GDEX for Slovene

Iztok Kosem[1], Milos Husak[2], Diana McCarthy[2]
[1]Trojina, Institute for Applied Slovene Studies, Ljubljana, Slovenia
[2]Lexical Computing Ltd., Brighton, UK
E-mails: iztok.kosem@trojina.si, milos.husak@sketchengine.co.uk, diana.mccarthy@sketchengine.co.uk

Abstract

Good Dictionary Examples or GDEX is a tool in the Sketch Engine designed to help lexicographers with identifying dictionary examples by ranking sentences according to how likely they are to be good candidates. The ranking is done automatically using various syntactic and lexical features. So far, only GDEX for English has been available. This paper presents the design and evaluation of Slovene GDEX, which was used for finding good examples for the new lexical database of Slovene, one of the activities in the Communication in Slovene project. Several different GDEX configurations were designed, evaluated and compared. The evaluation involved examining sentences of lemmas belonging to different word classes. Good sentences were logged for subsequent analysis with external data-mining software, WEKA. The observed behaviour was then used to adjust the parameters of the GDEX classifiers. We believe that the procedure of identifying features of good examples and their values, described in this paper, can be used for the development of GDEX for any language.

Keywords: dictionary example; GDEX; lexical database for Slovene; Sketch Engine; data mining

1. Introduction

Examples are a very important part of a dictionary entry, as they illustrate how the word is used in a particular meaning, construction or pattern. Examples provide additional support to the definition, which is sometimes hard to understand without reading the examples (Atkins & Rundell, 2008). Furthermore, examples can be of great help with navigating through longer entries, where the users can "identify the particular sense they are seeking by finding examples that are similar to the one they need or have in front of them" (Fox, 1987:137).

Good dictionary examples have to be natural and typical, informative and intelligible (Atkins & Rundell, 2008). Taking all these criteria into account makes finding a good dictionary example a time-consuming task for a lexicographer, as the search for a good example requires the inspection of a number of different features. These features include sentence length, full-sentence form, non-complex structure, and lack of rare words and/or anaphora. As corpora grow bigger and bigger, lexicographers have more sentences to choose from, so there are more likely to find good examples; on the other hand, this also means they need to inspect more examples.

Good Dictionary Examples or GDEX (Kilgarriff et al. 2008) is a tool in the Sketch Engine (http://the.sketchengine.co.uk) designed to help the lexicographers identify dictionary examples by ranking of sentences according to how likely they are to be good example candidates. The ranking is done automatically using various syntactic and lexical features. The usefulness of GDEX for English has been confirmed on an actual dictionary project, namely when selecting additional examples for the online version of the Macmillan English Dictionary. However, there were still parts of the heuristics that were identified as open for improvement. Furthermore, as most of the GDEX settings were English-specific, the usefulness of GDEX for other languages was limited.

This paper presents the development of GDEX for Slovene that was used in the building of the new lexical database of Slovene. First, the basic information on GDEX is presented, including an overview of the characteristics of examples that can be measured. Then, GDEX for English is discussed in more detail. Next, the design of GDEX for Slovene is provided, including a description of our approach for devising the heuristics. Also, the evaluation process of GDEX for Slovene is presented in detail, from the comparison of different configurations to the evaluation of the effectiveness of GDEX at words from different word classes. In conclusion, the lessons learned during the design of GDEX for Slovene are summarized and future plans are laid out.

2. GDEX

GDEX is a tool for ranking sentences according to specified criteria. It was designed for use by the Sketch Engine (Kilgarriff et al, 2004) to sort concordances in a way that is useful to lexicographers when creating dictionaries. The aim is to separate good candidates for dictionary examples from the bad candidates.

The most important criteria of good dictionary examples are usage typicality, informativeness and intelligibility (Kilgarriff et al., 2008), however these are difficult to describe and measure directly, therefore GDEX circumvents the problem by measuring observable features, such as sentence length, word length, presence/absence of black/whitelisted words/non-words (urls, numbers, etc.) which are related to the more covert criteria. Using corpora, GDEX can take into account word frequencies, common collocations and available word attributes (e.g. part-of-speech, lemma, grammatical

tags, etc.). Depending on the resources available for the language or domain of the text, it is also possible to use other sources of linguistic information such as the degree of ambiguity of words contained in the sentences.

Originally, GDEX was developed as a set of classifiers for specific features which each performed normalization and scoring in a fixed way and returned a score in the range from 0 to 1. These scores were then combined in a weighted average to provide a single score for each sentence. The number of parameters for each individual classifier was limited to facilitate their automatic optimization based on the training data. GDEX has subsequently been adapted so that the individual values of features (for example, the number of words in a sentence) can be accessed directly and any normalization and aggregation is now fully customizable. This makes it possible to manipulate GDEX more precisely.

The normalization can be performed with respect to sentence length, corpus size or a fixed interval of values. In case of measuring features of individual tokens (e.g. word frequency, collocation score) it is also necessary to decide whether to take an average, minimum, maximum or sum of all values of the tokens in a sentence. The aggregation functions are used to combine the measurements into a single value that is used for sorting the sentences according to their suitability to serve as a dictionary example. The specification of measured features and the way they are combined together is defined in files called GDEX configurations.

2.1 GDEX for English

The original GDEX configuration for English was based on a set of concordances with good sentences manually annotated. Using this data, a set of various classifiers was optimized so as to rate the sentences which were marked as good higher than the others.

It became apparent that the most successful criterion was the number of long words, which tend to be harder to understand. This single classifier improved the relative position of good sentences in the test concordances by 34% (100% is the case where all marked good sentences are at the top positions higher than any non-selected sentence, and 0% improvement corresponds to a random ordering of sentences).

After several experiments with different combinations of analysed features, the following subset of all classifiers was chosen with an overall improvement of 49% over the random baseline. Besides the long words penalization, the following classifiers were used in the collection:

- penalization for interpunction marks, brackets and apostrophes;
- preference for sentences with length within an optimal interval (between 6 and 28 tokens (i.e. words and interpunction marks));
- penalization for proper nouns (based on capital letters);
- penalizaton for multisense words (based on the number of WordNet (Fellbaum, 1998) synsets that the word belongs to);
- fraction of low frequency words within the sentence (the best results were achieved with a threshold frequency of 107 in the British National Corpus (Leech, 1992), which corresponds roughly to sentences focusing on the 35,000 most common words);
- relative keyword position in the sentence (in the training data the best results were achieved with the keyword at the beginning of the sentence);
- penalization for mixed symbol words (such as email address, urls, etc.);
- penalization for anaphoric expressions (based on a word-list);
- preference for complete sentences (starting with a capital letter and ending with ".", "?" or "!");
- since GDEX was intended to be used on corpora collected from the internet, a classifier that completely banned sentences mistakenly containing html or xml markup;
- for practical reasons, as the sentences proposed by GDEX were intended for publication in dictionaries, we also added a blacklist for sensitive and offensive nouns.

Values of all classifiers were then balanced one-by-one using an eager algorithm to obtain weights for a weighted average that was used as the main aggregation function.

The presumption was that after adjusting the parameters of individual classifiers, a similar set of classifiers would give good results for Slovene as well. Therefore, we used this as a start point for the project, but without language dependent blacklists and the polysemy classifier, for which no data were available.

3. GDEX for Slovene

There are two ways for devising a GDEX configuration for a new language. The first, which was pursued in the original work (Husak, 2008), depends on training classifiers on annotated data. The human input in that case is limited (besides annotating data) to the choice of classifiers.

The other approach, pursued in this project, arose from the assumption that humans experienced in lexicography can provide a useful set of heuristics based on their intuition or knowledge. We used a small amount of annotated data, far smaller than would be required for machine learning, with an external tool that helps visualize the data and provide the user with additional statistical information.

The tool of choice was WEKA (Hall et al., 2009), because not only can it help in visualizing the data and

providing the statistics, but it also contains extensive functions for data filtering and manipulation. The other reason was that machine learning algorithms implemented in WEKA open many new possibilities for future experiments.

WEKA itself does not work with corpora, therefore we use GDEX analysers to do the measurements, which we export into an attribute-relation file format (ARFF) that can be opened by WEKA. In WEKA, we can immediately see the minimum, maximum, mean value, standard deviation and value distribution for each of the analysed features (Figure 1). More importantly, any two measurements can be plotted in a 2D chart that shows how well they separate good and bad sentences (Figure 2). Using this information, one can make sense of the data and make better decisions when creating a GDEX configuration.

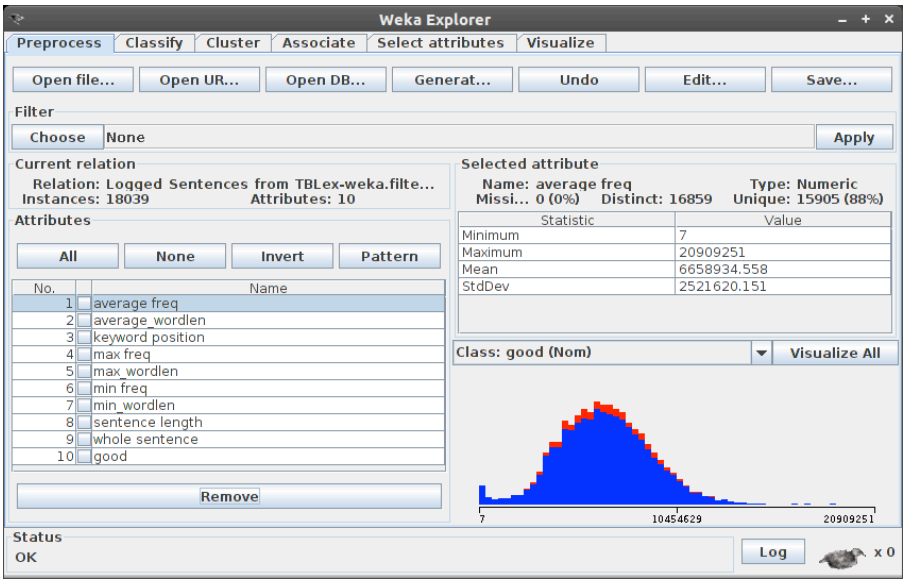

Figure 1: Different features of good examples (left) and the statistics for average frequency (right) in WEKA

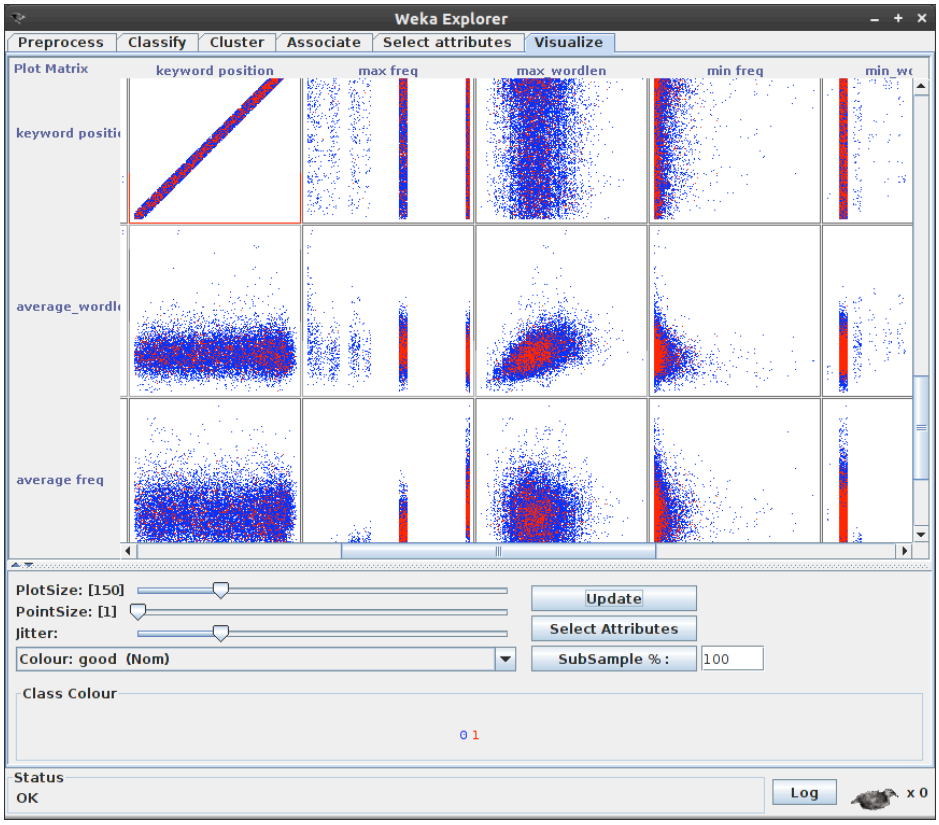

Figure 2: WEKA 2D visualisation charts of good (red) and bad (blue) examples according to different features

3.1 Design

The design of GDEX for Slovene was motivated by the needs of lexicographers working on the entries of the new lexical database for Slovene (Gantar & Krek, 2011)[1]. Their work involved selecting a great number of examples for each entry from the 620-million-word FidaPLUS corpus – each construction, pattern, phrase etc. had to be attested with at least one example from the corpus, though more than one was preferred.

Initially, the GDEX for English settings were used when selecting the examples, however this proved to be ineffective as good examples were rarely found among 10-15 examples (this setting was most commonly used) for a collocate in the word sketch. The survey among the lexicographers working on the project showed that the use of GDEX was even counter-productive – most of them switched the GDEX feature off or stopped using the TickBox Lexicography feature in Word Sketch. It was thus decided to devise a GDEX configuration that would take into account the linguistic characteristics of the Slovene language when ranking the examples. An important criterion considered when designing and evaluating configuration(s) was the needs of the project; namely, the purpose was to find good examples for a lexical database rather than a dictionary.

The design of GDEX for Slovene consisted of the following stages: selecting classifiers for the first GDEX for Slovene configuration, determining the values of classifiers, evaluating the configuration on the word sketches of selected lemmas, devising an improved configuration, evaluating the two configurations and comparing their results, devising the third configuration based on the findings, evaluating and comparing the results of the configurations, etc.

The classifiers used in configurations for GDEX for English were used as a point of departure, excluding English-specific classifiers such as polysemy (WordNet synsets), blacklists of offensive nouns, and lists of anaphoric expressions. The values of certain classifiers, e.g. preferred sentence length, were determined with the WEKA tool, using the existing examples in the lexical database for Slovene, which were selected manually by lexicographers, as a benchmark.

The first configuration, named Slovene1, had the following heuristics:

- preferred sentence length: 8 to 30 words;
- threshold of low frequency words: 104;
- keyword position: beginning of the sentence (in the first 20% of the tokens);
- penalty for words containing regular expressions;
- penalty for sentences containing urls, email addresses, etc.;
- penalty for sentences containing capital letters;
- penalty for proper nouns;
- penalty for pronouns;
- a good example had to be a whole sentence;
- a good example could not contain words that occur less than three times in the FidaPLUS corpus.

Higher weight (value of 2 rather than the normally used 1) was attributed to the preferred sentence length, threshold of low frequency words, proper noun penalization, and pronoun penalization as these features were reported by the lexicographers as both the most crucial criteria in identifying good examples, and the most indicative for identifying elements of bad examples.

When selecting examples, lexicographers are also looking for diversity, so that each selected example offers a different type of information. So if GDEX produces 10 good examples, which are all very similar, the lexicographers will probably select only one of them – the other examples are treated as "bad" ones due to their similarity to the selected example. To ensure diversity of examples offered by GDEX, a script was included in GDEX that ensured that the difference between the examples was at least 30% measured in the Levenshtein distance (Levenshtein, 1966).

Four other configurations (Slovene1b, Slovene2, Slovene3, and Slovene3b) were devised in succession during the process of evaluation. These configurations were basically variants of the initially devised Slovene1, as they included minor incremental tweaks to the classifier values and/or weights.

3.2 Evaluation

Evaluation was an important part of the GDEX for Slovene design process, since it helped to determine the efficiency of the configuration, and to identify frequent features of good and bad examples and suggest further improvements (i.e. tweaking) to the configuration.

Evaluation consisted of the manual examination and selection of good examples of collocates in the word sketches of selected lemmas. The evaluation was conducted on word sketches because it allowed us to log the selected examples using TickBox Lexicography, analyse them and identify any required changes in the values of classifiers, or create completely new classifiers. Another reason for using word sketches was to simulate the actual conditions in which GDEX for Slovene would be used. Word sketches are the main source of entry information in the lexical database for Slovene; once the

[1] The new lexical database for Slovene is part of the Communication in Slovene project (http://www.slovenscina.eu) which is partly financed by the European Union, the European Social Fund, and the Ministry of Education and Sport of the Republic of Slovenia. The operation is being carried out within the operational programme Human Resources Development for the period 2007–2013, developmental priorities: improvement of the quality and efficiency of educational and training systems 2007–2013.

lexicographers make an initial meaning division of the headword, they analyse the word sketch to obtain information on grammatical constructions, collocates, and patterns of the headword, and extract examples.

The default Word Sketch settings were adapted for evaluation purposes. The number of examples per collocate for the TickBox Lexicography view was set to 10, as this was the recommended setting, and the most frequently used setting of lexicographers working on the lexical database. On a related note, the minimum frequency of a collocate, i.e. the number for the word sketch from which the TickBox Lexicography selection is made, was set to 15, since there was little sense in examining examples of collocates with frequency of less than 15, given that the lexicographers can quickly examine (all) the examples of such collocates, and that the only change made by GDEX in such cases is the order in which the examples are provided.

Lemmas used for the evaluation were selected from the list of existing entries in the lexical database for Slovene. As the lexical database currently contains only nouns, verbs, adjectives, and adverbs, the list of lemmas included only words from these four word classes. The aim was to make the selection as heterogeneous as possible, so the lemmas included both abstract and concrete nouns, monosemous and (highly) polysemous words, words with few and many constructions, patterns, etc.

The evaluation was conducted by two people who examined and selected the examples offered by GDEX, and wrote their observations, comments and suggestions in a shared online document. When making comments and suggestions, the evaluators had to consider which measurable characteristics of examples, i.e. the classifiers of GDEX configurations, needed improvement, had to be given less/more weight, or had to be added to the configuration. This was then taken into account when devising new configurations.

It was considered important that the evaluators were able to compare the results of different configurations in order to quickly determine any improvements in results, if any, produced by newer configurations. For this reason, a special setup of GDEX in Tickbox Lexicography was designed to allow side-by-side comparison of example ranking by two different configurations (Figure 3), the left-hand column showing the results of the currently selected GDEX configuration, and the right-hand the results of the configuration used for comparison. Good examples could be selected for the current configuration only. The configurations, and their results, in the right-hand column could be quickly changed by selecting another configuration from the drop-down menu.

Figure 3: Side-by-side comparison of two GDEX configurations in TickBox Lexicography (Sketch Engine)

3.2.1. Evaluation criteria

The main evaluation criterion was the number of good examples per collocate. As the selected examples were logged, such information was easy to obtain. Initially, the aim was to devise a configuration that would yield at least 5 good examples out of 10 per collocate. However, after evaluating several different configurations, it became clear that 3 examples per collocate was a more

realistic aim. Such a figure was also considered acceptable because the number of examples per collocate, construction, or pattern in the lexical database rarely exceeds two.

The evaluators selected good examples considering the criteria of naturalness, typicality, and intelligibility. Informativeness was attributed less importance due to its close relation to the meaning division – "an informative example is one that complements the definition and helps the user understand it better" (Atkins & Rundell, 2008:460) – given that automatic word sense disambiguation is not possible (yet), we could not use meaning division as a criterion for selecting examples in GDEX evaluation; in other words, all meanings of the word had to be considered.

Examples were also considered good (for the lexical database) if they showed the potential to be turned into good dictionary examples. Consequently, minor breaking of the "good dictionary example" principles was tolerated, especially the ones related to the length and complexity of examples – e.g. sentences were allowed to be longer, i.e. they were allowed to have more context that could be reduced or removed, sentences could have "removable" relative clauses with less frequent words or proper nouns, etc. This of course meant that we made the decision to allow for subsequent modification of corpus examples for dictionary purposes; in fact, as evidence suggests, such practice cannot be completely avoided even in corpus-driven dictionary projects (Atkins & Rundell, 2008; Landau, 2001; Krishnamurthy, 1987).

3.2.2 Findings

The evaluation showed that certain classifiers played a much more significant role than others in the production of good examples by the configurations. These classifiers were preferred sentence length, relative keyword position in the sentence, penalty for keyword repetition, penalty for words exceeding the prescribed maximum length, and penalty for sentences exceeding maximum length. The parameters of these classifiers were consequently given more focus and were subject to changes when devising new configurations. The values of the aforementioned classifiers in different configurations, which also point to the differences between the configurations, are shown in Table 1.

The most significant classifier for good example identification was sentence length. In the first three configurations, the preferred sentence length was between 8 and 30 words, but the evaluation pointed to the lack of good examples, mainly on account of examples being too short. Shorter sentences often proved to lack context, whereas longer sentences were more often better examples, or at least had more potential to be turned into good examples. Once the preferred sentence length was increased (configurations Slovene3 and Slovene3b), and more importantly, once the minimum length was increased to 15, the average length of examples increased (e.g. compare the examples of two configurations in Figure 3), and the number of good examples per collocate improved considerably. The improvement was observed at almost all the lemmas used in the evaluation, regardless of word class.

Another key classifier for good example detection was relative keyword position. The initial criterion specifying that the position of the keyword should be at the beginning of the sentence (first 20 % of the sentence) frequently promoted bad examples as they were not informative enough or lacked the necessary context. In the Slovene2 configuration the condition was removed, but this did not considerably improve the ratio of good vs. bad examples. It was observed that in good examples, the keyword almost always occurred between the middle and the end of the sentence. Once the new span for keyword position was implemented, the number of good examples per collocate improved, in certain cases significantly. However, with certain verbs, the preferred keyword position towards the end of the sentence actually proved to be even more problematic as the verb lacked the context necessary for understanding its meaning.

	Slovene1	**Slovene1b**	**Slovene2**	**Slovene3**	**Slovene3b**
sentence length	min 8 max 30	min 8 max 30	min 8 max 30	min 15 max 35	min 15 max 35
keyword position	0–20% of the sentence	0–20% of the sentence	not used	40–100% of the sentence	40–100% of the sentence
penalty for keyword repetition	NO	NO	NO	YES	YES
maximum word length is 18 characters	NO	NO	YES	YES	YES
maximum sentence length is 60 tokens	NO	NO	YES	YES	YES

Table 1: Key differences between the GDEX configurations

Several comments made during the evaluation referred to the fact that examples that contained more than one occurrence of the keyword were almost never good examples, as they were too difficult to understand and/or lacked the necessary context to be a good attestation of the meaning, collocate, construction or pattern. As a result, the classifier that penalized all the repetitions of the keyword in the same sentence was added (to configurations Slovene3 and Slovene3b).

Two classifiers that banned sentences longer than 60 words, and/or containing words longer than 18 characters were added based on WEKA analysis of the examples selected during evaluation. The impact of the two classifiers was difficult to observe during the evaluation, however as far as the classifier for word length was concerned, the subsequent examination of the wordlist of lemmas from the 1,13-billion-word Gigafida corpus, an upgrade of the FidaPLUS corpus, revealed that very few lemmas consisting of 18 characters or more were actual words – the vast majority of lemmas were websites, parts of websites, email addresses, errors that occurred during the conversion of various file formats into txt (e.g. several words joined into one word), etc., i.e. mainly items that were also penalized by other classifiers.

Weights of certain classifiers were also subject to experimentation (see Table 2), however the evaluation showed that the original weight setting produced the best results. For example, when the weight of classifiers penalizing proper nouns and pronouns respectively was lowered to 1, the evaluators observed more cases of bad examples containing proper nouns and, to a lesser extent, pronouns. Similarly, changing the weight of relative keyword position in the sentence configuration did not have any significant effect on results – in the case of this classifier, the parameters were much more relevant for identifying good examples.

	Slovene1	**Slovene1b**	**Slovene2**	**Slovene3**	**Slovene3b**
keyword position	1	1	/	1	2
penalty for proper nouns	2	1	1	2	2
penalty for pronouns	2	1	1	2	2

Table 2: The classifiers and configurations where changes in weights were made

It is also noteworthy that the evaluation proved the usefulness of other classifiers, such as the classifier allowing only whole sentences, and classifiers penalizing for regular expressions, urls, email addresses, etc. Examples that were not whole sentences were found only at collocates with low frequency that lacked better examples, i.e. most examples were not whole sentences. Similarly, regular expressions were rarely encountered in the provided examples, while urls and email addresses never appeared in the examples, which was, at least at configurations Slovene2, Slovene3, and Slovene3b, also related to the introduction of the maximum word length limit.

In the end, Slovene3 was selected among all the GDEX configurations since it produced the best results for different types of lemmas (nouns, verbs, adjectives, adverbs). The configuration was implemented in the Sketch Engine and used by the lexicographers working on the lexical database for Slovene. Several lexicographers soon reported a significant improvement in the helpfulness of GDEX when searching for good examples.

3.2.3 Remaining issues

Some evaluation findings and observations could not be fully addressed during this particular development of the GDEX for Slovene, and we list them here as they indicate which direction the further development of the configurations for Slovene might take. Moreover, these findings may be useful for the developers of GDEXes for other languages.

One common feature of bad examples was the occurrence of the sentence initial adverb (e.g. *nato*, *tako, torej, potem, poleg tega, zaradi tega, zato ker*) [2] that linked the examples with the preceding sentence. This often meant that the example did not contain enough context to be understandable. The planned solution is to devise a blacklist of sentence-initial adverbs and any other words that feature in bad examples, based on the frequency list of sentence-initial words from the corpus.

Another issue was that examples ending with something other than full stop, question mark or exclamation mark, the punctuation marks allowed by the whole sentence classifier, were still offered in the results. The sentence-ending punctuation mark that proved problematic was ellipsis; sentences ending in ellipsis were often among the top 10 offered by GDEX (all the configurations), if not even at the very top. This was not caused by the lack of good example candidates or errors in tokenisation (i.e. ellipsis treated as three full stops).

[2] The English translations of the sample adverbs are *after that, so, then, in addition, because of that, because*.

Further fine-tuning of the whole sentence classifier will be required, based on the analysis of sentence-ending punctuation in bad examples.

As mentioned in 3.2.2, the parameters of certain classifiers (e.g. relative keyword position) did not work well for all types of lemmas, and the same was true for different configurations. For example, Slovene3 and Slovene3b produced much better results for nouns and adjectives than Slovene1, Slovene1b, and Slovene2. For verbs, the differences between the results of different configurations were much smaller, however sometimes Slovene1 and Slovene1b produced better results than the other settings. Furthermore, the differences between the effectiveness of different configurations were also observed at the level of grammatical relations in word sketches. For example, for some nouns, the average number of good examples in the grammatical relations containing preposition collocates was sometimes considerably lower than in grammatical relations containing lexical words. These findings reveal the disadvantage of using a single GDEX configuration for different types of lemmas, and suggest that perhaps GDEX configurations should be tailored more narrowly, e.g. to a particular word class or even a category of lemmas within a word class.

4. Conclusion

GDEX is a very helpful tool for any lexicographer, considering how many examples need to be examined and selected during dictionary compilation. By ranking examples according to their potential to be good example candidates, GDEX acts as a sort of a sieve, pushing bad example candidates towards the bottom of the list and thus making it less likely that the lexicographers will waste time inspecting them during selection. On the other hand, by pushing good example candidates towards the top of the list, GDEX saves lexicographers' time by making it more likely that good examples will be found quickly.

The experience with English GDEX has shown which characteristics of examples are important for automatically determining whether an example is good or bad. As the experience in designing the GDEX for Slovene has shown, certain classifiers are not language-specific, such as the classifier banning sentences that are not complete, and the classifier penalizing sentences containing urls, email addresses, and regular expressions. Also classifiers penalizing proper nouns and pronouns are easily transferred to other languages assuming the appropriate distinctions are made with the part-of-speech tagger. More language-specific appear to be classifiers such as relative keyword position, where at least for Slovene, the optimal setting seems to be almost opposite to the setting for English. Classifiers such as preferred sentence length in particular, but also the threshold of low frequency words are less language-specific and more project-specific. It is also noteworthy that despite the fact that the criteria of good examples for Slovene GDEX were somewhat different to the criteria for English GDEX, the weights attributed to classifiers did not change significantly.

One of the important contributions of this project to further development of GDEX is its methodology. The process of using a data visualisation tool (WEKA) for determining and improving the values of classifiers, based on analysis of (logged) good and bad examples, combined with manual evaluation and comparison of the examples produced by different configurations has proven effective. This methodology can be used in further development of English GDEX and Slovene GDEX, as well as in the development of GDEXes for other languages. There are still improvements to be made, for example we feel that the WEKA tool can be exploited much more extensively.

Plans for the future include improving the existing GDEX for Slovene by adding blacklists, e.g. of sentence-initial adverbs, and polysemy classifier based on synsets from the wordnet for Slovene, i.e. sloWNET (Fišer, 2009). In addition, a further analysis of good and bad examples with WEKA is foreseen, in order to identify and test new classifiers.

The next project in which GDEX for Slovene will be used, will be the automatic extraction of grammatical relations, collocates, and examples for the entries in the lexical database for Slovene. For this project, the aim will be to design a GDEX configuration where the **top** two or three examples offered are always good examples. To achieve this, we plan to devise and test configurations specific to a particular category of words, e.g. nouns, verbs, adjectives, or even specific to a particular subcategory of lemmas within a word class, e.g. monosemous nouns.

5. Acknowledgements

We would like to thank Simon Krek and Polona Gantar for helping with the design and evaluation of GDEX for Slovene, the lexicographers working on the new database for Slovene for valuable feedback on GDEX, and the Sketch Engine team for technical support.

6. References

Atkins, B.T.S., Rundell, M. (2008). *The Oxford Guide to Practical Lexicography.* Oxford: Oxford University Press.

Fellbaum, C. (ed.) (1998). WordNet: An Electronic Lexical Database. MIT Press.

Fišer, D. (2009). sloWNET - slovenski semantični leksikon. In M. Stabej (ed.) Proceedings of 28th symposium Obdobja. Ljubljana: University of Ljubljana, pp. 145-149.

Gantar, P., Krek., S. (2011). Slovene Lexical Database. In D. Majchráková, R. Garabík (eds.) *Natural language Processing, Multilinguality. Sixth*

International Conference. Modra, Slovakia, 20-21 October 2011. Slovenská akadémia vied, Jazykovedný ústav L'udovíta Štúra, pp. 72-80.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P. & Witten, I.H. (2009). The WEKA Data Mining Software: An Update. *SIGKDD Explorations*, 11(1), pp. 10-18.

Husak, M. (2008). Automatic Retrieval of Good Dictionary Examples, Bachelor thesis. Masaryk University, Brno, Czech Republic.

Kilgarriff, A., Husák, M., McAdam, K., Rundell, M. & Rychlý, P. (2008). GDEX: Automatically finding good dictionary examples in a corpus. In E. Bernal, J. DeCesaris (eds.) *Proceedings of the 13th EURALEX International Congress.* Barcelona: Institut Universitari de Linguistica Aplicada, Universitat Pompeu Fabra, pp. 425-432.

Kilgarriff, A., Kovar, V., Rychlý, P. (2009). Tickbox Lexicography. In S. Granger, M. Paquot (eds*.) eLexicography in the 21st century: New challenges, new applications, Proceedings of eLex 2009, Louvain-la-Neuve, 22-24 October 2009*. Louvain-la-Neuve: Presses Universitaires de Louvain, pp. 411-418.

Kilgarriff, A., Rychlý, P., Smrz, P. & Tugwell, D. (2004). The Sketch Engine. In G. Williams, S. Vessier (eds.) *Proceedings of the 11th EURALEX International Congress*. Lorient: Universite de Bretagne-Sud, pp. 105-116.

Krishnamurthy, R. (1987). The Process of Compilation. In J. Sinclair (ed.) *Looking up: An Account of the COBUILD Project in Lexical Computing.* London and Glasgow: Collins ELT, pp. 62-85.

Landau, S. (2001). *Dictionaries: the Art and Craft of Lexicography*. Cambridge: Cambridge University Press.

Leech, G. (1992). 100 million words of English: the British National Corpus (BNC). *Language Research* 28(1), pp. 1-13.

Levenshtein, V.I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady,* 10, 707–10.

The Communication in Slovene project. Accessed at: http://www.slovenscina.eu.