". Process only the document text, of course.
- take care of the DCT. It is provided as metadata (pub-date) in the document collection, however, as "number of milliseconds since 1970". Check on the Web, how to get a format required by HeidelTime: YYYY-MM-DD (day information is sufficient fur us). E.g., using *date --date='@733622400'* – if you use a bash.
- if you use UIMA, see the wiki page "reproduce evaluation results" and try to set up a pipeline to process the WikiWars corpus. (you can do so without using the TreeTagger and with using the All Languages Tokenizer, of course.)
- if you use UIMA, build a collection reader, which takes as input parameter the data set file, and which creates one CAS object per document. You have to use the HeidelTime type system to set the document creation time for each document (type DCT).
- if you use UIMA, build a cas consumer, which outputs all the temporal expressions found by HeidelTime, which are of type "date" or "time". You can create a json object for each file containing the document id and all the *value*s of the temporal expressions (we don't care about where in the document the expression occurred and so on).
- if you use the standalone version, use the HeidelTime jar as a library and call the process method with language set to English, domain set to "news" and the respective DCT. You should initialize HeidelTime only once and call process for each document with its DCT.
- if you use UIMA you can check the collection readers and cas consumers in the UIMA heideltime kit to see how you can access the annotations in the CAS object.

Please let us know if you run into any trouble. We could then give you some further details on how to use HeidelTime at the end of the next lecture.