Automation of lexicographic work: an opportunity for both lexicographers and crowd-sourcing

Iztok Kosem[1], Polona Gantar[2], Simon Krek[3]

[1]Trojina, Institute for Applied Slovene Studies, Ljubljana, Slovenia
[2]Fran Ramovš Institute for the Slovene Language, ZRC SAZU, Ljubljana, Slovenia
[3] Jožef Stefan Institute, Ljubljana, Slovenia
E-mail: iztok.kosem@trojina.si, apolonija.gantar@guest.arnes.si, simon.krek@guest.arnes.si

Abstract

A new approach to lexicographic work, in which the lexicographer is seen more as a validator of the choices made by computer, was recently envisaged by Rundell and Kilgarriff (2011). In this paper, we describe an experiment using such an approach during the creation of the Slovene Lexical Database (Gantar & Krek, 2011). The corpus data, i.e. grammatical relations, collocations, examples, and grammatical labels, were automatically extracted from the 1.18-billion-word Gigafida corpus of Slovene. An evaluation of the extracted data consisted of making a comparison between a manual entry and a (semi)-automatic entry, and identifying potential improvements in the extraction algorithm and in the presentation of data. An important finding was that the automatic approach was far more effective than the manual approach, without any significant loss of information. Based on our experience, we would propose a slightly revised version of the approach envisaged by Rundell and Kilgarriff in which the validation of data is left to lower-level linguists or crowd-sourcing, whereas high-level tasks such as meaning description remain the domain of lexicographers. Such an approach indeed reduces the scope of lexicographers' work; however, it also results in the ability of making content available to the users more quickly.

Keywords: automatic extraction, crowd-sourcing, Slovene Lexical Database, validation

1. Introduction

The last decade has been very eventful for lexicography, mainly due to technological progress. This allowed the building of larger and larger corpora, providing lexicographers access to increasingly larger databases of language. In addition, the introduction of the electronic medium and the online format in particular, which has truly established itself as the main medium for dictionary content in most parts of the world, has meant that dictionary content can be available to users faster than ever before.

However, technological progress has also brought about new challenges for lexicographers: there is (much) more data to analyze, and less time to do so due to (more) demanding users. Various tools such as Word Sketch (Kilgarriff and Tugwell, 2002) and TickBox Lexicography (Kilgarriff et al., 2010) have been designed as part of corpus query systems to help lexicographers tackle this problem, but their design and purpose still requires lexicographers to select and transfer relevant corpus information to the dictionary writing system.

These new challenges for lexicographers have prompted researchers to rethink the definition of what lexicographer's work should entail. Recently, a new approach to lexicographic work, in which the lexicographer is seen more as a validator of choices made by a computer, was envisaged by Rundell and Kilgarriff (2011). As they argue "it is more efficient to edit out the computer's errors than to go through the whole data-selection process from the beginning". This approach redefines not only the lexicographer's tasks but also the role of a corpus in the lexicographic process.

In this paper, we describe an experiment using such an approach during the creation of a new lexical database for Slovene. Firstly, we present the lexical database, describing its contents and structure. Next, we focus on the method of automatic data extraction from the corpus, outlining the elements needed for developing the algorithm for data extraction, and describing the output. Then, we focus on evaluation of the automatic method, by comparing it with the "manual" method used in the early stages of building the lexical database, examining its accuracy, and pointing out the parts that can still be improved. A section is dedicated to a planned implementation of automatic methods in the compilation of a proposed new dictionary of contemporary Slovene, where crowd-sourcing would also be utilized as a clean-up stage between automatic extraction of data and lexicographic editing. We conclude by considering future improvements of the method, as well as discussing which other approaches could be made more automatic and combined with the method presented here.

2. Slovene Lexical Database

The Slovene Lexical Database (SLD) is one of the results of the Communication in Slovene[1] project, a project that has developed language data resources, natural language processing tools and resources, and language description resources for Slovene. The SLD has a twofold goal: it is intended as the basis for the future compilation of different dictionaries of Slovene, both monolingual and bilingual, and as such its concept is biased towards lexicography. Secondly, it will be used for the enhancement of natural language processing tools for Slovene.

Reflecting its two-fold purpose, the SLD contains two different types of information. On the one hand, there is lexico-grammatical information – intended for human end users – such as sense descriptions in semantic frames, representing the starting point for whole sentence definitions (Sinclair, 1987), collocations attributed to particular senses of the lemma, and examples from the corpus. On the other hand, there is

[1] The operation 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 program Human Resources Development for the period 2007–2013, developmental priorities: improvement of the quality and efficiency of educational and training systems 2007–2013. Project web page: http://eng.slovenscina.eu/.

information designed for natural language processing tools. This information is encoded in a more complex way and, in addition to its immediate use in NLP tools, requires an expert to process or interpret it. Among this information is the formal encoding of syntactic patterns on the phrasal and clause level as well as the formal encoding of semantic arguments and their types.

The database is conceptualized as a network of interrelated lexico-grammatical information on six hierarchical levels with the semantic level functioning as the organizing level for the subordinate ones. The six levels are:

a) Lemma, or the headword, representing the top hierarchical level and functioning as the umbrella for all lexical units placed under it.

b) Senses and subsenses, labelled with semantic indicators, whose primary function is to form a sense menu intended for easy navigation within a polysemic entry structure. Another kind of information recorded on the sense level is semantic frames which are conceptually close to frames in the FrameNet project (Fillmore & Atkins, 1992; Baker, Fillmore & Cronin, 2003) and to prototypical syntagmatic patterns in the Corpus Pattern Analysis system (Hanks, 2013).

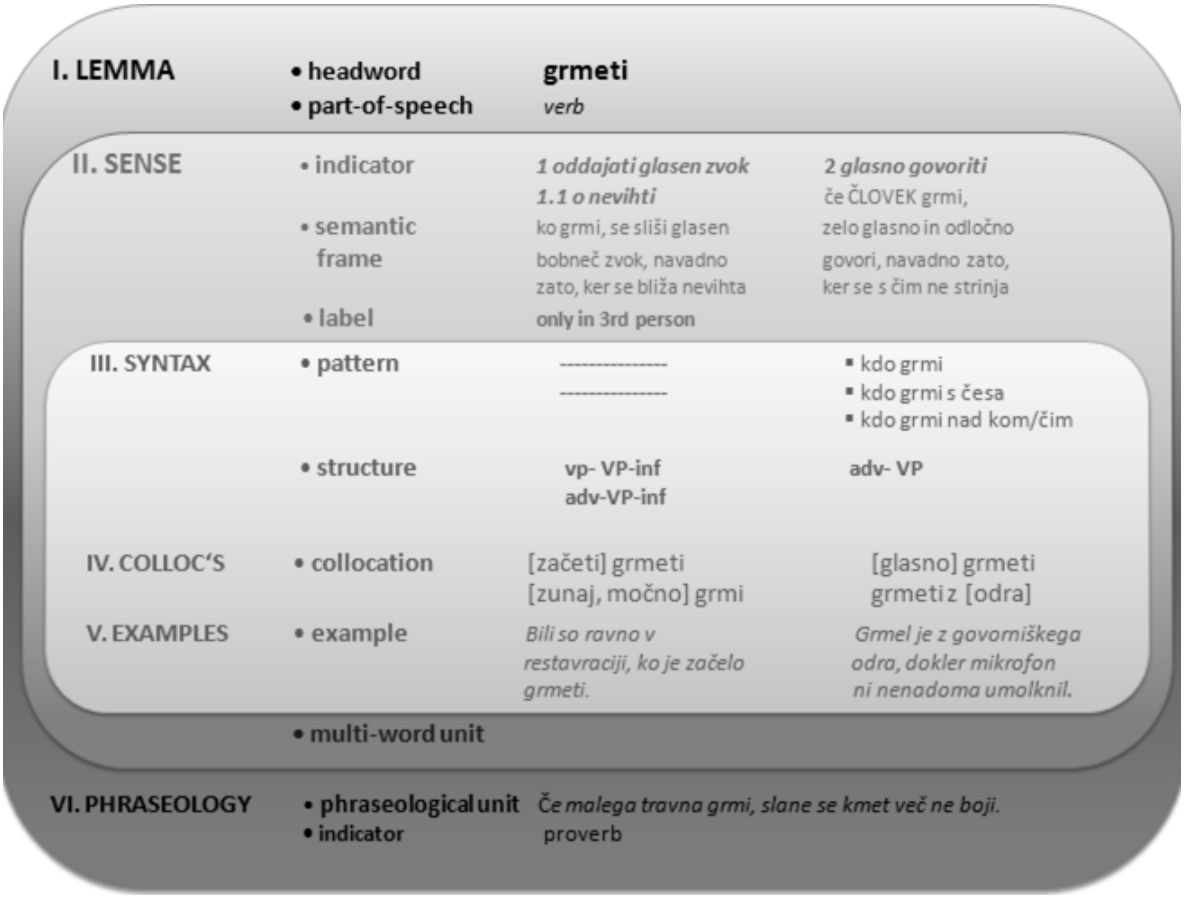

Figure 1. Structure of the Slovene lexical database

a) Multi-word expressions, which are registered only for noun or adjective headwords. Multi-word expression must demonstrate a non-compositional idiosyncratic sense.

b) Syntactic structures, representing a formalization of typical patterns on the clause and phrasal level and primarily intended for natural language processing tools.

c) Collocations and examples. On the collocation level, patterns and structures are verified by recording typical collocates of the headword realized in the anticipated syntactic positions. Collocations and its related parent levels (patterns, structures and frames with semantic types) are attested with corpus examples.

3. Compiling entries using automatic extraction of data

The decision to introduce automatic extraction of data from the corpus was made early in the process of compiling an entry, as it became obvious that there were several bottlenecks. We used the Sketch Engine (Kilgarriff et al., 2004), a leading lexicographic tool for corpus analysis, with (lexicographic) functions such as Word Sketch and TickBox Lexicography; however, the time spent on selecting under each syntactic structure the relevant collocates and their examples, and copy-pasting them into a dictionary-writing system was considered excessive.

The time-consuming nature of these tasks also had a negative effect on lexicographers' distribution of time (and effort) to different tasks. For example, for headwords with many (sub)senses and syntactic patterns, lexicographers could on average dedicate less time to identifying different (sub)senses and devising semantic frames and indicators for each (sub)sense.

3.1 Methodology

The procedure of automatic extraction provided lexical information, related to grammatical structures recorded in the lexical database, from the 1.18-billion-word Gigafida corpus of Slovene (Logar Berginc et al., 2012). The information was extracted in an XML format and imported into the iLex dictionary-writing software (Erlandsen, 2004). The relevant lexical information comprised collocations and related corpus examples. The procedure required the following:

i. a selection of lemmas for extraction,

ii. finely-grained sketch grammar, designed specifically for the purposes of automatic extraction,

iii. GDEX (Good Dictionary EXamples; Kilgarriff et al., 2008) configuration(s),

iv. an API script to extract data from word sketch information in the Sketch Engine, and

v. settings for extraction (e.g. minimum collocation frequency, minimum collocation salience).

3.2 Selecting lemmas

We wanted to focus on a group of lemmas that would enable an evaluation without the problem of large quantities of data, and that would be more homogeneous in nature as to facilitate gradual improvement of GDEX configurations and settings for extraction. Thus, lemmas had to fulfil three criteria:

a) Frequent enough to offer a good-sized word sketch. Namely, initial testing showed that word sketches for less frequent lemmas (less than 600 hits in Gigafida) did not provide enough relevant data. Consequently, we divided lemmas of each word class into five different frequency groups, and then focussed on frequency ranges that provided the best word sketches for a manageable number of lemmas.

b) Monosemous or having up to two synsets/senses in sloWNet, a Slovene version of Wordnet (Fišer, 2009), or, exceptionally, in the Dictionary of Standard Slovenian (SSKJ).

c) Found in sloWnet, preferably, but not in SSKJ, as we wanted to focus on new words and/or senses.

The final selection included 515 nouns, 260 verbs, 275 adjectives and 117 adverbs and was dominated by lemmas with frequency between 1000 (0.85 per million words) and 10,000 (8.5 per million words). There were a few lemmas with frequency below or above this range for the purposes of additional testing, especially for testing the effectiveness of the API script in extracting data for all grammatical relations in the sketch grammar.

3.3 Sketch grammar

The sketch grammar (Krek and Kilgarriff, 2006), designed specifically for automatic extraction, utilized the directives *CONSTRUCTION, *COLLOC and *SEPARATEPAGE; elements that represented new additions to the Sketch Engine at that time. The first of these three directives enables the identification of grammatical relations without collocations, which is particularly useful for extraction of verb patterns. The second directive is used to identify elements that are categorized as syntactic combinations in the lexical database, such as preposition-noun-preposition. The third directive is intended for creating a separate word sketch page for relations with three elements (directive *TRINARY), which enables the introduction of relations with prepositions that can have more specific definitions (for example they

can include the case of the preposition).[2]

directive	number of gramrels
TRINARY	36
DUAL	25
UNARY	2
CONSTRUCTION[3]	13
CONSTRUCTION+UNARY	6
COLLOC	3
SYMMETRIC	2
no directive	18
total	**105**

Table 1: Gramrels by directives

The new sketch grammar included all the structures registered in the lexical database, and therefore contains significantly more gramrels (grammatical relations) than the sketch grammar used for preparing data for manually compiled entries. There are 103 gramrels in total; categorization is shown in Table 1.

All the directives with three elements (*TRINARY) were used with a separate page output. The combination CONSTRUCTION+UNARY was used to alert the lexicographers, in a separate column called Constructions, to gramrels occurring very frequently in the corpus (this is the main function of the UNARY directive). Using this directive, we can also automatically generate alerts such as *pogosto zanikano* (often in negative), *pogosto v 3. os. ednine* (often in 3rd person singular), etc., that are recorded in <oznaka> (label) tag in the database and are candidates for labels in the dictionary.

Each gramrel in the sketch grammar contains the information about the name of the structure in the lexical database, for example:

*DUAL
=S_v_rodil-s/S_s-koga-česa

The structure used to extract combinations of a noun in any case with a noun in genitive (e.g. *delovanje motorja*, 'working of an engine' (gen.)) is recorded in the lexical database as SBZ0 sbz2, if the headword is the head noun, or as sbz0 SBZ2, if the headword is a noun in the genitive case. The relevant information is added to each gramrel:

[2] This was not possible in earlier sketch grammars as it would result in a very high number of relations/columns in the word sketch.

[3] For more on the CONSTRUCTION directive, see Rychlý (2010) and Krek (2012).

```
# LBS-XX ##########
# /1/ <struktura>SBZ0 sbz2</struktura>
# /2/ <struktura>sbz0 SBZ2</struktura>
###################
```

The sketch grammar presented above is intended solely for the purposes of automatic extraction of data from the corpus, as it produces word sketches that are difficult to process by a human user due to a high number of relations and their complex naming system.

3.4 GDEX configurations

Corpus examples are an important part of the lexical database, as they attest word senses, definitions, collocations, patterns, domain and genre-related characteristics, pragmatics, etc. According to Atkins and Rundell (2008: 458), a good corpus example should meet at least three criteria: naturalness and typicality, informativeness and understandability. However, as corpora are becoming larger and larger, it means there is more data to analyze, which is making the search for good examples more and more difficult and time-consuming.

GDEX is a tool that assists lexicographers in finding good corpus examples by ranking them according to their quality. Ranking is done on the basis of parameters such as example length, whole sentence form, syntax, and presence/absence of rare words, etc., which are measurable and in some way connected with the aforementioned criteria for a good example.

The first version of GDEX for Slovene (Kosem et al., 2011) was developed to meet the needs of lexicographers compiling manual entries in the lexical database. The existing version of GDEX for Slovene was not suitable for the purposes of automatic extraction due to differences in the relationship between computer and lexicographer. In the normal, "manual" procedure the lexicographer uses corpus tools to analyze corpus data, selects them and transfers them into dictionary-writing software. The role of GDEX was to provide at least three good examples among the ten offered in the TickBox Lexicography.

In the automatic procedure, on the other hand, the data is automatically exported from the corpus into dictionary-writing software, where they are examined, selected and edited by the lexicographer. The main aim was to reduce manual inserting of data in the database, and to reduce the need for manual removal of irrelevant or incorrect information; therefore, the aim was to design a GDEX configuration where the **top three** examples would meet the criteria of a good example.

The experience from designing the first GDEX for Slovene indicated that GDEX results could be improved by devising a separate configuration for each word class. Thus, four different GDEX configurations were prepared, for nouns, verbs, adjectives, and adverbs, respectively. All configurations contained classifiers, listed in Table 2,

but differed in settings. Initial configurations, which did not contain all the listed classifiers, were devised from the first GDEX of Slovene, with values of classifiers set by analyzing existing examples in the lexical database that were manually selected by lexicographers.

- whole sentence
- contains token with frequency of less than 3
- sentence longer than 7 tokens
- sentence shorter than 60 tokens
- lemma is repeated
- contains email address or URL
- optimum length (between X and Y tokens)
- contains rare lemmas
- contains token, longer than 12 characters
- number of punctuation marks (excluding commas)
- number of commas
- tokens starting with a capital letter
- tokens containing mixed symbols (e.g. letters and numbers)
- number of personal names
- number of pronouns
- position of lemma
- stop list of words at the beginning
- stop list of phrases at the beginning
- second collocate (collocate of a collocation)
- Levenshtein distance

Table 2: GDEX classifiers for automatic extraction

After initial configuration for each word class was devised, it was tested in the Sketch Engine by evaluating examples for a sample of lemmas from the selection that would be used in the automatic extraction. Then, values for classifiers were modified according to observations during evaluation, and a new configuration was devised. The evaluation then compared the results given by both configurations, and further modifications were made. The procedure was repeated until the GDEX configurations that provided the most satisfactory results were obtained. An important consequence of this method was the formation of several new classifiers, which were not found in the first GDEX for Slovene. Particularly noteworthy additions are stop lists of words and phrases at the beginning of examples and second collocate (collocate of a collocation). The latter classifier brought significant improvement to the results of automatic extraction because it indirectly detects colligational typicality of a collocation. For example, for the collocation *klavrn + podoba* (‘poor image’), the classifier awards points to examples with the second collocate *kazati* (‘show’), and consequently, the configuration containing this classifier offers examples containing typical structures of this collocation: *kazati klavrno podobo česa* (‘show poor image of sth’).

3.5 Preparing the API script

The API script for automatic extraction was written in Python and required certain updates to the Sketch Engine tool. Before the API script could be run, word sketch had to be created using the sketch grammar for automatic extraction. The following parameters had to be set when running the script:

- corpus
- lemma (or a list of lemmas in a file)
- gramrel (or a list of gramrels in a file)
- GDEX configuration
- number of examples per collocate
- number of collocates per grammatical relation
- minimum frequency of a collocate
- minimum frequency of a grammatical relation
- minimum salience of a collocate
- minimum salience of a grammatical relation.

An XML template for extracted data had to be prepared, and its structure matched with the DTD of the lexical database to enable importing of automatically extracted data into the dictionary-writing program. In order to make the exported data easier to view, we added attributes to <kolokacija> and <zgled> in the DTD, namely, an ID for a collocate, so that the connection between a collocate and its examples was maintained; the index number of a token in the <zgled> element, which also enables an identification of an example in the corpus; and a number for each example of a collocate, reflecting the GDEX ranking.

3.5.1 Setting the parameter values

Initial tests in automatic extraction used the following settings: 10 collocates per relation, 6 examples per collocate, minimum salience of a relation or collocate = 0, minimum frequency of a collocate = 0, and minimum frequency of a relation = 25; however, the evaluation showed that the same settings cannot be used for all the relations and collocates, since the output contained many irrelevant relations and associated collocates, or missed relevant relations and collocates. Also, the number of examples had to be reduced as editing took too long.

Initial settings were improved by obtaining the statistical data for grammatical relations and collocates, available in word sketches, of all the lemmas for automatic extraction; then, the values for each relation within lemmas of a word class were analyzed to obtain the optimal minimum frequency and salience of the relation. Also relevant was information on the percentage of the lemma occurrences in a particular relation.

The statistical analysis was combined with manual analysis of word sketches, and the finding was that if a relation covered a low percentage of occurrences of a lemma, it was often not a candidate for automatic extraction for that lemma. An additional benefit of manual analysis of word sketches was that it led to the identification of a few shortcomings in the sketch grammar (e.g. incorrectly defined or classified gramrel), which were then corrected before the final automatic extraction. Minimum frequency and salience values for collocates were determined by examining the collocates under each gramrel for each of the word classes, and identifying the lowest values where the collocation still yielded relevant results.

The analysis of data extracted using initial settings showed that the number of collocates per grammatical relation was a very important parameter. Namely, if the first ten collocates (default settings) did not exceed the minimum frequency or salience, the relation was not extracted, even if it is very frequent. As a result, the minimum number of collocates per relation was increased to 25, and the selection of relevant collocates was 'left' to minimum frequency and salience settings. The number of examples per collocate was reduced to three, as the evaluation showed that in most cases at least one of the top three examples offered by GDEX was good (in fact, often all three were good).

Another issue encountered was that in some cases an entire relation, which was frequent for a particular lemma, was not extracted because none of its collocates was above the frequency and/or salience threshold. However, this issue was mainly observed with low frequency lemmas and was solved by dividing lemmas into frequency groups, and preparing separate settings for each group.

3.6 Evaluation

In order to be able to evaluate whether using automatically extracted data is time-effective, we first finalized the entries for headwords with automatically extracted data. Then, we compared the time needed to manually devise an entry in the lexical database (i.e. selecting the relevant corpus data, mainly on the basis of analysing word sketches, transferring it into the dictionary-writing system, and adding other information), with the time needed to devise an entry using the automatic method. The results clearly favoured the approach using the automatic method: on average, using the manual method, it takes a lexicographer just over four hours to devise an entry (0.23 entries per hour), whereas using the automatic method, a lexicographer devises an entry in two hours (0.5 entries per hour). Consequently, the automatic method more than halves the time required to devise dictionary entries.

Another aim of evaluation was to identify the (lexicographic) work required to create final entries from the automatically extracted data, and to assess the reliability of the automatic method. The automatic method renders some routine tasks unnecessary, such as copying the data to a dictionary-writing system, but under the condition that

the lexicographer does not often need to consult the corpus to add missing information. The evaluation showed that the automatic method was very reliable, and extracted examples always attested for all (sub)senses of the headword. In comparison with the manual method, the entries showed differences in terms of sense division and definitions, which was expected as they were devised by different lexicographers, but the main finding was that none of the information needed to devise the entries was lost using the automatic method.

Tasks still allocated to lexicographers are of two types: analytical and editorial. Analytical tasks comprise sense division, preparing sense indicators and definitions, identification of compounds, phrases and pragmatic characteristics of meanings, and adding style and domain labels. Editorial tasks include distributing the extracted information according to the information added by lexicographers (e.g. collocates under the relevant sense), copying grammatical relations and collocates if they are typical for more than one (sub)sense, and deleting irrelevant relations, collocates and corpus examples.

The evaluation indicated that editorial tasks can sometimes still take a considerable amount of time when devising an entry. Although some can be eliminated or shortened by improving the automatic extraction method or by automating some of the steps (e.g. grouping collocates using the Thesaurus function in the Sketch Engine), these tasks are likely to remain an integral part of lexicographic work. Nonetheless, as the tasks are relatively less demanding in nature, and some are in fact very routine, we wanted to test whether they can be successfully completed by non-lexicographers (people with good knowledge of a language but without lexicographic experience), using the crowd-sourcing process.

3.6.1 Crowd-sourcing

One of the main challenges of trying to introduce crowd-sourcing into the lexicographic process was the design of procedures that would enable quick and successful completion of editorial tasks without the need for extensive learning of the concept and nature of work on the lexical database. We identified three activities that were potentially suitable for crowd-sourcing:

a) evaluating examples to identify false collocations,

b) evaluating examples to identify incorrect examples (i.e. the ones where the collocation does not match the grammatical relation it belongs to), and

c) distributing collocations and their examples under (sub)senses.

The first two activities can be conducted on automatically extracted data and should follow one another, whereas the third activity requires that the analytical work is completed first.

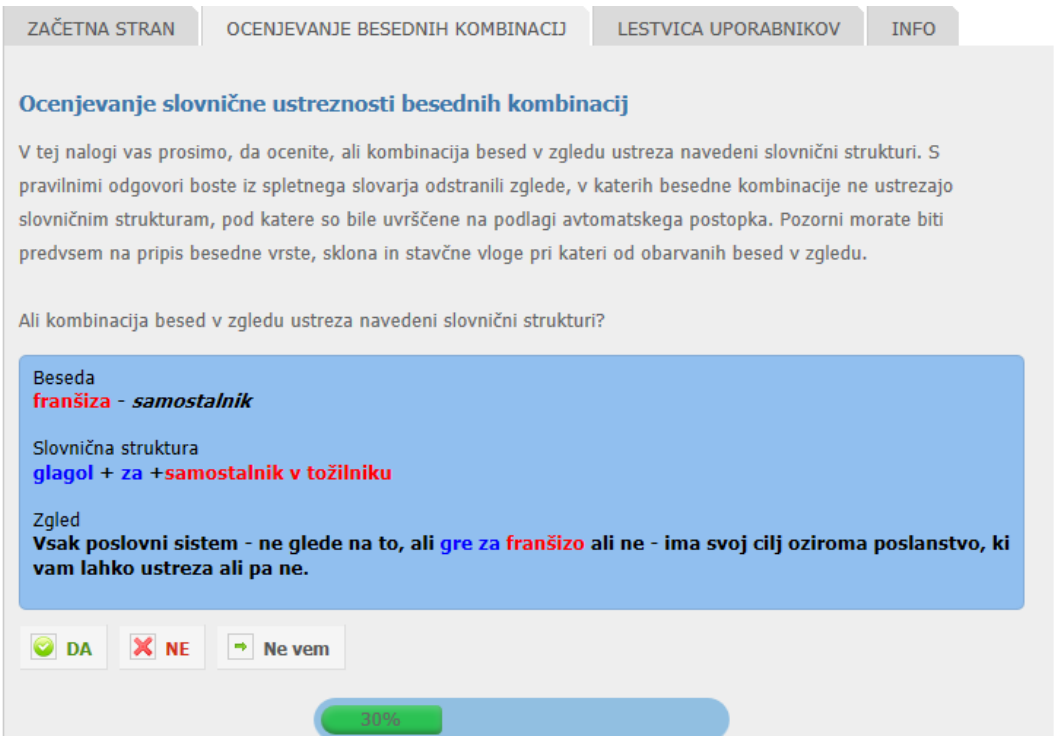

Figure 2: Evaluating examples (Task 2) in an online tool

The crowd-sourcing experiment comprised two tasks (covering activities a and b above) that were prepared in an online tool designed specifically for crowd-sourcing and was first used for checking translations in sloWNet (Tavčar et al., 2012).

In Task 1, we wanted to identify false collocations through their corpus examples. In many cases, false collocations can be identified with a great degree of certainty without even looking at corpus examples; however, we have established that it is much easier, and more reliable, for non-lexicographers to identify such collocations indirectly, i.e. by evaluating corpus examples. In Task 2, which follows Task 1, the focus is on removing incorrect examples for the remaining collocations (see Figure 2), i.e. examples that do not show the collocation correctly (e.g. do not contain the collocate in the case defined in the relation). Task 2 is more demanding than Task 1, and we provided help for the evaluators in the form of colours for different elements of a grammatical relation.

Both tasks are designed in a way that the question is asked and the data shown, and then the evaluator is offered three possible answers: YES, NO, and DON'T KNOW. For example, the question at Task 1 is: *Would you expect to find the example below in a dictionary under the entry X?* We intentionally wanted to avoid questions such as *How good do you think this example is?* that would require the evaluators to grade the example on a scale.

When preparing the data for crowd-sourcing, we decided not to include all the grammatical relations, as some were too complex for evaluation (e.g. verb

constructions, who + verb + to whom) and some often provided poor results and thus needed an improvement of their definition in the sketch grammar. For each task, we needed to provide a so-called "gold standard", a set of collocates and their examples with the answer already provided. The examples from the gold standard are then used randomly during the task to help determine the reliability of the evaluator.

The crowd-sourcing experiment is still in its early stages but initial tests have shown high reliability of crowd-sourcing data, also confirming that the tasks are designed appropriately.

4. Putting it all together in a dictionary project

The Slovene Lexical Database has, from the very beginning, been seen as a project that would provide and test new methods, and which could be used in the making of a new dictionary of Slovene. It is worth noting that the last comprehensive dictionary of Slovene (SSKJ) was published in 1991, and since that dictionary took more than 20 years to make, many of its entries were already outdated or lacked information on new meanings and usage by the time the dictionary was published. The new version of SSKJ is expected to be published in 2014; however, since it will combine old data with new information, it is bound to suffer several of the shortcomings of its predecessor. In addition, the second version of SSKJ is likely to be initially available in print format only, which is surprising given that the research shows that Slovene dictionary users, especially younger generations, rarely or almost never use printed dictionaries.

The Slovene language is in need of a completely new description that would reflect the way words and their meanings are perceived in the modern world. In addition, such a description would have to be updated regularly to meet the needs of its users; consequently, it has to exist in an online format. Such a description needs to be made available quickly, and Krek et al. (2013) prepared a proposal for a dictionary of contemporary Slovene (SSSJ) that would provide exactly that, using the methods described in this paper. The proposed dictionary envisages the use of a process of making dictionary entries in five phases:

a. Red phase: completely automatic and involves the extraction of grammatical relations, collocates and examples from the corpus.

b. Orange phase: consists of crowdsourcing activities, where incorrect or irrelevant data from the red phase are identified and excluded from the database (and the dictionary).

c. Yellow phase: the most important phase, in which lexicographers carry out all analytical tasks (e.g. sense division, identifying compounds) on the extracted data, adding missing information if needed. This phase also includes crowdsourcing for routine tasks of distributing collocates and examples under

relevant (sub)senses.

d. Blue phase: in which specialists such as terminologists and etymologists are consulted.

e. Green phase: the final editorial check is performed.

Considering the reliability demonstrated by the automatic method, SSSJ would not be offered to users after all entries are completed, but immediately after the automatic extraction of data for all entries, i.e. in the red phase. Then, entries would be updated after the completion of subsequent phases. To alert users to any changes and potential incompleteness of an entry, each entry would contain the information on the phase of the entry and the date of the last update (see Figure 3).

During the making of SSSJ, priority would be given to topical and core vocabulary, and to terminology that is becoming part of general language (even if only for a certain period). Topical vocabulary would be detected by monitoring webpages of news portals, newspapers and other resources. Moreover, new words and meanings would be added regularly, either based on corpus monitoring or on user feedback.

Figure 3: Date and stage information in the proposed dictionary of contemporary Slovene

The methods to be used in making the proposed dictionary are not new, if taken individually, as similar methods have been used in dictionary projects around the world. For example, automatic extraction has been used in the making of automatic collocation dictionaries (Kilgarriff et al., 2013); crowdsourcing, albeit in a different form, has been used by the Oxford English Dictionary, Macmillan English Dictionary, and Wordnik, etc. However, the proposal introduces a new concept of compiling a dictionary using automatically extracted data as a point of departure. Lexicographic analysis is still corpus-based (or driven); however, the initial selection of corpus data to be analyzed is left to the computer. The lexicographer then examines, validates, and completes the information and shapes it into the final dictionary entry. The benefits of using this approach for making a dictionary are particularly significant for languages where a dictionary needs to be made from scratch, and needs to be available to users almost immediately.

5. Conclusion

Lexicography is not far from making the vision of Rundell and Kilgarriff a reality. Automatization can be implemented in many aspects of lexicographers' work, saving considerable amounts of time and money. Nonetheless, some tasks, especially anything connected with meaning, remain in the domain of lexicographers, at least for now.

Our experience from preparing the Slovene Lexical Database supports these claims, but also shows that the implementation of automatic procedures calls for a different division of human work, and the introduction of a new participant to the lexicographic project. In this new division of work, lexicographers focus on more difficult, analytical tasks, whereas non-lexicographers (via crowdsourcing) are used for less demanding, more routine tasks. Such a division of work speeds up the dictionary-making process and should be particularly useful in the age of e-lexicography, when users demand immediate access to up-to-date lexicographic information.

In summary, we propose a slight revision of the approach proposed by Rundell and Kilgarriff; in our adaptation, there are three elements: a computer, a non-lexicographer and a lexicographer. The computer provides data, the non-lexicographer cleans it for the lexicographer (separating the wheat from the chaff), as well as redistributing it, and the lexicographer shapes it into the final product.

6. References

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

Baker, C.F., Fillmore, C.J. & Cronin, B. (2003). The Structure of the FrameNet Database. *International Journal of Lexicography* 16(3), pp. 281-296.

Erlandsen, J. (2004). iLex – new DWS. *Third International Workshop on Dictionary Writing systems (DWS 2004). Brno, 6. – 7. september 2004*. Available at: http://nlp.fi.muni.cz/dws2004/pres/#15.

Fillmore, C.J., Atkins, S.B.T. (1992). Towards a Frame-based organization of the lexicon: the semantics of RISK and its neighbors. In A. Lehrer, E. Kittay (eds.) *Frames, Fields, and Contrasts: New Essays in Semantics and Lexical Organization*. Hillsdale: Lawrence Erlbaum, pp. 75-102.

Fišer, D. (2009). SloWNet – slovenski semantični leksikon. In M. Stabej (eds.) *Infrastruktura slovenščine in slovenistike (Obdobja 28)*. Ljubljana: Znanstvena založba Filozofske fakultete, pp. 145–149.

Gantar, P., Krek, S. (2011). Slovene lexical database. In D. Majchraková, R. Garabík (eds.) *Natural language processing, multilinguality: sixth international conference, Modra, Slovaška, 20-21 Oktober 2011*, pp. 72-80.

Hanks, P. (2013). *Lexical Analysis: Norms and Exploitations*. Cambridge, MA: MIT Press.

Kilgarriff, A., Husak, M., Jakubicek, M. (forthcoming) *eLex 2013 Proceedings, 17-19 October 2013, Tallinn, Estonia*.

Kilgarriff, A., Husák, M., McAdam, K., Rundell, M. & Rychly, 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., Kovář, V., Rychlý, P. (2010). Tickbox Lexicography. In S. Granger, M. Paquot. *eLexicography in the 21st century: New challenges, new applications*. Brussels: Presses universitaires de Louvain, pp. 411-418.

Kilgarriff, A., Rychly, 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.

Kilgarriff, A., Tugwell, D. (2002). Sketching words. In H. Corréard (ed.) *Lexicography and Natural Language Processing: A Festschrift in Honour of B. T. S. Atkins*. Euralex, pp. 125-137.

Kosem, I., Husák, M., McCarthy, D. (2011). GDEX for Slovene. In I. Kosem, K. Kosem (eds.) *Electronic Lexicography in the 21st Century: New applications for new users, Proceedings of eLex 2011, Bled, 10-12 November 2011*. Ljubljana:

Trojina, Institute for Applied Slovene Studies, pp. 151-159.

Krek, S. (2012). *New Slovene sketch grammar for automatic extraction of lexical data*. Presented at SKEW3 workshop, 21-22 March 2012, Brno, Czech Republic. Available at: http://trac.sketchengine.co.uk/attachment/wiki/SKEW-3/Program/Krek_SKEW-3.pdf?format=raw

Krek, S., Kilgarriff, A. (2006). Slovene Word Sketches. In T. Erjavec, J. Žganec Gros (eds.) *Proceedings of the 5th Slovenian and 1st International Language Technologies Conference*. Ljubljana, Slovenia. Available at: http://nl.ijs.si/is-ltc06/proc/12_Krek.pdf.

Krek, S., Kosem, I. Gantar, P. (2013). *Predlog za izdelavo Slovarja sodobnega slovenskega jezika*, v1.1. Available at: http://www.sssj.si/datoteke/Predlog_SSSJ_v1.1.pdf

Logar Berginc, N., Grčar, M., Brakus, M., Erjavec, T., Arhar Holdt, Š., Krek, S. (2012). *Korpusi slovenskega jezika Gigafida, KRES, ccGigafida in ccKRES: gradnja, vsebina, uporaba*. Ljubljana: Trojina, zavod za uporabno slovenistiko; Fakulteta za družbene vede.

Rychlý, P. (2010). *Extensions (completed, and planned) to formalism*. Presented at Sketch Grammar Workshop, 3-4 February 2010, Faculty of Social Sciences, Ljubljana, Slovenia. Available at: http://projekt.slovenscina.eu/Media/BesedneSkice/Predstavitve/Pavel/extensions_cql_skegr.pdf.

Rundell, M., Kilgarriff, A. (2011). Automating the creation of dictionaries: where will it all end? InF. Meunier, S. De Cock, G. Gilquin, M. Paquot (eds.). *A Taste for Corpora. A tribute to Professor Sylviane Granger*. Amsterdam: Benjamins, pp. 257–281.

Sinclair, J. (ed.) (1987). *Looking up: An Account of the COBUILD Project in Lexical Computing*. London: Collins.

SSKJ: *Slovar slovenskega knjižnega jezika* (1991) Ljubljana: ZRC SAZU. Online version available at: http://bos.zrc-sazu.si/sskj.html.

Tavčar, A., Fišer, D., Erjavec T. (2012). SloWCrowd: orodje za popravljanje wordneta z izkoriščanjem moči množic. V: T. Erjavec in J. Žganec Gros (ur.) *Zbornik Osme konference Jezikovne tehnologije. Proceedings of the Eighth Language Technologies Conference.* 8. do 12. oktober 2012 / October 8th - 12th, 2012 Ljubljana, Slovenija, pp. 197–202.