THE RoWaC CORPUS AND ROMANIAN WORD SKETCHES

Monica MACOVEICIUC[*], Adam KILGARRIFF[**]

[*] Alexandru Ioan Cuza University, Iași, Romania
[**] Lexical Computing Ltd, Brighton, UK

E-mail: monica.macoveiciuc@info.uaic.ro, adam@lexmasterclass.com

Abstract: Romanian has, to date, been without a large, accessible, general-language corpus. We have created such a corpus, RoWaC, using methods pioneered in the Web-as-Corpus community. We describe the procedures we used and the resulting 50-million-word corpus. Word sketches are one-page, corpus-driven summaries of a word's grammatical and collocational behaviour. For English, they are being widely used for dictionary-making, research in linguistics and language technology, and language teaching. English word sketches were first prepared in 1999 and since then, they have been developed for a dozen other languages. They are produced by the Sketch Engine corpus software, and the inputs are a large, general-language, part-of-speech-tagged corpus and a `sketch grammar'. We describe and document Romanian word sketches based on RoWaC.

Key words: Romanian word sketches, web corpus, grammatical relations, sketch grammar.

1. INTRODUCTION

How do we study a language? A standard scientific answer might be "start by taking a sample". While this approach has been contentious, with Chomsky, in particular, making the case against, it has been gaining momentum for the last two decades. The samples are called corpora. It has been gaining momentum for a number of reasons, all related to computers. Firstly, they make it possible to handle large datasets easily. Secondly, people write on them, so it becomes easy to gather large sets of documents that are already in electronic form. And thirdly, as technology progresses, so the tools for processing, querying and finding patterns and structures in the data improve. Language technology can both make corpora richer, by contributing tools to the preparation and markup of the data, and is a customer for corpora as it needs them to test, train and evaluate systems.

Linguists and lexicographers need not only corpora, but also tools that make it easy to explore and interrogate them. As, for many purposes, corpora should be large, comprising millions or even billions of words, these tools need to be designed to handle large data. It will assist corpus users if they do not have to manage the data themselves, but this is taken care of by experts: the web makes this model viable, with corpora being queried over the web (Kilgarriff, 2010). One tool which supports fast corpus querying, even for multi-billion word corpora, is the Sketch Engine (Kilgarriff et al., 2004)[1]. The distinctive feature of the Sketch Engine is its ‘word sketches’ one-page, corpus-

[1] http://www.sketchengine.co.uk

driven summaries of a word’s grammatical and collocation behaviour. These have been in use for dictionary-writing for English since 1999 (Kilgarriff & Rundell, 2002) and were first used in the preparation of the Macmillan English Dictionary for Advanced Learners (2002). They have since been developed for twenty languages and used in a large number of linguistic and lexicographic projects.

To date, Romanian has not had a large, accessible, general-language corpus, nor has it has word sketches. In this paper we discuss the creation of RoWaC, a large corpus for Romanian, and then the work involved in setting up the Sketch Engine for Romanian. First we give an overview of web corpora, then a detailed description of the preparation of RoWaC, then an overview of the Sketch Engine and of the sketch grammar for Romanian.

2. CORPORA FROM THE WEB AND CORPORA FOR ROMANIAN

Corpus collection used to be long, slow and expensive - but then came the web: texts, in vast number, are now available by mouse-click. The prospects of web as corpus were first explored in the late 1990s by Resnik (1999) and Jones and Ghani (2000). Grefenstette and Nioche (2000) showed just how much data was available for various languages. Keller and Lapata (2003) established the validity of web corpora by comparing models of human response times for collocations drawn from web frequencies with models drawn from traditional-corpus frequencies. They showed that they compared well.

In 2004 Baroni and Bernardini presented BootCaT, a toolkit for preparing ‘instant corpora’ for a sublanguage from the web by

- inputting some ‘seed terms’ from the domain
- sending the seed terms, three at a time, to one of the main search engines (Google, Yahoo, more recently Bing)
- collecting the pages referenced in the search hits page.

The output of this process then needed filtering and de-duplicating.

Sharoff (2006) has prepared web corpora, typically of around 100 million words, for ten major world languages, primarily for use in teaching translation. Scannell (2007) has gathered small corpora (in most cases less than a million words) for several hundred languages. Baroni et al. (2009) describe DeWaC, ItWaC and UKWaC, each of between 1.5 and 2 billion words: how they were gathered, cleaned and evaluated. Kilgarriff et al. (2010) describe a ‘corpus factory’ for preparing web corpora for a growing list of languages.

While it is possible to use the web as a corpus with Google, Yahoo or Bing as the interface, and no intermediate step of corpus-gathering, there are numerous disadvantages to this approach, as documented in Kilgarriff (2007).

The most important collection of corpora for Romanian has been created at RACAI (Cristea & Forăscu, 2006). Most of them have homogeneous content. They are either based on individual texts (George Orwell's '1984', Plato's Republic), newspapers (Evenimentul Zilei - 92,000 words, ROCO - 7 million words), or they are the Romanian version of some already existing corpus:

- Romanian FrameNet: 1,094 sentences from the original FrameNet 1.1 corpus;
- RomanianTimeBank: 186 news articles, with 72,000 words, translated from TimeBank 1.1;
- RoSemCor: 12 articles from SemCor;
- Acquis Communautaire:12,000 Romanian documents and 6,256 parallel English-Romanian documents, with 16 million words.

Prior to the work reported here, there was no large, accessible, general-language corpus for Romanian.

3. CORPUS CREATION AND ANNOTATION

The Romanian corpus (RoWaC) was gathered from the web using web crawling, BootCaT, a newspaper archive and a site for copyright-free books. The corpus contains 50 million words, distributed as shown in Table 1.

Table 1: RoWaC sources

Source	**Size in tokens (words+punctuation)**	**Percentage**
WebBootCaT	20,625,141	38.6
Heritrix	12,740,859	23.8
www.adevarul.ro	1,351,847	2.5
www.biblioteca-online.ro	18,739,675	35.1
Total	**53,457,522**	**100.0**

3.1. Web crawling with Heritrix

We used Heritrix for web crawling. It was designed for web archiving and can gather huge amounts of text fast. Starting from an URL, it access the links encountered, downloads the pages, cleans them and stores them in .arc files.

The URL chosen for Heritrix was the homepage of a Romanian news portal (www.realitatea.net). The content was extracted using the ArcReader tool from Internet Archive, and the resulting files ranged between 100 and 600 MB. One problem occurred: even though Heritrix contains mechanisms for extracting only text from the web pages, the results were not perfect. Everything that was not useful text - HTML tags, JavaScript code, comments, URLs - needed to be removed. This step was accomplished by passing the text through a Perl script which applied various regular-expression-based filters.

3.2. BootCaT procedures using WebBootCaT

WebBootCaT is an implementation of the BootCaT procedure described above (Pomikalek et al., 2006). We used WebBootCaT with words from each of the following 26 areas as seeds:

```
Banking, Cars, Chemistry, Culture, Dogs, Economy, Education,
Elections, Fishing, Journal, Library, Literature, Local News,
Mountain Trips, National News, Pamphlet, Philosophy, Planes,
Politics, Public Events, Real Estate, Robots, Sports, Stock
Exchange, TV Shows
```

The seeds were selected by the first author. The list for `banking` (with phrases in quotation marks) was

```
   "cont de economii" "transfer bancar" comision numerar
bancomat credit depozit
```

There were between seven and ten seeds for each category. WebBootCaT searches for pages using combinations of these words. Using the default settings of WebBootCaT, combinations of three words are sent to the search engine and a maximum of ten URLs are retrieved per query. Replacing one of the words, for example *comision* with *balanță*, the results were often quite different. Although *balanță* is a frequent word in the banking field, the following tuples returned no results:

```
balanță bancomat depozit
"cont de economii" "transfer bancar" balanță
"transfer bancar" balanță credit
balanță depozit numerar
```

We found that Yahoo returned no results for these queries whereas Google returned large numbers. We were using Yahoo owing to its more flexible terms and conditions. In the future we intend to explore the strengths and weaknesses of different search engines in relation to Romanian.

Each of the 26 corpora gathered with WebBootCaT contains between 400 000 and 1.5 million words.

3.3. Books and newspapers

Adevarul.ro is one of the most popular online newspapers in Romania. It includes 36 local editions, for the most important cities. An archive of local, social and political articles from Iaşi, written between December 2008 and June 2009, was added to RoWaC. It represents only 2.5% of the text, but it is valuable since it is a clean corpus, a good sample of the current state of the Romanian language.

Biblioteca-online.ro is an online collection of free books, donated by the authors. It contains, mostly, novels and studies of contemporary authors. The corpus includes 57 books from this collection, representing 35% of the corpus.

3.4. Linguistic processing

Next, the text was part-of-speech tagged and lemmatized using TTL (Tokenizing, Tagging and Lemmatizing free running texts), developed by RACAI (Tufiș et al., 2008, 2010, this volume).

Standard Romanian uses diacritics. However much of the text on the web does not conform to the standard. This was the most difficult problem to deal with, and it is not completely solved in this first version of the corpus. We used TTL to address the issue: it has a first phase of processing which adds missing diacritics back in, disambiguating between several possible word forms that may or may not contain diacritics where necessary. Naturally, this process is not 100% accurate.

Other TTL functions are Named Entity Recognition, sentence splitting, tokenization, POS tagging, lemmatization and chunking.

- The Named Entity Recognition function, written in Perl, uses regular expressions to identify sequences of tokens that constitute named entities (names of persons, numbers, dates, times etc.). This function needs to be applied prior to the sentence splitting one, so that the punctuation marks that constitute parts of a name are not be mistaken for sentence markers.
- POS-tagging is based on Hidden Markov Models technology, described in Brants (2000), with some supplementary heuristics for unknown words and 'tiered tagging' (Ceaușu, 2006), a technique that first uses intermediary tagging with a reduced tagset, and then a further phase to replace the reduced tags with full tags.
- Chunking is implemented using regular expressions over POS-tag sequences.
- Lemmatization is lexicon-based. A statistical module, which automatically learns normalization rules from the existing lexical stock, is used for solving the out-of-lexicon cases.

TTL is provided as a web service which incorporates all of these functions. We invoked it through a small Java application. The text was split into small files which were then sent to TTL. The application received the annotated text and stored it in .txt files that were merged into a single file.

3.5. Tagset

We use the tagset developed in MULTEXT-East, an EU Project for developing standardised language resources for Central and East European languages (Bulgarian, Czech, Estonian, Hungarian, Romanian, Slovene). Since the project started in 1997, many versions and languages have been added; the latest version was released in May 2010.

The standard describes the minimal encoding level that a corpus must achieve in order to be considered standardized, and provides encoding conventions. A corpus of parallel and comparable texts and word-form lexicons are among the resources developed through the project.

A morpho-syntactic description (MSD) consists of a sequence of characters. Each position in this string corresponds to an attribute; for each attribute, several one-character values are defined. The positions are numbered as 0, 1, 2, etc. The first (0) identifies the part of speech; all others encode the value of an attribute: for a noun, they hold the values of the type, gender, number etc. If an attribute does not apply, its corresponding position is replaced with the '-' character.

For a full account, see Erjavec (2010, this volume).

3.6. Loading into the Sketch Engine

Loading the corpus into the Sketch Engine required a conversion of the data into the format specified by the Sketch Engine. The Sketch Engine input format, often called "vertical" or "word-per-line", is as defined at the University of Stuttgart in the 1990s and widely used in the corpus linguistics community. Each token (eg, word or punctuation mark) is on a separate line and where there are associated fields of information, typically the lemma and POS-tag, they are included in tab-separated fields. Structural information, such as document beginnings and ends, sentence and paragraph markup, and meta-information such as the author, title and date of the document, its region and its text type, are presented in XML-like form on separate lines. A short perl script converted from TTL output to Sketch Engine input format, (including converting the original Windows character encoding to Unicode). The Romanian sentence:

```
Alegerile europene de la sfârșitul săptămânii trecute s-au
lăsat cu râs sau plâns pentru partidele politice din România
```

was now represented as:

```
Alegerile            Ncfpry              alegere
europene             Afpfp-n             european
de_la                Spca                de_la
sfârșitul            Ncmsry              sfârșit
săptămânii           Ncfsoy              săptămână
trecute              Afpfson             trecut
s-                   Px3--a--y-----w     sine
au                   Va--3p              avea
lăsat                Vmp--sm             lăsa
cu                   Spsa                cu
râs                  Ncms-n              râs
sau                  Ccssp               sau
plâns                Vmp--sm             plânge
pentru               Spsa                pentru
partidele_politice   Ncfpry              partid_politic
din                  Spsa                din
România              Np                  România
.                    .                   .
```

The final version of the corpus is stored in 32 plain text files in vertical format with sizes between 1 and 84 MB.

4. THE SKETCH ENGINE FOR ROMANIAN

The Sketch Engine is a corpus query system with standard corpus query functions such as concordancing, sorting, filtering, and also word sketches, one page summaries of a word's grammatical and collocational behaviour. The Sketch Engine also produces a distributional thesaurus for the language, in which words sharing the same collocates are put together, and sketch differences, which specify similarities and differences between near-synonyms. The system is implemented in C++ and Python and designed for use over the web.

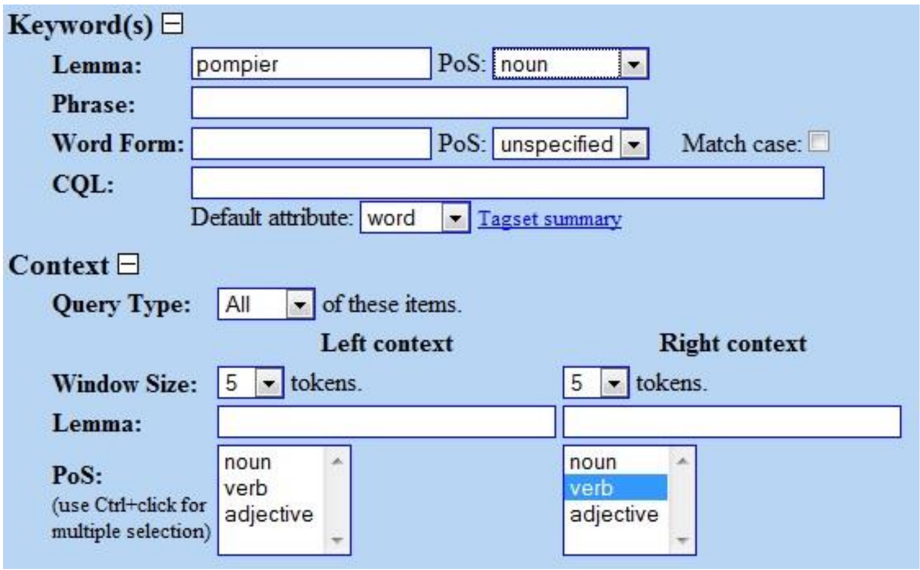
Below we describe the various features of the Sketch Engine in relation to Romanian. We focus on the concordance function, the word lists and the word sketches.

4.1. Concordance functions

Once the corpus was loaded into the Sketch Engine, the concordance functions were available. The linguist could immediately use the search boxes provided, searching, for example, for a lemma specifying its part of speech. This search is case-sensitive as generally lemmas starting with uppercase need to be distinguished from those starting with lower case. For instance, the lemma *Pădurar* is not the same as the lemma *pădurar*. The former is a proper name, whereas the latter is a common noun meaning 'forester'.

We must note here that the quality of the output of the system depends heavily on the input, i.e. the quality of tagging and lemmatisation, which is not as accurate as one might wish. Errors in lemmatisation and tagging have a substantial impact and lead to unexpected results for the user.

A wide range of search options are offered by using the 'Context' section. Here the linguist can specify the left and/or right context of the search word, with a window of up to ten items on either side. Thus a linguist editing the lemma *pompier* (*fireman*) may wish to see which verbs can follow this lemma. To this end, *pompier* needs to be typed in the lemma box and 'verb' needs to be selected as the part of speech of the right context, as shown in Figure 1.



Keyword(s) ⊟

Lemma: pompier PoS: noun

Phrase:

Word Form: PoS: unspecified Match case: □

CQL:

Default attribute: word Tagset summary

Context ⊟

Query Type: All of these items.

	Left context	Right context
Window Size:	5 tokens.	5 tokens.
Lemma:		
PoS: (use Ctrl+click for multiple selection)	noun verb adjective	noun verb (selected) adjective

Fig. 1. Concordance form, context section

On the results page the concordances are shown using KWIC view. Under ‘View options’ it is possible to change the concordance view to a number of alternative views. One is to view additional attributes such as POS tags or lemma alongside each word. This can be useful for finding out why an unexpected corpus line has matched a query, as the cause could be an incorrect POS-tag or lemma.

Some corpus sentences make good examples for the word, phrase or grammatical construction, but others do not. Perhaps they are too long, or too short, or are not well-formed sentences, or contain obscure words or spelling mistakes or abbreviations or strange characters. To find a good example is a high-level linguistic skill. But to rule out lots of bad sentences is easy, and the computer can help by doing this groundwork. The GDEX (Good Dictionary Example eXtractor) function was added to the Sketch Engine in 2008 (Kilgarriff et al., 2008). This takes the first 200 (by default) sentences matching a query, scores them according to how good an example the computer thinks they will make, and returns them in order, best first. The scoring is done with a series of simple rules addressing the considerations listed above: how long is the sentence; does it contain words outside core Romanian vocabulary; does it begin with a capital letter and end with a full stop, exclamation mark or question mark; does it contain an excessive number of characters other than lower-case a-to-z? The goal is that the average number of corpus lines that a linguist has to read, before finding one suitable to use or adapt for the dictionary entry, is substantially reduced, so they rarely have to look beyond the first ten whereas without GDEX, they may often have had to look through thirty or forty.

The GDEX rules were prepared for English. To date only minimal customisation has taken place for Romanian, replacing an English wordlist with a Romanian one taken from RoWaC.

4.2. Word Lists

The word list function offers the linguist three options, namely the creation of a word list, finding keywords which are characteristic of a particular subcorpus and finding words that are most 'X', as described below.

Creating a word list

The first option allows the linguist to create a word list. It is useful for many purposes including detecting compounds in Romanian as regular expressions can be used in the search box. Lists can be prepared for the whole corpus or for a particular subcorpus.

Keywords

The second option, KEYWORDS, allows the lexicographer to find keywords that are characteristic for a particular language variety or subcorpus. As RoWaC contains subcorpora from different fields, it was possible to generate lists of keywords for each of them. Given two subcorpora as input, one containing literary analysis, and the other, politics news, two lists of keywords were generated. For politics, the keyword list contains

> președinte (president), echipă (team), problemă (problem), partid (political party), guvern (govern), ministru (minister).

Keywords for literary analysis are:

> poet (poet), poezie (poem), operă (literary production), autor (author), moarte (death), formă (form).

FindX

The Sketch Engine FindX functionality allows us to ‘find the words that are most X’, where 'X' may be any of a wide range of characteristics. Thus, a linguist can now find an answer to questions such as which verbs characteristically display a particular complementation pattern, or which nouns have the greatest tendency to be used in the plural, in addition to which words are distinctive of a particular domain or genre, as covered in the previous section.

For linguists this is useful information as we often want to know which words are good exemplars of a phenomenon, and how they stand in relation to the whole population of words. Consider a characterisation of the nouns that are very often plural. Even if the right corpus, with the right markup, is available, it is still a programming task to do the counting, compute the statistics, sort the list, and make the results conveniently accessible. The Sketch Engine provides this facility and for Romanian a list of nouns which are most often used in plural has been generated. An extract of the resulting list (excluding the 'always plural' nouns, whose behavior is already well-known) is shown in Table 2.

Table 2: List of nouns which are most plurals

Lemma	**Frequency**	**Ratio**
ban	13971	93.8
minut	12836	80.2
stea	8521	73.6
trăsătură	3159	83.7
instrucțiune	1439	94.1
algă	1220	96
aliment	1139	90.9
implicație	620	87.9
legumă	601	87.7
pleată	548	98.4

4.3. Word Sketches

As noted above, word sketches are one-page automatic, corpus-based summaries of a word's grammatical and collocational behavior. Word sketches improve on standard collocation lists by using a grammar and parser to find collocates in specific grammatical relations, and then producing one list of subjects, one of objects, etc. rather than a single grammatically blind list.

In order to identify a word's grammatical and collocational behaviour, the Sketch Engine needs to know how to find words connected by a grammatical relation. For this to work, the input corpus needs to be parsed or at least POS tagged. If the corpus is parsed, information about grammatical relations between words is already embedded in the corpus and the Sketch Engine can use this information directly. If the corpus is POS-tagged but not parsed, grammatical relations can be defined by the developer within the Sketch Engine.

In this model, grammatical relations are defined as regular expressions over POS-tags. For example, a grammatical relation specifying the relation between a noun and a premodifying adjective may look like this.

```
=adj+SUBST[2]
2:"A.*"  1:"N.*"
```

[2] For Romanian, we adopted the following naming convention in the grammatical relations: the word corresponding to the keyword is written in upper case, whereas the one corresponding to the collocate is written in lower case.

The first line, following the =, gives the name of the grammatical relation. The 1: and 2: mark the words to be extracted as first argument (the keyword) and second argument (the collocate).

A more complex rule defines the relation between a direct object and its verb.

```
    =COMPL_DIR+verb / VERB+compl_dir
    [tag = "[^R].*"] prec_V? 2:[tag = "Vm[ism].[^3].*" & lemma !=
"fi" & lemma != "rămâne"] [tag = “T.*”]? 1:[tag = "N...[o-].*" |
tag = "R.*"]

    define(`prec_V', `([tag = "Q.*" | tag = "Va.*" & lemma !=
"fi"])')
```

The pattern contains an adverb, potentially followed by a ‘prec_V’ sequence. The verb we need is a *main* one (*Vm*), having the indicative (*i*), subjunctive(*s*) or imperative(*m*) form. One of the frequent POS tagging errors is the incorrect Type and VForm of the verb *a fi* (*to be*). The nominal predicate (*to be* + *noun*) can be easily mistaken for *verbal predicate* + *noun*. In order to avoid this confusion, we specify that the lemma we are looking for should not be one of *fi* or *rămâne* (the verbs marking a nominal predicate in Romanian). A noun in the oblique case or an adverb, preceded or not by an article, completes the pattern. prec_V is defined as a macro. It matches the situations in which the main verb is preceded by a particle or an auxiliary verb.

The result is a regular expression grammar which we call a Sketch Grammar. It allows the system to automatically identify possible relations of words to the keyword. These grammars are of course less than perfect, but given the errors in the POS-tagging, this is inevitable however good the grammar. The problem of noise is mitigated by the statistical filtering, to find only recurring collocates, which is central to the preparation of word sketches.

4.4. Romanian Sketch Grammar

Romanian is a relatively free word order language. There are no general rules for placing the subject and the predicate in a sentence. The subject can precede the predicate just as the predicate can precede the subject. Moreover, there is no markup difference between the Nominative and Accusative (they are both seen as Direct) cases of a noun, generally corresponding to the Subject and Object parts of a sentence.

We have focused on the behavior of the noun, the verb and the adjective. The sketch grammar contains 25 relations, which fall into four classes, *symmetric, dual, trinary* and *unary*, depending on whether the relation is symmetric between two collocates, or defines complementary relations between them, or is a relation between three collocates, or is a fact about the keyword. They are presented in tables 3 to 6 below.

Symmetric relations are relations between two items of equal status such as coordinate structures with conjunctions *și* (*and*) and *sau* (*or*) or with a comma. Two symmetric relations have been defined for Romanian as given in Table 3.

Table 3: Symmetric Relations for Romanian sketch grammar

Relation	**Example**	**Triple**
AND_OR și_sau	foc și apă *fire and water*	<și_sau, foc, apă>
SYMMETRY simetrie	și femeile, și fetele *women, as well as girls*	<simetrie, femeie, fată>

Dual relations are relations between two dependent items. They are the most common. They work similarly to symmetric relations but inversing a dual relation results in a different grammatical relation, whereas symmetric relations do not give rise to separate inverse relations. A typical dual is the pair, "subject for the verbal predicate" and "verbal predicate for the subject". There are twelve inverse relations in the Romanian sketch grammar. The two names are separated by a forward slash '/' in the table below.

Table 4: Dual Relations for Romanian sketch grammar

Relation	**Example**	**Triple**
SUBJECT+VERBAL PREDICATE SUBJ+pred vb / PRED VB+subj	pasărea zboară *the bird flies*	<SUBJ+pred_vb, pasăre, zbura>
SUBJECT+NOMINAL_PREDICATE SUBJ+pred_nom / PRED_NOM+subj	copacii sunt sacri *the trees are sacred*	<SUBJ+pred_nom, copac, fi>
MODIFIER_ADJ ATR+subst / SUBST+atr_adj	femeie frumoasă *beautiful woman*	<ATR+subst, frumos,femeie>
MODIFIER_NOUN SUBST+atr_subst / ATR_SUBST+subst	statuie de marmură *marble statue*	<SUBST+atr_subst, statuie, marmură>
POSSESSED SUBST_pos / POS_subst	părinții copilului *the child's parents*	<SUBST_pos, copil, părinte>
DIRECT_OBJECT COMPL_DIR+verb / VERB+compl_dir	conduce o mașină *drives a car*	<COMPL_DIR+verb, mașină, conduce>
INDIRECT_OBJECT COMPL_IND+verb / VERB+compl_ind	îi spune doctorului *he tells the doctor*	<COMPL_IND+verb, doctor, spune>
COMPLEMENT_CIRCUMSTANCE COMPL_CIRC+marc/MARC+compl_circ	de-a_lungul cărării *along the path*	<COMPL_CIRC+marc, cărare, de-a_lungul>
ADJ_MODIFIER_ADJ MODIF+adj / ADJ+modif	incredibil de complex *incredibly complex*	<MODIF+adj, incredibil, complex>
PREDICATE_AUXILIARY_VERB PRED+verb_aux / VERB_AUX+pred	va aștepta *he will wait*	<PRED+verb_aux, vrea, aștepta>
NOMINAL_PREDICATE_WITH PRED_NOM+np/NP+pred_nom	camera este curată *the room is clean*	<PRED_NOM+np, fi, curat>

Trinary relations describe relations between three dependent items. In the Romanian sketch grammar, they are mainly used for identifying prepositional patterns. A separate relation is generated for each preposition, as it is exemplified in Table 5. For instance, we can find combinations of a verb and all the possible prepositions, followed by a noun.

Given the keyword *călători* (*to travel*), a first preposition is considered – *cu* (*by*). We can identify the situations: "călătorește cu trenul" ("travels by train"), "călătorește cu mașina" ("travels by car") etc. Then, the next preposition is taken – *spre* (*towards*), and new situations are observed: "călătorește spre casă" ("travels towards home"), "călătorește spre oraș" ("travels towards the city").

Table 5: Trinary Relations for Romanian sketch grammar

Relation	Example	Triple
NOUN_PRECEEDED_BY SUBST_PREC_%s	capul pe masă *head on the table*	<SUBST_PREC_pe, masă, cap>
NOUN_FOLLOWED_BY SUBST_URM_%s	casă în copac *tree house*	<SUBST_URM_în, casă, copac>
VERB_PRECEEDED_BY VERB_PREC_%s	începea de întreba *began to ask*	<VERB_PREC_de, întreba, începe>
VERB_FOLLOWED_BY VERB_URM_%s	zbura printre stele *he was flying among the stars*	<VERB_URM_printre, zbura, stea>
ADJ_PRECEEDED_BY ADJ_PREC_%s	atât de deștept *so smart*	<ADJ_PREC_de, deștept, atât>
ADJ_FOLLOWED_BY ADJ_URM_%s	frumoasă de pică *very beautiful*	<ADJ_URM_de, frumos, pică>

Finally, unary relations can be defined. They are used to extract certain complementation patterns. For instance, a linguist would like to know that a noun is frequently followed by a series of adjectives or that a noun is preceded by an article or not. The Romanian sketch grammar contains two unary relations, as shown in Table 6.

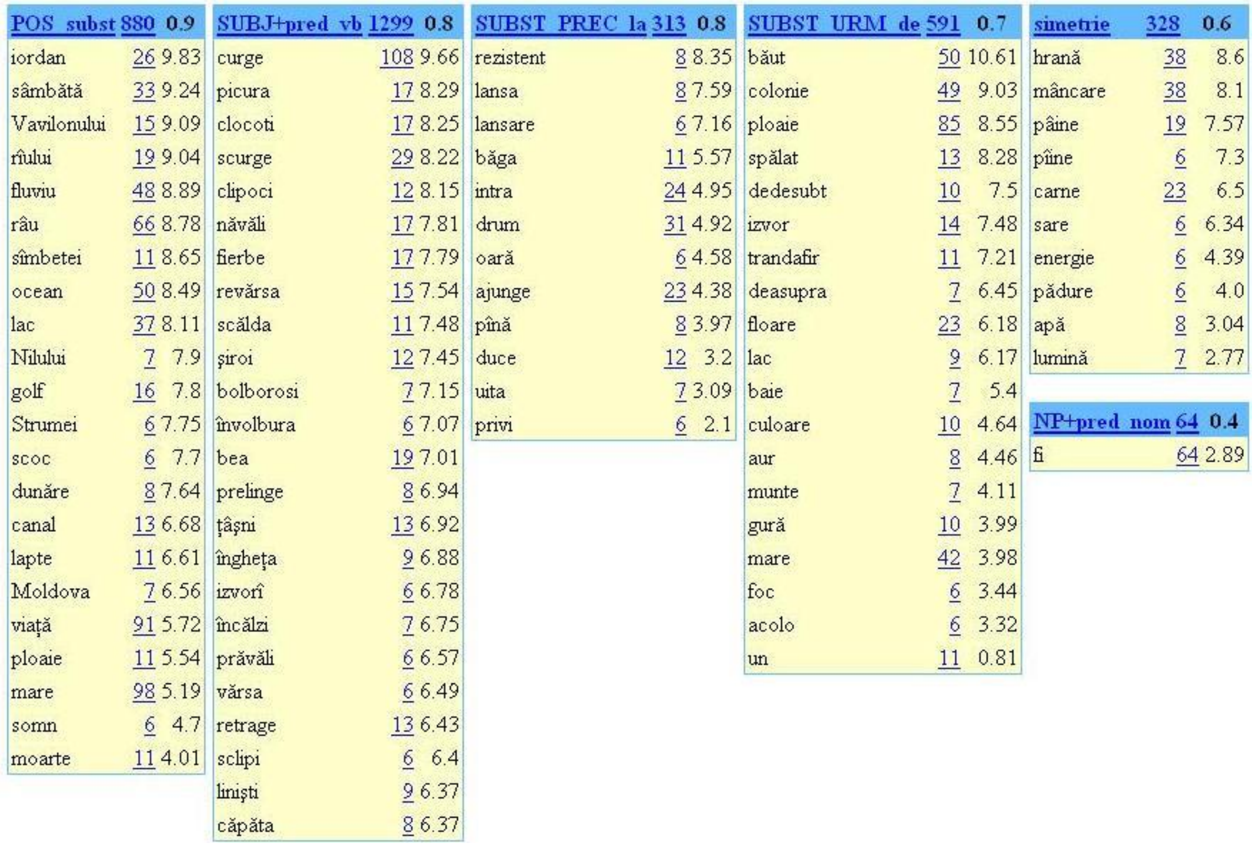
Table 6: Unary Relations for Romanian sketch grammar

Relation	Example	Triple
NOUN_ADJS SUBST+listă_adj	femeie înaltă, slabă, bătrână *tall and thin old woman*	<SUBST+lista_adj, femeie>
NOUN_SERIES serie_subst	scaunul, patul, zidurile *the chair, the bed, the walls*	<serie_subst, scaun>

Figure 2 shows part of a word sketch for the noun *apă* (*water*). Under the column atr+SUBST we find typical qualifying adjectives, denoting kinds of water distinguished by their properties or origin, e.g. *apă caldă* (*warm water*), *apă rece* (*cold water*). Idioms such as *apă sărată* (*salt water*) and *apă dulce* (*river water*) are revealed. The noun *ploaie* (*rain*) has an idiomatic use in the combination *apă de ploaie* (*not serious, unimportant*).

A key role of word sketches is to help linguists and lexicographers draw up a lexical entry for the dictionary without missing out senses of the word or idiomatic uses. In the word sketch for *apă*, we note that there are collocates relating to different uses of the noun water. For instance, the collocate *potabil* (*drinkable*) relates to the sense of water as "a drink, satisfying thirst"; *gaz* (*gas*) and *electricitate* (*electricity*) relate to the sense "supplied for domestic needs"; *teritorial* (*territorial*), *maree* ("*tide coming in*") refer to the liquid of which seas, lakes, and rivers are composed.

The user can set various preferences for the display of the word sketches. Collocates can be ranked according to the frequency of the collocation, or according to its salience score (see Rychly 2008 for the formula used to compute salience). The user can set a frequency threshold so low-frequency collocations are not shown. On the results screen the user can go to the related concordance by clicking on the number next to the lemma which refers to the number of instances. There is also a button which allows the user to show more or less data on the screen.



POS subst	880	0.9
iordan	26	9.83
sâmbătă	33	9.24
Vavilonului	15	9.09
rîului	19	9.04
fluviu	48	8.89
râu	66	8.78
sîmbetei	11	8.65
ocean	50	8.49
lac	37	8.11
Nilului	7	7.9
golf	16	7.8
Strumei	6	7.75
scoc	6	7.7
dunăre	8	7.64
canal	13	6.68
lapte	11	6.61
Moldova	7	6.56
viață	91	5.72
ploaie	11	5.54
mare	98	5.19
somn	6	4.7
moarte	11	4.01

SUBJ+pred vb	1299	0.8
curge	108	9.66
picura	17	8.29
clocoti	17	8.25
scurge	29	8.22
clipoci	12	8.15
năvăli	17	7.81
fierbe	17	7.79
revărsa	15	7.54
scălda	11	7.48
şiroi	12	7.45
bolborosi	7	7.15
învolbura	6	7.07
bea	19	7.01
prelinge	8	6.94
țâşni	13	6.92
îngheța	9	6.88
izvorî	6	6.78
încălzi	7	6.75
prăvăli	6	6.57
vărsa	6	6.49
retrage	13	6.43
sclipi	6	6.4
linişti	9	6.37
căpăta	8	6.37

SUBST PREC la	313	0.8
rezistent	8	8.35
lansa	8	7.59
lansare	6	7.16
băga	11	5.57
intra	24	4.95
drum	31	4.92
oară	6	4.58
ajunge	23	4.38
pînă	8	3.97
duce	12	3.2
uita	7	3.09
privi	6	2.1

SUBST URM de	591	0.7
băut	50	10.61
colonie	49	9.03
ploaie	85	8.55
spălat	13	8.28
dedesubt	10	7.5
izvor	14	7.48
trandafir	11	7.21
deasupra	7	6.45
floare	23	6.18
lac	9	6.17
baie	7	5.4
culoare	10	4.64
aur	8	4.46
munte	7	4.11
gură	10	3.99
mare	42	3.98
foc	6	3.44
acolo	6	3.32
un	11	0.81

simetrie	328	0.6
hrană	38	8.6
mâncare	38	8.1
pâine	19	7.57
pîine	6	7.3
carne	23	6.5
sare	6	6.34
energie	6	4.39
pădure	6	4.0
apă	8	3.04
lumină	7	2.77

NP+pred nom	64	0.4
fi	64	2.89

Fig. 2. Word Sketches for Romanian noun *apă* (*water*)

4.5. Thesaurus and Sketch Differences

Once the corpus has been parsed and the tuples extracted, we have a very rich database that can be used in a variety of ways.

Table 7: Thesaurus output for *mic* (adj). RoWaC frequency = 34469

Lemma	Score	Frequency
Mare	0.319	86661
singur	0.289	44862
Nou	0.274	27588
uriaș	0.258	77451
Vechi	0.258	16174
Imens	0.237	4802
Frumos	0.237	16175
Lung	0.233	17424
Adevărat	0.231	19924
Simplu	0.231	13056
Înalt	0.222	13003
Plin	0.219	19129
Negru	0.219	19617
Puternic	0.218	16120
Întreg	0.212	23496
Alb	0.212	17327
Ciudat	0.206	11294

We can ask "which words share most tuples", in the sense that, if the database includes both <direct_object, bea, vin> and <direct_object, a bea, apă>, then we can say that *wine* (*vin*) and *water* (*apă*) share a triple, namely there are both the object of the verb *a bea* (*to drink*). A shared triple is a small piece of evidence that two words are similar. Now, if we go through the whole lexicon, asking, for each pair of words, how many triples they share, we can build a "distributional thesaurus", which, for each word, lists the words most similar to it (in an approach pioneered in Grefenstette (1994) and Lin (1998)). The Sketch Engine computes such a thesaurus. Table 7 presents an extract of the thesaurus entry for the adjective *mic* (*small*). One notable fact is that the word with the highest score (*mare* / *big*) is the exact antonym of *mic*. As Justeson and Katz (1991) observed, antonymous words tend to share many contexts and collocations, often occurring in the two parts of parallel constructions.

Another question we are well-placed to answer is: how do near-synonyms (or other pairs of similar words) differ? For this we compare the word sketches of the two words to prepare a "sketch diff", which shows the collocates that the two words have in common and those that that are distinctive of each but do not occur with the other. The adjectives *good* and *bad* were chosen.

Table 8: Sketch Difference of adjectives *bun* (freq=44541) and rău (freq=26182)

Shared patterns			***bun*-only patterns**			***rău*-only patterns**		
ATR+subst	**bun-freq**	**rău-freq**	**ATR+subst**	**num**	**sal**	**ATR+subst**	**num**	**sal**
dispoziție	388	7	Rămas	755	9.8	duh	104	9.1
Noapte	638	10	Prieten	640	9.3	piază	32	7.9
Idée	427	79	Voie	530	9.2	presimțire	27	7.5
intenție	166	76	Seamă	745	9.1	prevestire	17	6.9
Veste	276	69	simț	381	8.7	cuget	17	6.8
Lucru	703	247	Bucată	310	8.1	sânge	48	6.8
Gând	103	162	Venit	195	7.8	vis	40	6.8
Om	867	275	Rezultat	149	7.3	**simetrie**	**num**	**sal**
Vreme	233	103	Seară	157	7.2	bine	247	8.1
Aspect	113	217	înțelegere	120	7.1	urgisit	6	7.8
Caz	9	158	dimineață	134	7.0	bună	7	7.4
și_sau	**bun-freq**	**rău-freq**	**Simetrie**	**num**	**sal**	urât	9	6.8
Rău	257	12	Rău	241	8.8	**și_sau**	**num**	**sal**
Bun	18	257	Credincios	21	8.0	urgisit	6	8.3
Prost	18	8	Bland	25	8.0	urât	7	6.5
Mare	9	8	înțelept	20	7.7	galben	6	5.2
GRAD_COMP +part	**bun-freq**	**rău-freq**	Frumos	64	7.3			
Mai	8889	2771	Harnic	7	7.2			
Foarte	1196	268	Nobil	14	7.2			
destul_de	295	66	Cinstit	11	7.1			
la_fel_de	227	61	Generous	6	6.6			
Prea	317	117	Prost	17	6.6			
Tare	35	51	**ADJ_URM_de**	**num**	**sal**			
extrem_de	8	18	Nimic	191	9.1			
			Mâncat	14	8.3			
			Băut	14	8.1			
			Face	142	5.5			

5. CONCLUSION

We have prepared the first large, publicly-available general-purpose corpus of contemporary Romanian, from web sources, using methods pioneered in the web-as-corpus community. We have made the corpus accessible in the Sketch Engine.

The distinctive feature of the Sketch Engine is its word sketches. To set them up for Romanian involved writing a Sketch Grammar to define a set of Romanian grammatical relations. Each grammatical relation is defined using a regular-expression grammar over part-of-speech tags. The paper documents the grammatical relations for Romanian. The word sketch for a word can now be the starting point for a linguist's or lexicographer's analysis of how a word behaves.

The Sketch Engine also prepares a distributional thesaurus and generates sketch differences. These have been introduced.

5.1. Further work

We shall shortly be adding meta-information, in the form of header fields for each document stating which component of the corpus it belongs to, and for the books and newspaper components, details including date, author, and book or newspaper title. Once this is in place, it can be used with the keywords function to find the characteristic vocabulary of each component. Once we have the keywords, further exploration of the nature of the corpus is possible. Lexicographers can compare the different language varieties of the subcorpora.

The corpus is gathered from the web and has many different sources, such as online newspapers, blogs, some literature websites etc. Each of these sites uses a different kind of language, more or less formal. It is difficult to capture the same relation in a sentence taken from a politics article and a sentence coming from a comment a user has made on a blog. As a future improvement, the corpora will be organized in more homogeneous sub-corpora, and more accurate rules will be written based on each of them.

We observe that most errors in the word sketches are caused by annotation errors. As the tagging and lemmatizing tools available for the Romanian language are in a continuous process of improvement, simply re-processing the corpus in future versions should improve the word sketches.

For four languages (Dutch, English, Japanese, Slovene) we have now conducted a quantitative evaluation of word sketches, identifying that, for these languages, around two thirds of the collocates found by the Sketch Engine would be appropriate for inclusion in a published collocations dictionary for the language (Kilgarriff et al., 2010). This kind of exercise evaluates the whole system: corpus, linguistic software, statistics and sketch grammar. We shall replicate this exercise for Romanian, and expect, thereby, to identify a range of ways to improve the corpus, the linguistic tools and the grammar.

REFERENCES

1. BARONI, M., KILGARRIFF, A., POMIKALEK, J., and RYCHLY, P. WebBootCaT: A Web Tool for Instant Corpora. *Proceedings of EURALEX-2006*, Turin, Italy, 2006.
2. BARONI, M., BERNARDINI, S., FERRARESI, A., and ZANCHETTA, E. The WaCky Wide Web: A collection of very large linguistically processed Web-crawled corpora, *Journal of Language Resources and Evaluation*, vol. 43, 3, pp. 209-226, 2009.
3. BRANTS, T. TnT - A statistical Part-Of-Speech Tagger. In *Proceedings of the 6th Applied NLP Conference ANLP*, pp. 224-231, 2000.
4. CEAUȘU, A. Maximum Entropy Tiered Tagging. In *Proceedings of the 11th ESSLLI Student Session*, pp. 173-179, 2006.

5. CRISTEA, D. and FORĂSCU, C. Linguistics Resources and Technologies for Romanian Language, *Journal of Computer Science of Moldova*, vol. 14, no. 1 (40), pp. 33-73, 2006.
6. ERJAVEC, T. MULTEXT-East and TEI: an Investigation of a Schema for Language Engineering and Corpus Linguistics, *this volume*. 2010
7. GREFENSTETTE G. and NIOCHE, J. Estimation of English and non-English Language Use on the WWW. In *Proceedings RIAO,* pp. 237-246, 2000.
8. JONES R. and GHANI R. Automatically Building a Corpus for a Minority Language from the Web. In *Proceedings ACL-2000 Student Workshop*, pp. 29-36, 2000.
9. JUSTESON, J. and KATZ, S. Co-occurrence of the antonymous adjectives and their contexts. In *Computational Linguistics*, vol. 17, pp. 1-19, 1991.
10. KELLER, F., LAPATA, M. Using the web to obtain frequencies for unseen bigrams. In *Computational Linguistics,* vol. 29, no. 3, pp. 459-484, 2003.
11. KILGARRIFF A. and RUNDELL M. Lexical Profiling Software and its Lexicographic Applications: a Case, Mining the Web for Bilingual Text. In *Proceedings of ACL-1999*, pp. 527-534, 1999.
12. KILGARRIFF, A., RYCHLY, P., SMRŽ P., TUGWELL, D. The Sketch Engine. In *Proceedings of Euralex 2004*, Lorient, France, pp. 105-116, 2004.
13. KILGARRIFF, A. Googleology is Bad Science. In *Computational Linguistics*, 33 (1), 2007.
14. KILGARRIFF, A. Corpora by Web Services. In *Proceedings of the Workshop on Web Services and Processing Pipelines in HLT, Seventh International Conference on Language Resources and Evaluation (LREC 2010),* Malta, 2010.
15. KILGARRIFF, A., REDDY, S., POMIKÁLEK, J., and AVINESH, P. A Corpus Factory for Many Languages. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC 2010),* Malta, 2010.
16. KILGARRIFF, A., KOVAR, V., KREK, S., SRDANOVIC, I., and TIBERIUS, C. A Quantitative Evaluation of Word Sketches. In *Proceedings of EURALEX 2010,* Leeuwarden, The Netherlands, 2010
17. RESNIK, P. Mining the Web for Bilingual Text. In *Proceedings of ACL 1999*, pp. 527-534, 1999.
18. SCANNELL, K. The Crubadan Project: Corpus building for under-resourced languages. In *Proceedings of WAC-3: Building and Exploring Web Corpora*, Louvain-la-Neuve, Belgium, 2007.
19. SHAROFF, S. Creating General-Purpose Corpora Using Automated Search Engine Queries. In *WaCky! Working papers on the Web as Corpus*, Gedit, 2006.
20. TUFIȘ, D., ION, R., CEAUȘU, A., and ȘTEFĂNESCU, D. RACAI'S Linguistic Web Services. In *Proceedings of The 6th Language Resources and Evaluation Conference (LREC 2008)*, Marrakech, pp. 28-30, 2008.
21. TUFIȘ, D., ION, R., CEAUȘU, A., and ȘTEFĂNESCU, D. Reifying the alignments, *this volume*. 2010.