Scientific Report of Short Term Scientific Mission

COST STSM Reference Number: ECOST-STSM-IS1305-080616-078223

Period: 2016/06/12 – 2016/06/26

Duration: 14 days

COST Action: IS1305

STSM type: Regular (from Czechia to Slovenia)

STSM Title: Graphical user interface for GDEX (Good Dictionary Examples)

Guest/STSM applicant: Jan Michelfeit; Lexical Computing CZ; jan.michelfeit@sketchengine.co.uk

Host: Iztok Kosem; Trojina, Institute for Applied Slovene Studies; iztok.kosem@trojina.si

1 Purpose of the STSM

GDEX is a software part of Sketch Engine (a corpus querying tool) used to sort sentences according to their suitability as dictionary examples using a wide range of classifiers. It can be adapted for various languages or tagsets by means of a configuration file describing how candidate sentences are to be scored. Adjusting and testing the effect of various classifiers and their significance had been a tedious task, which involved repeated downloading, editing, and uploading of the configuration file. The purpose of the visit was to develop a convenient graphical user interface (GUI) for GDEX configuration development based on the input from Iztok Kosem and Kristina Koppel.

The original goal was to enable users to create a GDEX configuration file with a set of pre-determined classifiers in a point-and-click interface, adjust the classifiers' parameters, group them into tiers by significance and set their weight in the total score. Users would be able to choose a set of testing sentences, on which any changes in the configuration would immediately be visible. Other features that were considered were a break-down of the score by classifier and a comparison of different configurations.

Apart from sorting, GDEX also performs the filtering of near-duplicates, a feature whose effect had been less than apparent to users, so the editor would offer more insight into the difference between filtered an unfiltered results.

Lastly, exact evaluation of GDEX configurations is still an open question, mainly due to the scarcity of data on which to evaluate. I hoped to start a discussion with my collaborators on how to obtain such data and how to design the evaluation.

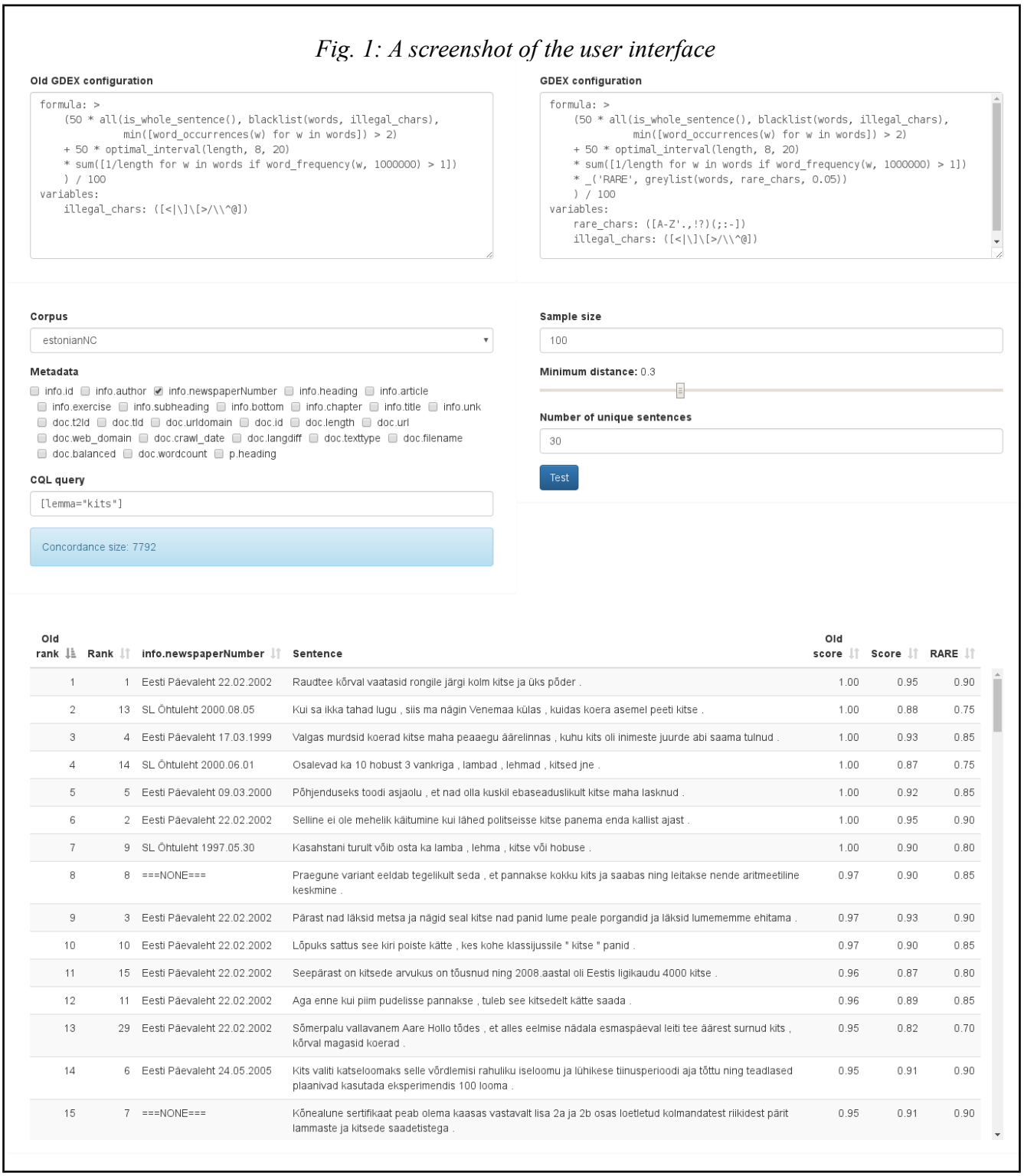
2 Description of the work carried out during the STSM

Upon my arrival in Ljubljana, we formulated a development plan based on Iztok's experience and input. It turned out a point-and-click interface was not a priority, as both Iztok and Kristina proved more than capable of editing the code directly. It is also questionable if a useful trade-off between expressiveness and ease of use can be found. In the end, we decided to prioritize the break-down of the total score to enable insight into individual classifiers and a comparison between two different configurations.

In the following days, I gradually implemented a GDEX configuration editor with the

aforementioned features based on Iztok's a Kristina's continuous feedback. The editor is now deployed at https://beta.sketchengine.co.uk/gdex_editor (configured mainly for use with the Estonian National Corpus) and Trojina's own server. It is a standalone tool independent of Sketch Engine, but integrating the two is planned, as it would be even more useful to users.

Fig. 1: A screenshot of the user interface



Features of the GDEX editor

The basic usage of the editor involves three fields: the user needs to select a **Corpus**, type in a **CQL query** to perform on the corpus and provide a **GDEX configuration**. The resulting concordance is shown in a table at the bottom of the screen. The GDEX score and rank is shown for each sentence in the concordance (or concordance sample, see below). The table is initially sorted according to rank, but the ordering can be changed by clicking any of the column headers.

The rank of the sentences is determined not only by their score, but also by de-duplication. Sentences with a high score but too similar to one of the preceding sentences are shifted down in favour of the next sentence, possibly more unique. This process can be adjusted using the **Minimum distance** parameter which determines the threshold of Jaccard distance between two sentences (treated here as sets of word-forms) under which they are deemed near-duplicates. For a minimum distance of 0, the sorting by rank equals the sorting by score.

Because scoring a large number of sentences can take a long time, especially with complex configurations, GDEX usually only takes a random sample of the whole concordance. This is true for the editor as well and users can easily change the **Sample Size**. The size of the whole concordance is also displayed for each query.

Depending on the corpus, users are able to select **Metadata** (structure attributes) to be displayed next to each sentence. This feature can highlight the representation of various parts of the corpus in the top sentences and enable the user to adjust the configuration for a more balanced, or more targeted, result.

One of the most important features of the editor is the ability to provide a second, **Old GDEX configuration** and compare the results on the same set of testing sentences side-by-side and immediately see the effect of any changes.

Apart from comparison, users are also able to mark and label parts of the mathematical expression for score calculation by wrapping the sub-expression in this code:

```
_('LABEL', <sub-expression>)
```

The value of the sub-expression, the sub-score, is then displayed in the results table in a column with the specified label. These sub-scores can be labeled and reported for both the current GDEX configuration and the old one. This finally allows the users to see what the the scores are for individual classifiers, greatly improving the ease of development.

Future plans

The editor is now in fully working condition and it has already been used by Kristina to develop a new configuration for Estonian. However, it could be integrated into Sketch Engine in some form, primarily to enable users to load their own existing GDEX configurations into the editor and save them back. It could also leverage existing access control mechanisms of Sketch Engine in order to enable access to a wide range of corpora to authorized users, without the need to configure them by hand.

As the editor enables the users to use arbitrary CQL queries, including queries for word sketch results, it would be possible to link to the editor directly from the Sketch Engine interface, enabling the users to quickly diagnose any problems with the configuration when stumbling upon them during normal corpus searching.

Kristina has brought valuable data from the automatic example sentence extraction for the Estonian Collocation Dictionary. Primarily, she was able to use this data to extend the GDEX configuration for Estonian, but I was happy to learn it also presents an opportunity for a more exact evaluation of

configurations.

The data consists of sets of 5 candidate example sentences for a number of word sketch triples from the Estonian National Corpus. From each set of 5, one sentence was selected by a human as the example sentence (although some have been edited by hand). I propose a metric for evaluating GDEX configurations for Estonian that would measure how many of the 4 rejected sentences get a lower score than the selected sentence. For a number of such quintuples, arithmetic mean seems like a useful aggregate, although looking at specific quintuples/concordances where one configuration performs significantly better or worse than another could be useful as well.

I will pursue this idea in the coming months and hopefully gain some valuable results. If the approach proves useful, the same can also be performed for other languages if we are able to obtain similar data for them.

3 References

- Kallas, Jelena; Kilgarriff, Adam; Koppel, Kristina; Kudritski, Elgar; Langemets, Margit; Michelfeit, Jan; Tuulik, Maria; Viks, Ülle (2015). Automatic generation of the Estonian Collocations Dictionary database. In: Electronic lexicography in the 21st century: linking lexical data in the digital age. Proceedings of the eLex 2015 conference, 11–13 August 2015, Herstmonceux Castle, United Kingdom. Ljubljana/Brighton: Trojina, Institute for Applied Slovene Studies/Lexical Computing Ltd., 1–20.
- Michelfeit, Jan, Jan Pomikálek, and Vít Suchomel. Text Tokenisation Using unitok. 8th Workshop on Recent Advances in Slavonic Natural Language Processing, Brno, Tribun EU. 2014.
- Kilgarriff, Adam, Vít Baisa, Jan Bušta, Miloš Jakubíček, Vojtěch Kovář, Jan Michelfeit, Pavel Rychlý, and Vít Suchomel. The Sketch Engine: ten years on. Lexicography 1, no. 1 (2014): 7-36.