The Sketch Engine for Dutch with the ANW corpus

Carole Tiberius (Instituut voor Nederlandse Lexicologie)
Adam Kilgarriff (Lexical Computing Ltd)

1. Introduction

Dictionary making involves finding the distinctive patterns of usage of words in texts. State-of-the-art corpus query systems help the lexicographer with this task. They support searching for phrases, collocates and grammatical patterns; sorting concordances to a wide range of criteria and constraining searches to texts of a particular genre or type. The Sketch Engine (Kilgarriff et al. 2004)[1] is such a corpus query system.

In May 2007, the *Algemeen Nederlands Woordenboek* (ANW) project started working with the Sketch Engine. The ANW corpus was loaded and the system was tuned towards the specific characteristics of the language, corpus and project. In this paper we describe the process. The distinctive feature of the Sketch Engine is 'word sketches': one-page summaries of a word's grammatical and collocational behaviour. We detail the 'Sketch Grammar' for Dutch which is required for word sketches. We also evaluate the word sketches, and find that two thirds of the collocates identified in the word sketches are good.

2. The *Algemeen Nederlands Woordenboek* and the Sketch Engine

2.1 The ANW dictionary

The ANW is a comprehensive online scholarly dictionary of contemporary standard Dutch in the Netherlands and in Flanders (the Dutch-speaking part of Belgium). The project runs from 2001 till 2019 and the first results will be published on the web in 2009. Ultimately, the dictionary will contain 80,000 headwords with a complete description and about 250,000 smaller entries. There will not be a printed version of the dictionary.

2.2 The ANW Corpus

The ANW is a corpus-based dictionary. It is based on the ANW corpus, a balanced corpus of just over 100 million words compiled at the Institute for Dutch Lexicology (INL) and completed in 2004.[2] It comprises: present-day literary texts (20%), texts containing neologisms (5%), texts of various domains in the Netherlands and Flanders (32%) and newspaper texts (40%). The remainder is the 'Pluscorpus' which consists of texts, downloaded from the internet, with words that were present in an INL word list but absent in a first version of the corpus.

To support searches by lemma and part of speech, the corpus has been annotated with lemmas and POS-tags using the technology which was originally developed for the Dutch PAROLE corpus (Does, Van der Voort van der Kleij 2002): a combination of statistical taggers including TnT[3] and three taggers developed at the INL.[4] Lemmatisation was a deterministic procedure, based on an extensive lexicon developed within INL.

2.3 The Sketch Engine

The Sketch Engine is a sophisticated corpus query system. It has standard corpus query functions such as concordancing, sorting and filtering. It also provides word sketches, a distributional *thesaurus* for the language, in which words occurring in similar settings, sharing

[1] <www.sketchengine.co.uk>
[2] For neologisms new corpus material continues to be gathered until the end of the project.
[3] <www.coli.uni-saarland.de/~thorsten/tnt/>
[4] The technology used for POS-tagging is now several years old and we believe there are better tools now available. We are currently investigating the possibility of using Tadpole (Van den Bosch et al. 2007). A dependency parser would also help in determining subject and object relations in Dutch.

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.

2.4 Preparing the corpus

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 (e.g., word or punctuation mark) is on a separate line and where there are associated fields of information, typically the lemma and a 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.

For the ANW, the original tagged and lemmatised corpus format was converted to word-per-line. Document ID numbers were added and information about language variety, which was deduced from the path information of the source text (it being in a Belgian or Dutch folder), was properly encoded in a feature-value pair. Lemmas had two characters appended: a minus sign and a one-letter abbreviation for a word class, since for many purposes we wish to treat the verb *varen* ('sail') as a different lemma to the noun *varen* ('fern'). A special tag, <g>, was added before punctuation marks: it has the effect of suppressing the space character which is otherwise output between one token and the next. Finally, the original Windows character encoding was converted to Unicode to ensure future compatibility. A sample of the ANW as prepared for loading into the Sketch Engine is presented in Figure 1.

```
<doc subcorpus="Neologismen" id="9493" variant="NN" bronentitel="Spits" datering="17 oktober 2000">
<s>
Drie        M(ca,pl)                    drie-m
jaar        N(comm,n,sg)                jaar-n
geleden     R(general,pos,partpast)     geleden-r
kon         V(aux,ind,impf,3,sg)        kunnen-v
Gianna      N(proper,fm,sg)             Gianna-n
Angelopoulou    N(proper,-,sg)          Angelopoulou-n
niets       P(indf,-,-,-,-)             niets-p
fout        N(comm,fm,sg)               fout-n
doen        V(mai,inf,-,-,-)            doen-v
<g/>
.
</s>
```

Figure 1 ANW corpus format as prepared for the Sketch Engine

2.5 Concordances

Once the corpus was loaded into the Sketch Engine, it could be searched for, for example, a lemma with word class specified. This search is optionally case-sensitive as frequently, lemmas starting with uppercase need to be distinguished from those starting with lower case: the lemma *Schilder* is not the same as the lemma *schilder*. The former is a proper name, whereas the latter is a common noun meaning 'painter'.

Errors in lemmatisation and tagging often lead to unexpected results for the lexicographer. There always is an explanation, but it often requires a closer examination of the tagging and lemmatisation to unearth it. Recurring errors for the ANW corpus include separate lemmatisation of singular and plural forms of the same or a derived lemma. For instance, it turns out that a lot of the plural forms of compounds formed with the lemma

fanaat ('fanatic') are lemmatised incorrectly with the plural form. Thus, the plural word form *filmfanaten* ('film fanatics') is wrongly lemmatised as *filmfanaten* whereas the correct analysis would be the lemma *filmfanaat.* Thus, the corpus contains

WORD	LEMMA	POS-TAG
filmfanaat	filmfanaat	N(comm,fm,sg)
filmfanaten	**filmfanaten**	N(comm,fm,pl)

instead of:

filmfanaat	filmfanaat	N(comm,fm,sg)
filmfanaten	**filmfanaat**	N(comm,fm,pl)

So when the lexicographer wants to search for all instances of the lemma *filmfanaat*, they are at risk of missing the plural instances since they will not be retrieved if the lexicographer only makes the search *filmfanaat.* The lexicographer needs to, first, realise what they are missing, and then make a second search for *filmfanaten*.

A wide range of search options are offered by using the CONTEXT section. Here the lexicographer 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 lexicographer editing the lemma *cartograaf* ('cartographer') may wish to see which verbs can follow this lemma. To this end, *cartograaf* 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 2.

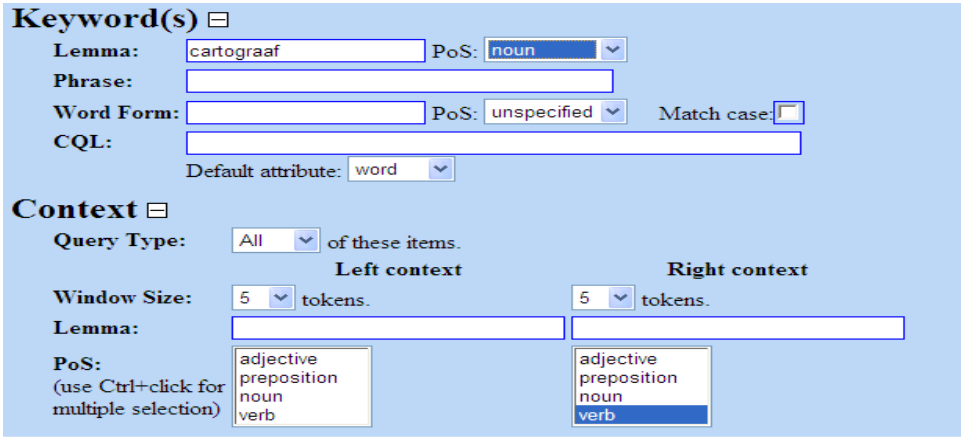

Figure 2 Context-dependent concordance search

On the results page the concordances are shown using KWIC view (one line per instance, keyword centered). With VIEW options we can change the view to whole sentences. We can also view additional attributes such as POS tags or lemma alongside each word which is useful for finding out why an unexpected corpus line has matched a query. By selecting fields in the references column, we can determine what source information should appear in blue at the left-hand end of the concordance line. For the ANW, 'subcorpus' and 'variant' are usually selected so the lexicographer can immediately tell which subcorpus and which language variety (Belgian Dutch or Dutch Dutch) the concordance line is from.

2.6 One-click copying

It is central to the process of corpus lexicography that lexicographers often want to insert example sentences from the corpus into the dictionary. Until recently, the method for this was the standard one offered by the computer's operating system: select the sentence using the

mouse, copy it (e.g. using CTRL-C) and paste it in the correct place in the dictionary text being edited. This was a process the lexicographers repeated many times, and it was cumbersome: to see the whole sentence (which typically is not in the KWIC line) they first had to click on the node word to call up more context; they then had to look to see where the sentence began and ended, and then they needed to manoeuvre the mouse first to the beginning, then to the end of the sentence. To streamline the process ONE-CLICK COPYING was introduced. An icon is provided, which appears at the right-hand end of each concordance line (Figure 3). By clicking this icon, the full sentence is copied directly onto the clipboard. It can then be pasted into the dictionary entry as before. It is also possible to do this for a set of concordance lines.

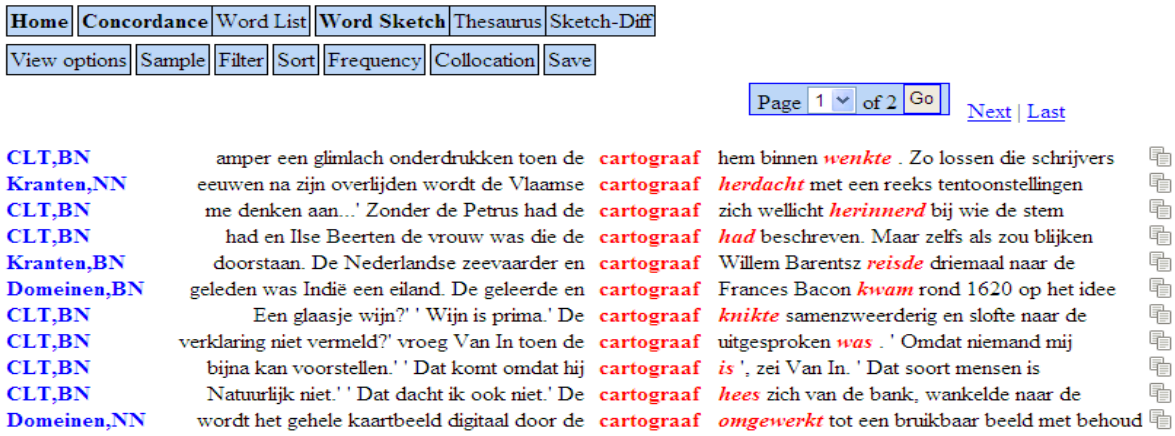

Figure 3 Concordance view with one-click multiple line copying

In the ANW entries, examples are entered into the editor together with their bibliographic reference. To this end, the Sketch Engine team developed an extension to the one-click copying facility in which the bibliographic information was gathered and placed on the clipboard along with the sentence. The ANW editing software has been adapted so that, when the example and source information are pasted into it, both the concordance and the reference go directly into the appropriate fields.

2.7 Good dictionary example finding

Some corpus sentences make good dictionary examples 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 dictionary example is a high-level lexicographic skill. But to rule out lots of bad sentences is less demanding, and the computer can help by doing this groundwork. A new function, GDEX (Good Dictionary Example eXtractor) 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 a dictionary 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 Dutch 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 lexicographer 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.

While the GDEX rules were prepared for English and only minimal customisation has taken place (replacing an English wordlist with a Dutch one), evidence to date is that it works well for Dutch.

2.8 Word Lists

The word list function offers the lexicographer three options, basic, keywords and 'findX'.

2.8.1 Basic word lists

The first option allows the lexicographer to create a word list. Regular expressions can be used in the search box which is useful for detecting compounds in Dutch. Word lists can be for word forms or lemmas, and for the whole corpus or a particular subcorpus.

2.8.2 Keywords

KEYWORDS allows the lexicographer to find words that are characteristic for a particular language variety or subcorpus. As the ANW covers material from Flanders (47m tokens) and the Netherlands (68m), it was possible to generate keywords for Belgian Dutch and Dutch Dutch. In the top 50 words of the Belgian list, we find *frank* ('franc'), *gewestplan* ('regional plan'), *zoekertjes* ('ads'), *omzendbrief* ('circular') and words which are spelt differently in Flanders such as *tornooi* ('tournament'), *fiskaal* ('fiscal'), *organizatie* ('organisation'). Typical Dutch Dutch words are *peuterspeelzaal* ('playgroup'), *wethouder* ('councilor'), *woningbouwcorporatie* ('housing corporation'), *strippenkaart* ('bus and tram card') and *tientje* ('tenner').

2.8.3 FindX

The option FINDX allows us to find 'the words that are most X', where 'X' may be replaced by a wide range of characteristics. Thus, a lexicographer 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.

For lexicographers this is useful information as they often want to know whether a noun needs to be marked as usually plural, or a verb as having a particular complementation pattern. Lexicographers are rarely in a position to check. 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 accessible to the lexicographers. The Sketch Engine function does all these tasks.

A list of the Dutch nouns which have the strongest tendency to be used in the plural (excluding the 'always plural' nouns, whose behaviour is already well-known) is shown in Table 1.

Highly plural Dutch nouns	
gepensioneerde (*pensioner*)	zoekopdracht (*query*)
militant (*militant*)	leveringsvoorwaarde (*term of delivery*)
arbeidsongeschikte (*s.o. unable to work*)	behoeftige (*needy*)
uitkeringsgerechtigde (*s.o. on welfare*)	kiesgerechtigde (*voter*)
thuisloze (*homeless*)	zoekterm (*search term*)
gereformeerde (*member of the Dutch Reformed Church*)	progressief (*liberal*)
geslaagde (*successful candidate*)	milieuoverweging (*environmental consideration*)
internetprovider (*internet provider*)	bloedvat (*blood vessel*)
lager (*bearing*)	meerderjarige (*adult*)
handelspraktijk (*commercial practice*)	voorlichtingsactiviteit (*information activity*)
vredesonderhandeling (*peace talk*)	opleidingseis (*education requirement*)
milieukwaliteitsnorm (*environmental quality norm*)	rechtsbijstandverlener (*provider of legal expenses insurance*)
katholiek (*catholic*)	zeevarende (*seaman*)
zorgverstrekker (*health care provider*)	onderwijsresultaat (*education result*)
edele (*noble*)	dementerende (*s.o. growing demented*)
succesfactor (*success factor*)	

Table 1 List of nouns which are most 'plural'

We note that almost half are nouns denoting people: *gepensioneerde* ('pensioner'), *militant* ('militant'), *katholiek* ('catholic'), *edele* ('noble'). Most often we talk about the group. The list also contains a number of IT related words such as *internetprovider*, *zoekterm* ('search term'), *zoekopdracht* ('query').

2.9 Word Sketches

Word sketches are the distinctive feature of the Sketch Engine. They are one-page, automatic, corpus-based summaries of a word's grammatical and collocational behaviour. Word sketches improve on standard collocation lists by finding 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 latter 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[5]
2:"A.*" 1:"N.*"
```

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).

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

[5] For Dutch we adopted the following naming convention in the grammatical relations. The syntactic category of the node word is written in upper case, whereas the syntactic category of the collocate is written in lower case.

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 which is central to the preparation of word sketches.

2.9.1 Dutch Sketch Grammar

The Dutch sketch grammar is geared to the ANW. When compiling an entry, lexicographers are asked to provide collocations, in specified grammatical relations. The list for nouns is given in Table 2. For all word classes the number of grammatical relations amounts to around 70. These relations formed the basis for the sketch grammar for Dutch, which defines 55 relations.

als object bij een werkwoord *object-of*	met dat-zin *with 'dat'-compl*
als subject bij een werkwoord *subject-of*	met vraagzin *with wh-compl*
met koppelwerkwoord *with auxiliary*	met of-zin *with 'of'-compl*
met adjectief ervoor *premodifying adjective*	met alsof-zin *with 'alsof'-compl*
met adjectivisch tegenwoordig deelwoord *premodifying present participle*	met aanwijzend voornaamwoord *with demonstrative pronoun*
met adjectivisch voltooid deelwoord *premodifying past participle*	met bezittelijk voornaamwoord *with possessive pronoun*
met voorzetselgroep *with PP*	met onbepaald voornaamwoord *with indefinite pronoun*
in voorzetselgroep *in PP*	met persoonlijk voornaamwoord *with personal pronoun*
met substantief ervoor *premodifying noun*	voorafgegaand door naamvalsgenitief *premodifying genitive*
met substantief erachter *postmodifying noun*	gevolgd door naamvalsgenitief *postmodifying genitive*
met telwoord ervoor *premodifying numeral*	met eigennaam *with proper noun*
met telwoord erachter *postmodifying numeral*	met lidwoord *with article*
met adjectief erachter *postmodifying adjective*	met ander, nevengeschikt substantief *with coordinated noun*
met infinitief met te *with infinitive plus 'te'*	overig *other*
met infinitief met om te *with infinitive plus 'om te'*	

Table 2 List of grammatical relations for nouns in the ANW

Not all patterns specified in the ANW can be automatically deduced from the corpus as in some cases the markup is not detailed enough. For instance, genitive forms are not marked in the corpus so a pattern like 'premodifying genitive' cannot be found automatically.

The grammatical relations in the Dutch sketch grammar fall into four classes, i.e. symmetric, dual, trinary and unary relations. They are presented in tables 3 to 6 below. The third column in each table gives the information as a triple. When the corpus is parsed with the grammar, the output is a set of tuples, one for each case where each pattern matched. The tuple comprises (for the two-argument case), the grammatical relation, the headword and the collocate. This work is all done on lemmas.

Symmetric relations

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

Relation	Example	Triple
AND-OR en/of	koek en ei *biscuit and egg*	<en/of, koek, ei>
COMMA komma	lopen, fietsen *walk, cycle*	<komma, lopen, fietsen>

Table 3 Symmetric Relations Dutch sketch grammar

The same instance may have more than one relation of the same kind, as in *boter*, *kaas en eieren* ('butter, cheese and eggs') where *kaas* has two AND-OR relations[6], one with *boter* and one with *eieren*.

Dual relations

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. A typical dual which can be inversed, is the pair 'verb and its object' and 'noun and the verb it is object of'. In the table below, inverse relations are separated from their counterpart by a forward slash (/), for instance 'object/object_bij'. This is also the character used for separating inverse relations in the sketch grammar. The vertical bar | in the table below, indicates alternative syntactic categories that can occur in a particular grammatical relation. We use it here to group similar relations. For instance, SUBST|ADJ| WW+te_inf under INF-COMPLEMENT represents three grammatical relations in the sketch grammar, i.e. SUBST+inf, ADJ+inf and WW+inf.

[6] The AND-OR relation includes coordinate structures with a comma for nouns and adjectives. For verbs a separate COMMA relation was defined for this particular pattern.

Relation	Example	Triple
OBJECT object/object_bij	een huis kopen *buy a house*	<object_bij, huis, kopen>
SUBJECT subject/subject_bij	het paard hinnikt *the horse whinnies*	<subject_bij, paard, hinniken>
MODIFIER adj+SUBST/ADJ+subst adj_deelw_tt|adj_deelw_vt| adj_te_inf+SUBST adj|subst|ww+ADJ/ADJ|SUBST|WW+adj	jonge kinderen *young children*	<adj+SUBST, kind, jong>
NUMERICAL-MODIFIER telw+SUBST/SUBST+telw	hoofdstuk 3 *chapter 3*	<telw+SUBST, hoofdstuk, 3>
NOUN-MODIFIER subst+SUBST/SUBST+subst	een mand appels *a basket of apples*	<subst+SUBST, appel, mand>
GENITIVE-NOUN-POSTMODIFIER SUBST+des_der_dezer+subst	de plek des onheils *the scene of the disaster*	<SUBST+des_der_dezer+subst, plek, onheil>
PROPER-NOUN-MODIFIER SUBST+eigennaam/ subst+SUBST-eigennaam	de stad Antwerpen *the city of Antwerp*	<SUBST+eigennaam, stad, Antwerpen>
POSSESSED bezit_vnw+SUBST	mijn broek *my trousers*	<bezit_vnw+SUBST, broek, mijn>
DETERMINER aanw_vnw|onbep_vnw+SUBST	deze bloem *this flower*	<aanw_vnw+SUBST, bloem, deze>
ADV-COMPLEMENT bijw+WW (verb particle)/WW+bijw bijw+ADJ	hij blijft thuis *he stays at home*	<WW+bijw, blijven, thuis>
INF-COMPLEMENT SUBST|ADJ|WW+te_inf SUBST|ADJ| WW+om_te_inf WW_inf	de verwachting te slagen *the expectation to succeed* een kind om te zoenen *lit.: a child to be kissed*	<SUBST+te_inf, verwachting, slagen>
WH-COMPLEMENT SUBST|ADJ|WW+vraagzin	weten waarom hij kwam *know why he came*	<WW+vraagzin, weten, waarom>
EXPERIENCING OBJECT WW+ondervindend_vw	het verbaast me *it surprises me*	<WW+ondervindend_vw, verbazen, mij>
INDIRECT OBJECT WITH PP 'AAN' WW+io_met_aan	ik gehoorzaam aan hem *I obey him*	<WW+io_met_aan, gehoorzamen, hem>
LIKE-COMPLEMENT WW+als_bepaling	lopen als een kievit *run like the wind*	<WW+als_bepaling, lopen, kievit>
PP-COMP SUBST|ADJ|WW+vzg	vers uit zee *fresh from the sea*	<ADJ+vzg, vers, uit>
IN_PP SUBST_in_vzg	op zijn hoede *be on one's guard*	<SUBST_in_vzg, hoede, op>

Table 4 Dual relations

Trinary relations

Trinary relations describe relations between three dependent items. They are generally used for extracting prepositional complements. The system interprets a trinary relation as a set of binary ones, one for each preposition. For instance, a trinary relation for noun-PP-noun constructions results in a separate relation for noun-of-noun (*bank of the river*), noun-over-noun (*bridge over the river*), noun-in-noun (*house in town*), etc. We experimented with this for Dutch PP complements, but found that this resulted in too many relations crowding the display of the word sketches. For now PP complements are included as dual relations and the task of separating them out is left to the lexicographer. Only one trinary relation is currently used, namely where the lexicographer would like to know the possible combinations of a verb with an object and a complement. This pattern is particularly characteristic for verbs of change of state and judgement as in the sentence *ik verklaar hem schuldig* ('I declare him

guilty'). For this kind of grammatical relation a separate set of results will be generated for each typical object.

Relation	Example	Triple
VERB-OBJECT-ADJ WW+%s_object_adj	ik pleit hem vrij *I plead for him to be freed*	<WW+hem_object_adj, pleiten, vrij>

Table 5 Trinary relation

Unary relations

Finally, unary relations can be defined. They are used to extract complementation patterns. For instance, a lexicographer would like to know that a verb is frequently followed by a relative clause starting with *dat* ('that') or that a noun is preceded by an article. Here there is just a headword showing a particular pattern or construction, but there are no separate collocates. In the Dutch sketch grammar six unary relations have been defined.

Relation	Example	Triple
ARTICLE lidw+SUBST	de stad *the city*	<lidw+SUBST, stad>
COMP dat-zin of-zin alsof-zin	de verwachting dat het goed kwam *the expectation that it would be all right* de vraag of de trein vertraging had *the question whether the train was delayed* het gevoel alsof ze niet gegeten had *the feeling as if she had not eaten*	<dat_zin, verwachting>
PP vzg_aan_het+WW vzg_uit+WW	aan het schilderen *painting* uit vissen *gone fishing*	<vzg_uit+WW, vissen>

Table 6 Unary relations

2.9.2 Word sketch display

Table 7 shows a word sketch for the noun *water* ('water'). Under the column adj+SUBST we find typical qualifying adjectives, denoting kinds of water distinguished by their properties or origin, e.g. *warm water* ('warm water'), *koud water* ('cold water') but also fixed idioms, such as *zout water* ('salt water') and *zoet water* ('river water') are revealed. The adjective *zuiver* ('clear') has an idiomatic use in the combination *van het zuiverste water* ('of the best kind').

One key role of word sketches is to help lexicographers not to miss senses, phrases and idioms for the word. In the word sketch for *water* we note that there are collocates relating to different uses of this noun. The adjective *drinkbaar* ('drinkable') relates to *water* as 'a drink, satisfying thirst'; *gas* ('gas') and *elektriciteit* ('electricity') relate to the sense 'supplied for domestic needs'; *opkomend* ('tide coming in') refers to the liquid of which seas, lakes, and rivers are composed. The combination *territoriaal water* ('territorial water') is a specific concept in maritime law.

Not all verbs which typically have water as an object are found. This is due to the restricted nature of the grammatical patterns for object in the sketch grammar for Dutch, as discussed further below.

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 Rychlý 2008 for the formula used to compute salience). The user can set a frequency threshold so low-frequency collocations are not shown, or click a button for 'more data' or 'less data'. They can go to the related concordance by clicking on the hit-count for a collocation.

adj+SUBST	num	sal
warm	556	57.63
koud	402	55.28
lauw	132	53.58
zout	159	51.57
territoriaal	109	48.03
ijskoud	88	47.92
ondiep	83	47.22
zoet	140	46.28
drinkbaar	41	45.7
heet	152	45.66
overtollig	87	44.99
zuiver	129	44.14
troebel	50	40.5
schoon	129	37.35
brak	30	34.56
snelstromend	19	34.47
handwarm	14	33.05
helder	82	32.74
zuurstofrijk	17	31.44
kristalhelder	16	30.63

adjdeelwtt+SUBST	num	sal
stromend	294	80.88
kokend	203	76.55
stilstaand	88	58.27
wassend	22	42.13
opspattend	12	34.55
klotsend	12	34.28
kolkend	14	31.54
rimpelend	8	29.44
voldoende	49	28.61
schuimend	11	28.14
levend	22	23.55
stijgend	15	23.12
glinsterend	9	22.56
opkomend	10	21.59
stinkend	8	20.08
dampend	6	18.51
staand	11	18.11
bruisend	6	18.05
golvend	6	17.67
lopend	9	15.69

adjdeelwvtt+SUBST	num	sal
gedistilleerd	24	49.74
vervuild	32	45.2
gezuiverd	21	44.27
gezouten	20	43.39
gekookt	21	37.5
bevroren	19	34.28
verontreinigd	11	29.88
besmet	17	29.73
gefilterd	7	26.82
gewijd	6	21.14
gekleurd	7	18.05

subst+SUBST	num	sal
liter	254	61.83
emmer	140	58.83
druppel	169	57.83
glas	302	57.83
hoeveelheid	182	45.15
slok	68	43.94
plas	46	38.12
fles	70	35.35
laagje	30	32.96
straal	27	32.32
sanitair	19	31.25
meter	72	31.17
teil	13	30.95
deciliter	12	28.69
vuil	23	28.27
ketel	15	27.79
kuub	9	27.46
oppervlakte	20	26.68
beetje	55	26.18
afdeling	47	25.32
bak	20	25.07

en/of	num	sal
elektriciteit	83	49.2
bodem	77	42.75
zeep	45	41.93
lucht	94	41.82
vuur	58	39.72
gas	36	32.64
brood	37	32.33
voedsel	35	31.88
melk	25	27.68
aarde	25	26.84
wind	25	26.24
natuur	30	26.23
modder	12	25.72
energie	30	25.58
alcohol	18	24.88
azijn	9	24.46
zand	16	23.03
licht	28	22.41
stoom	8	22.03
bloed	18	21.83

SUBST_in_vzg	num	sal
boven	891	56.93
onder	1192	42.88
met	1713	25.91
over	235	13.07
zonder	85	12.92
in	1215	12.8
te	16	12.8
aan	226	7.73
uit	132	7.22
door	153	6.83
op	279	4.38

object_bij	num	sal
drinken	14	27.53
horen	15	22.35
vragen	6	13.74
krijgen	7	11.64
zien	6	9.72

SUBST+te_inf	num	sal
drinken	37	33.97
halen	42	29.82
zuiveren	11	27.48
zetten	30	24.7
lozen	7	23.95
filteren	6	22.93
vissen	7	21.64
geven	37	21.43
koken	8	20.81

Table 7 Word Sketch for the noun *water* ('water'), ANW frequency=32858

2.10 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.

We can ask 'which words share most tuples', in the sense that, if the database includes both <object, drinken, bier> and <object, drinken, wijn>, then we can say that *bier* ('beer') and *wijn* ('wine') share a triple, namely there are both the object of the verb *drinken* ('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 8 presents an extract of the thesaurus entry for the adjective *duidelijk* ('clear'). The resulting list contains adjectives such as *helder* ('clear'), *bekend* ('known'), *zichtbaar* ('visible'). There are also two antonyms, i.e. *slecht* ('bad') and *moeilijk* ('difficult'): in this approach, antonymy is a special form of similarity.

Lemma	**Score**	**Freq**
helder *(clear)*	0.283	5464
concreet *(concrete)*	0.276	7808
belangrijk *(important)*	0.272	64877
sterk *(strong)*	0.272	22284
mogelijk *(possible)*	0.268	40491
positief *(positive)*	0.268	13010
bekend *(known)*	0.249	33315
zichtbaar *(visible)*	0.24	7282
moeilijk *(difficult)*	0.238	18570
interessant *(interesting)*	0.228	8521
slecht *(bad)*	0.224	19525
goed *(good)*	0.223	104850
mooi *(nice)*	0.222	28651
specifiek *(specific)*	0.214	11507

Table 8 Thesaurus output for *duidelijk* (adj), ANW frequency=33722

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 are distinctive of each but do not occur with the other. For example, the adjectives *zwart* ('black') and *white* ('wit'). Both are often found conjoined with other colour terms, and premodify nouns such as *jurk* ('dress') and *haar* ('hair'). The contrast becomes apparent from the *zwart*-only and *wit*-only patterns. *Zwart* occurs with premodifying adjectives such as *stinkend* ('smelly'), *versleten* ('used') and *vuil* ('dirty'), whereas white is used with adjectives such as *stralend* ('dazzling') and *smetteloos* ('immaculate').

Sketch Difference of zwart ('black) and wit ('white')					
shared patterns	***zwart*-only patterns**			***wit*-only patterns**	
en/of num num	**adj+ADJ num sal**		**ADJ+adj num sal**	**adj+ADJ**	**ADJ+adj num sal**
wit 123 14	sluik 7 23.2		glimmend 18 31.1	smetteloos 33 42.1	katoenen 21 36.8
zwart 5 114	nauwsluitend 6 22.0		leren 14 29.2	stralend 23 29.7	gepleisterd 10 33.5
rood 20 71	versgemalen 5 18.2		fluwelen 13 28.6	verblindend 15 29.2	schuimend 9 28.0
bruin 29 37	versleten 6 14.7		lederen 11 27.2	helder 11 14.7	linnen 10 27.3
geel 16 35	immens 6 14.5		rubberen 9 24.5	fris 7 13.9	gekalkt 6 26.9
blauw 8 22	stinkend 5 14.1		Amerikaans 31 22.3	opvallend 9 13.6	geschilderd 8 23.3
grijs 16 14	ouderwets 6 13.0		gemarmerd 5 22.0		betegelde 5 20.5
paars 9 11	vuil 6 12.9		gekleed 9 22.0		gestreept 6 20.2
groen 12 16	berucht 5 12.8		gelakt 6 21.1		porseleinen 5 18.9
gekleurd 6 7			behaard 6 20.8		
			krullend 5 19.8		
ADJ+subst	**ADJ+subst**			**ADJ+subst**	
haar 210 70	gat 386 56.4			wijn 483 53.8	
overhemd 9 97	woud 52 35.2			bloedcel 124 50.9	
laken 10 78	magie 42 35.2			schort 56 40.0	
jurk 84 92	weduwe 61 35.0			poeder 62 39.3	
	Piet 112 34.3			bloedlichaampje 35 37.6	
	markt 199 32.9			kerst 39 35.6	
	kas 39 31.0			blouse 40 34.1	
	komedie 34 30.6			roos 57 33.0	

Table 9 Extract of Sketch Difference of adjective *zwart* (ANW-freq=15534) and adjective *wit* (ANW-freq=16538)

3. Evaluating the Dutch Sketch Grammar

In 2008, the Sketch Engine team set up an international experiment to evaluate word sketches. Although word sketches had been in use since 1999, and had received many favourable reviews from linguists and lexicographers, these had all been informal and it was now time to undertake a more formal and quantitative evaluation.

There are many ways in which evaluation could be approached. We decided to focus on the user's perspective, and specifically, to look at the case where the user is a lexicographer. The word sketch aims to provide the lexicographer with the salient collocates for a word, so we decided that a good test for the word sketches is: 'how many of the collocates provided in the word sketches, are suitable for including in a collocations dictionary (and which collocates which should be there, are missing)'. We hoped the question was one that professional lexicographers could answer systematically and consistently. We took the Oxford Collocations Dictionary (2002) as a model for the kind of dictionary we had in mind.

Teams developing and using the Sketch Engine for Dutch, English, Japanese and Slovene took part. In each case more than one person passed judgement on each collocate, so it was possible to work out how consistent and objective the judgements were.

One critical factor was to choose the sample of headwords with care. We wanted to assess the word sketches across the vocabulary. We sampled nouns, verbs and adjectives (which, between them, make up over 99% of the headwords of most standard dictionaries), which were high, medium and low frequency words. Within this sampling frame, we took a random sample of 40 Dutch words.

The system selected the collocates for each of those words, and then, for each collocate, the lexicographer was asked to judge whether the collocate was

- Good
- Good but wrong grammatical relation
- Maybe (not striking collocate)
- Maybe (specialised vocab), or
- Bad.

The evaluation for Dutch was carried out by Fons Moerdijk and by the first author of this paper using a customised version of the Sketch Engine[7] in which word sketches contained only the twenty highest-scoring collocates for each word, and in which each collocate was associated with a menu of the five possible responses as shown in Figure 4.

strand **ANW-INL freq = 3601**

Rubric: **G** = Good **Gb** = Good but wrong grammatical relation **M** = Maybe (not striking collocate)
Ms = Maybe (specialized vocab) **B** = Bad

Gramrel	Collocation	Rating					Freq
		G	Gb	M	Ms	B	
adj+SUBST	scheveningse	○	○	○	○	○	12
adj+SUBST	parelwit	○	○	○	○	○	8
adj+SUBST	zonovergoten	○	○	○	○	○	10
adj+SUBST	Normandisch	○	○	○	○	○	6
adj+SUBST	zonnig	○	○	○	○	○	11
adj+SUBST	overvol	○	○	○	○	○	9
adj+SUBST	stuk	○	○	○	○	○	12
adj+SUBST	breed	○	○	○	○	○	19
adj+SUBST	tropisch	○	○	○	○	○	9
adj+SUBST	wit	○	○	○	○	○	20
adj_deelw_vt+SUBST	verlaten	○	○	○	○	○	15
en/of	duin	○	○	○	○	○	9
en/of	zee	○	○	○	○	○	6

Figure 4 Word sketch for evaluation for Dutch noun *strand* ('beach')

In total, 782[8] collocations were judged and the evaluators agreed on 501 instances. Of those, 332 were evaluated as 'good' by both evaluators, 46 as 'maybe' and 123 as 'bad'. Thus, looking only at the collocates where there was agreement, the system was 66% (two thirds) correct in identifying good collocates.[9]

The table below gives an overview of the number of different evaluations per word. The number before the / indicates the number of different evaluations; the number after the / the total number judged. Thus, the evaluators scored all collocations generated by the sketch engine for *strand* the same except for one.

[7] A slightly reduced set of rules was used for this experiment. Rules describing rare constructions were excluded, i.e.: =komma, =SUBST+telw, SUBST+vraagzin, WW+vraagzin, aanw_vnw+SUBST, bezit_vnw+SUBST, onbep_vnw+SUBST, =WW+%s_object_adj, =WW+io_met_aan, WW+ondervindend_vw, lidw+SUBST, =dat_zin, =of_zin, =alsof_zin.

[8] For the noun *behulp* ('help') only 2 collocates were generated by the system.

[9] There was never agreement for evaluations in the categories 'good but wrong grammatical relation' and 'maybe (specialised vocabulary)'. There seemed to be uncertainty on how to score collocations which were clearly good collocations but in specialised vocabulary. They were often judged as 'good' by one of the evaluators and as 'maybe (not striking collocate)' or 'maybe (specialised vocabulary)' by the other.

NOUN	eval diff	ADJECTIVE	eval diff	VERB	eval diff
Common					
adres (*address*)	3/20	aardig (*nice*)	0/20	aanwijzen (*indicate*)	9/20
argument (*argument*)	5/20	emotioneel (*emotional*)	3/20	bewaren (*keep*)	8/20
behulp (*help*)	2/2	sociaal (*social*)	2/20	oproepen (*call up*)	4/20
bezoeker (*visitor*)	4/20	volledig (*complete*)	6/20	veroordelen (*condemn*)	4/20
onderdeel (*part*)	8/20				
strand (*beach*)	1/20				
Mid					
dirigent (*director*)	4/20	grafisch (*graphical*)	5/20	logeren (*stay*)	9/20
notering (*quotation*)	10/20	slim (*smart*)	5/20	nestelen (*nestle*)	9/20
rubriek (*section*)	7/20	wederzijds (*mutual*)	6/20	overtreffen (*exceed*)	4/20
samenspel (*combined play*)	9/20			variëren (*vary*)	4/20
schrijfster (*fem. writer*)	13/20				
straling (*radiation*)	5/20				
Low					
immuunsysteem (*immune system*)	12/20	beweeglijk (*agile*)	11/20	inpalmen (*charm*; *grab*)	16/20
		onverlet (*unobstructed*)	11/20	knabbelen (*nibble*)	7/20
leerprobleem (*learning problem*)	8/20	weerzinwekkend (*disgusting*)	8/20	neuriën (*hum*)	6/20
octaaf (*octave*)	11/20	wollen (*woollen*)	0/20	spuwen (*spit*)	11/20
Schipbreuk (*shipwreck*)	11/20				
scholier (*pupil*)	10/20				

Table 10 Results of evaluation for Dutch

We see that there is most agreement between the evaluators for common words and least for the low frequency words.[10]

We also observe that there is less agreement for verb collocates and that for subject and object relations striking collocations are missed. For instance, with the lemma *bewaren* ('keep'), we get the collocate *koelbloedigheid* ('level-headedness'), whereas the collocate *kalmte* ('calm') which is more frequent is absent. This is due to the restrictive nature of the subject and object patterns that are currently included in the sketch grammar for Dutch. Verb-object and verb-subject, while frequently the most significant grammatical relations for describing the behaviour of nouns and verbs, are also relatively complex to identify. Dutch allows word order variation and both subject and object can occur in the same position before the verb. Subject occurs by default in this preverbal position, but corpus evidence shows this is actually just so 70% of the time (Bouma 2008). So word order is not a reliable source of information for assigning grammatical functions and subject/object rules easily generate a lot of noise. We hope parsing the corpus with a state-of-the-art parser will improve results in the future.

4. Conclusion

We have loaded the ANW corpus into the Sketch Engine. The process was designed to support ANW lexicography, which it now does well. The ANW in the Sketch Engine has been in extensive daily use by a team of 15 lexicographers since May 2007.

The distinctive feature of the Sketch Engine is word sketches. To prepare them for Dutch involved writing a Sketch Grammar to define the set of Dutch grammatical relations, as detailed in the ANW dictionary. Each grammatical relation is defined using a regular

[10] Common words are words with a frequency between 2781 and 91523; mid frequency words have a frequency between 574 and 2779 whereas low frequency words have a frequency between 126 and 574.

expression over part-of-speech tags. The paper documents the grammatical relations for Dutch. The word sketch for a word is now the starting point for an ANW lexicographer's analysis of how a word behaves. The use of the word sketches makes the ANW lexicography more complete and more consistent – and faster.

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

We have evaluated the Dutch word sketches, in an exercise carried out in parallel for four languages. We found that two thirds of the collocates were 'good', and would not be out of place in a Dutch collocations dictionary.

Most errors in the word sketches result from errors in lemmatisation and POS-tagging. We are currently exploring alternative tools for Dutch linguistic processing, including the Tadpole software from Tilburg, which would also provide dependency parsing, which will, we hope, further improve the accuracy of the word sketches.

We see an extensive further role for the evaluation framework. If we know, for a fair-sized sample of words, what collocates the word sketches ***should*** contain, we can use that information to evaluate different versions of the corpus, or different POS-taggers, or different sketch grammars. The exercise that has already been completed gives us some such data. We are considering how it can best be extended so we can use it to evaluate further developments.

Bibliography

Bosch, A. van den, G.J. Busser, W. Daelemans and S. Canisius (2007), 'An efficient memory-based morphosyntactic tagger and parser for Dutch', in: F. van Eynde, P. Dirix, I. Schuurman and V. Vandeghinste (eds), *Selected Papers of the 17th Computational Linguistics in the Netherlands Meeting*, Leuven, Belgium, 99-114.

Bouma, G. (2008), *Gathering Corpus Evidence of Word Order Freezing in Dutch*. Poster at the *International Conference on Linguistic Evidence 2008*, Tübingen, 31 January - 3 February 2008.

Daelemans, W., J. Zavrel, K. van der Sloot and A. van den Bosch (2001), TiMBL: Tilburg Memory Based Learner, version 4.0, Reference Guide. ILK Technical Report 01-04, *Technical report,* ILK.

Does, J. de, J. van der Voort van der Kleij (2002), 'Tagging the Dutch PAROLE Corpus', in: M. Theune et al. (eds), *Computational Linguistics in the Netherlands 2001; Selected Papers from the Twelfth CLIN Meeting*. Amsterdam - New York, Rodopi, 62-76.

Grefenstette, G. (1994), *Explorations in Automatic Thesaurus Discovery*. Kluwer.

Kilgarriff, A., P. Rychlý, P. Smrž and D. Tugwell (2004), 'The Sketch Engine', in: *Proceedings Euralex*. Lorient, France, July: 105-116. Reprinted in *Lexicology: Critical concepts in Linguistics* Hanks, editor. Routledge, 2007.

Kilgarriff, A., M. Husák, K. McAdam, M. Rundell, P. Rychlý (2008), 'GDEX: Automatically finding good dictionary examples in a corpus', in: *Proceedings EURALEX*, Barcelona, Spain.

Lin, Dekang (1998), 'Automatic retrieval and clustering of similar words', in: *Proc. COLING-ACL*, Montreal 1998, 768-774.

Oxford Collocations Dictionary for Students of English (2002). Oxford University Press.

Rychlý P. (2008), 'Statistics used in the Sketch Engine', in: *Proceedings RASLAN workshop*, Masaryk University, Brno. Also available from Sketch Engine website.