Renovating a wordclass tagset: from WOTAN to WOTAN-2

Hans van Halteren
Department of Language and Speech
University of Nijmegen

Introduction

WOTAN

In 1994, the WOTAN wordclass tagset for Dutch was created as part of a Master's thesis project (Berghmans, 1994). The starting point was the classification used in the most popular descriptive grammar of Dutch, the *Algemene Nederlandse Spraakkunst* (ANS; Geerts *et al.*, 1984). The actual distinctions encoded in the tagset were to be selected on the basis of their importance to the potential users, as estimated from a number of in-depth interviews with interested parties in the Netherlands. However, the project also included the upgrade of more than a million words of corpus material tagged in an earlier project. Since there was only a modest amount of time available for manual adjustments, this upgrade had to be feasible with mostly automatic means and we were forced to abandon some of the interesting but labour-intensive distinctions. Even so, the resulting tagset was judged to be a very useful one and has since been used in several tagging projects and experiments in the Netherlands and Belgium, its popularity probably being based on both its detailedness (around 250 tags) and the availability of a large training corpus (some 1.4Mw).

WOTAN-2

In 1998, however, we decided that it was time to start work on a successor, WOTAN-2. This would not only bring the tagset in line with recent developments such as the new, revised version of the ANS (Haeseryn *et al.*, 1997) and the EAGLES guidelines, but would also allow us to add a number of important distinctions that were left out earlier. Furthermore, the upgraded tagset would be designed so as to provide a better compatibility with other major Dutch NLP resources, viz. the CELEX database and the AMAZON syntactic parser.

Now, June 1999, the initial design of WOTAN-2 has been finalized. An upgrade of the written part of the Eindhoven corpus (750Kw; uit den Boogaart, 1975) from WOTAN-1 to WOTAN-2 (version 1) is well underway. The newspaper section (150Kw) is finished; the rest (600Kw) is at about three quarters of the upgrade path. In the next stage we will focus mostly on the feasibility of an automatic tagger producing the WOTAN-2 tagset and the interface between WOTAN-2 and AMAZON. It is to be expected that the experiments will lead to adjustments in the tagset, but we hope these to be only minor ones. Another potential source of adjustments is the present development of the wordclass tagset which is to be used for the Spoken Dutch Corpus (*Corpus Gesproken Nederlands*; CGN).

An example

On the page below you find an example utterance tagged with the WOTAN-2 tagset. The first column shows the words, the second the lemma and the third the tags. The utterance reads:

> Van zijn gezicht was die teleurstelling bepaald niet af te lezen,
> From his face was that disappointment certainly not off to read,
> It was certainly not possible to see that disappointment in his face,
>
> want de man uit Eelde is niet iemand die zich snel blootgeeft.
> for the man from Eelde is not someone who himself fast bares.
> for the man from Eelde is not someone who shows his feelings easily.

Example WOTAN-2 Tagging

```
Van           van           Adp(type=prep,Pcompl=obl+dat,Ccompl=obl,infl=unm,
                                Psynuse=adp+nampart+synmark,Csynuse=adp,spel=unm)
zijn          zijn          Pron(Ptype=poss,Ctype=poss,per=unu,numgen=unm,case=unm,
                                pol=unm,str=str,Pprag=poss3smn,Cprag=poss3smn,
                                Psynuse=det,Csynuse=det,spel=unm)
gezicht       gezicht       N(type=com,numgen=singn,case=unm,dim=unm,
                                Psynuse=nom,Csynuse=nom,spel=unm)
was           zijn          V(stat=aux,Ptype=pass+perf,Ctype=pass,form=past,
                                Pconc=sing,Cconc=sing,infl=unu,aux=zijn,sep=nonsep,
                                Psynuse=verb,Csynuse=verb,spel=unm)
die           die           Pron(Ptype=dem,Ctype=dem,per=unu,numgen=nonsingn,
                                case=unm,pol=unu,str=unu,Pprag=unm,Cprag=unm,
                                Psynuse=nom+det,Csynuse=det,spel=unm)
teleurstelling teleurstelling N(type=com,numgen=singmf,case=unm,dim=unm,
                                Psynuse=nom,Csynuse=nom,spel=unm)
bepaald       bepaald       Adj(deg=pos,infl=unm,dim=unm,Psynuse=adj+adv,Csynuse=adv,
                                spel=unm)
niet          niet          Adv(type=gener,Pfunc=neg,Cfunc=neg,deg=pos,infl=unm,
                                Psynuse=adv,Csynuse=adv,spel=unm)
af            aflezen       Adp(type=vprt,Pcompl=unu,Ccompl=unu,infl=unm,
                                Psynuse=adv,Csynuse=adv,spel=unm,
                                linkid=1,linktype=beforevc)
te            te            Uniq(type=inf-te,Psynuse=synmark,Csynuse=synmark,
                                spel=unm)
lezen         aflezen       V(stat=lex,Ptype=trans,Ctype=trans,form=infin,
                                Pconc=unu,Cconc=unu,infl=unm,aux=unm,sep=sepcore,
                                Psynuse=verb,Csynuse=verb,spel=unm,
                                linkid=1,linktype=nfinvpos)
,             &comma;       Punc(type=comma)
want          want          Conj(type=coord,subtype=simp,Psynuse=conj,Csynuse=conj,
                                spel=unm)
de            de            Art(type=def,numgen=nonsingn,case=unm,
                                Psynuse=det+nampart,Csynuse=det,spel=unm)
man           man           N(type=com,numgen=singmf,case=unm,dim=unm,
                                Psynuse=nom,Csynuse=nom,spel=unm)
uit           uit           Adp(type=prep,Pcompl=obl+dat,Ccompl=obl,infl=unm,
                                Psynuse=adp,Csynuse=adp,spel=unm)
Eelde         Eelde         N(type=prop,numgen=sing,case=unm,dim=unm,
                                Psynuse=nom+nampart,Csynuse=nom,spel=unm)
is            zijn          V(stat=cop,Ptype=unu,Ctype=unu,form=pres,
                                Pconc=s3,Cconc=s3,infl=unu,aux=zijn,sep=nonsep,
                                Psynuse=verb,Csynuse=verb,spel=unm)
niet          niet          Adv(type=gener,Pfunc=neg,Cfunc=neg,deg=pos,infl=unm,
                                Psynuse=adv,Csynuse=adv,spel=unm)
iemand        iemand        Pron(Ptype=indef,Ctype=indef,per=unu,numgen=unm,case=unm,
                                pol=unu,str=unu,Pprag=exist,Cprag=exist,
                                Psynuse=nom,Csynuse=nom,spel=unm)
die           die           Pron(Ptype=rel+indrel,Ctype=rel,per=unu,numgen=nonsingn,
                                case=unm,pol=unu,str=unu,Pprag=unm,Cprag=unm,
                                Psynuse=nom,Csynuse=nom,spel=unm)
zich          zich          Pron(Ptype=refl,Ctype=refl,per=third,numgen=unm,case=obl,
                                pol=unu,str=unu,Pprag=unm,Cprag=unm,
                                Psynuse=nom,Csynuse=nom,spel=unm)
snel          snel          Adj(deg=pos,infl=unm,dim=unm,Psynuse=adj+adv,Csynuse=adv,
                                spel=unm)
blootgeeft    blootgeven    V(stat=lex,Ptype=refl,Ctype=refl,form=pres,
                                Pconc=s2+s3,Cconc=s3,infl=unu,aux=hebben,sep=septot,
                                Psynuse=verb,Csynuse=verb,spel=unm)
.             &period;      Punc(type=period)
```

Important Changes from WOTAN-1

Notation

The most obvious, but also most trivial changes are the notational ones. WOTAN-1 has a straightforward two-level notation, where the meaning of an attribute is determined by its value, its position and possibly the preceding attributes, e.g. `N(soort,ev,neut)`. Attribute values are expressed in Dutch. WOTAN-2 is meant to be much more flexible. There is an internal form, as shown in the example text, which uses (EAGLES inspired) English terminology and explicit attribute-value pairs. However, the idea is that the supporting software will allow the user to choose between English and Dutch terminology and to switch off the presentation of attribute names. It should also be possible to leave out all values which are unused, unmarked and/or unknown (see below). This will lead to representation such as:

```
man  N(type=com,numgen=singmf,case=unm,dim=unm,Psynuse=nom,Csynuse=nom,spel=unm)
man  N(com,singmf,unm,unm,nom/nom,unm)
man  N(com,singmf,nom)
```

Attributes

As can be expected, most of the differences are found in the presence of the encoded attributes and their ranges. A full comparison is shown elsewhere on the poster, but we will highlight some of the more important differences here:

- The closed classes are completely redesigned to be as compatible as possible with the ANS-97. The differences from WOTAN-1 concern both the incompatibilities between WOTAN-1 and ANS-84 and the revisions from ANS-84 to ANS-97. An example where the two types interact is the treatment of what used to be indefinite numerals and are now mostly indefinite pronouns.
- WOTAN-2 distinguishes between auxiliary and copular verbs. Since these classes largely overlap and it appears that syntactic or even semantic analysis is needed to make the distinction, WOTAN-1 underspecified it. However, its importance made us include it and experiments will show how well an automatic tagger will be able to cope.
- Much the same thing can be said for the distinction between interrogative, relative and independent relative use of pronouns and adverbs, which WOTAN-2 encodes more fully than WOTAN-1.

Separated verbs

A change which goes beyond the individual token attributes is the treatment of separable verbs. In WOTAN-1, separated verb particles were marked as such (most of the time), but the corresponding verbal token was not especially marked, let alone that the two were linked. Since WOTAN-2 is meant to be used in conjunction with lemmatization, we felt it necessary to add such a linking mechanism. For instance, take the verb “aflezen” (“off-read”: “read off”), which is present as “af” (“off”) and “lezen” (“read”) in the example text. The verb part is marked `sep=sepcore` and the particle `type=vprt`. In addition, the two are linked with `linkid=1` and their positioning in the clause marked with `linktype`, in the case at hand the verb takes a standard verbal position in the non-finite clause (`linktype=nfinvpos`) and the particle is placed just before the verbal cluster (`linktype=beforevc`).

Relation to EAGLES

Attributes

The EAGLES guidelines provide one shared list of attributes for the whole set of European languages. This means that any specific tagset, for a specific language, will probably deviate somewhat from this list. WOTAN-2 certainly does. First of all, there are several EAGLES attributes which are not encoded in WOTAN-2. Voice for verbs and Gender for nouns, pronouns and adjectives are not encoded as Dutch does not mark for these properties. Inflection Type for adjectives is not encoded as we feel it to be a property of the lemma rather than of the wordform, and hence more at home in the lexicon than in a tagged corpus. The adjectival attributes Use and NP Function are only partly covered (by WOTAN-2's `synuse`) as we feel the additional information belongs in syntactic analysis rather than in tagging. Finally, there is the Countability of nouns, which we would very much like to add, but for which we lack the lexical resources.

There are also additions to EAGLES. Some due to morphological processes being active in more wordclasses (following the classification in ANS-97) than described by EAGLES. Examples are inflection for verbs, adverbs, adpositions and numerals, degree for numerals and plural and genitive forms of residuals. A process not described at all by EAGLES is diminutive formation, which for Dutch is possible with nouns, adjectives and numerals. Other additions are due to a higher granularity in the attributes, e.g. the types of residual and the pragmatic properties of pronouns, or different classification criteria, e.g. adposition types, definiteness of numerals and R-adverbs. The last additions are due to an extended analysis at the syntactic level, e.g. the treatment of separable verbs, complementation for adpositions and `synuse`, or the lexical level, viz. spelling conformity.

Finally, WOTAN-2 sometimes combines several EAGLES attributes into a single attribute. The two reasons for this, which are almost always both active, are to keep close to the ANS tradition and to allow easier underspecification. Combinations are the number/gender system for nouns, pronouns and articles, verb form and concord, inflection for adjectives, the (sub)type system for verbs, pronouns and conjunctions, and the function system for adverbs.

Underspecification

A completely different type of deviation from EAGLES is the treatment of under-specification. EAGLES allows underspecified values, but only as replacement of actual values. We feel that there are situations where the potential value (i.e. that found in the lexicon) and the contextual one (i.e. that selected by a linguist or tagger) can both be useful, e.g. the potential value can serve as backup in case a syntactic analysis is blocked by the selected contextual one. As a result, for a number of attributes, both values are encoded, e.g. the word "die" ("that") near the end of the example text can be either relative or independent relative (`Ptype=rel+indrel`), but in context is found to be relative (`Ctype=rel`).

A related point is the treatment of unspecified attributes. The intermediate tagset of EAGLES uses the value 0 for all underspecified values. In WOTAN-2, we differentiate between different situations. The value `unu` is used if the attribute does not apply at all, e.g. concord for participle verbs. The value `unm` is used if the wordform is unmarked for the attribute, e.g. "lopen" is unmarked for auxiliary selection, i.e. allows both "zijn" and "hebben". Also, all base forms are unmarked for morphological properties, e.g. "huis" is unmarked for diminutive. The value `unk`, finally, is used when the value is as yet unknown, e.g. after lexicon lookup, but before disambiguation, only the potential values are known and the contextual ones have not yet been selected.

The Tagsets Compared: open classes

Class:Attribute	Number of Values in WOTAN-2	Number of Values in WOTAN-1	Number of Values in EAGLES
N:type	2	2	2
N:number+gender	6	2 (only number)	2 (number) 4 (gender)
N:countability	- (considered)	-	2
N:case	3	3	7
N:diminutive	2	-	-
N:synuse	2 [PC]	-	-
V:status	3 (lex, aux, cop)	2.5 (cop underspec.)	3 (lex, aux, semi-aux)
V:type+compl	13 [PC]	4	2 (refl) 2 (aux-function)
V:form	8	7	2 (finiteness) 9 (form/mood) 4 (tense) 2 (aspect)
V:voice	- (unm in Dutch)	-	2
V:concord	7 [PC]	4 (person) 2 (number)	4 (person) 2 (number)
V:gender	- (unm in Dutch)	-	3
V:inflection	5	4	-
V:aux selection	3	-	3
V:separability	3	-	2 (only potential)
V:synuse	5 [PC]	2	-
Adj:degree	3	3	3
Adj:gender	-	-	3
Adj:inflection	6	4	2 (number) 6 (case)
Adj:infl-type	-	-	3
Adj:diminutive	2	-	-
Adj:synuse	5 [PC]	3	2 (use) 3 (NP function)
Adv:type	3	4 (incl Adp types)	4 (incl Adp types)
Adv:function	8 [PC]	6	2 (polarity) 3 (wh-type)
Adv:degree	3	3	3
Adv:inflection	3	2	-
Adv:synuse	1 [PC]	-	-
Misc:type	9	3	6
Misc:form	3	-	2 (number only)
Misc:gender	- (unm in Dutch)	-	3
Misc:synuse	11 [PC]	-	-
ALL:spelling	11	-	-

The Tagsets Compared: closed classes

Class:Attribute	Number of Values in WOTAN-2	Number of Values in WOTAN-1	Number of Values in EAGLES
Pron:type	10 [PC]	8	3 (category) 5 (pron-type) 5 (det-type) 3 (spec-type) 3 (wh-type)
Pron:person	3	3	3
Pron:number+gender	9	3 (only number)	2 (number) 4 (gender)
Pron:case	5	5	7
Pron:politeness	3	-	2
Pron:strength	3	-	2
Pron:pragmatics	25 [PC]	2	2 (poss-number)
Pron:synuse	5 [PC]	2	-
Art:type	2	2	3
Art:number+gender	3	4	2 (number) 4 (gender)
Art:case	3	3	6
Art:synuse	3 [PC]	-	-
Adp:type	5	4 (incl Uniq(te))	4
Adp:compl	4 [PC]	-	-
Adp:inflection	2	-	-
Adp:synuse	3 [PC]	-	-
Conj:type	2	2	2
Conj:subtype	8	2	4 (coord) 3 (subord)
Conj:synuse	2 [PC]	-	-
Num:type	2	2	2
Num:definiteness	2	2	-
Num:degree	3	3	-
Num:inflection	6	4	2 (number) 4 (case)
Num:gender	- (unm in Dutch)	-	3
Num:diminutive	2	-	-
Num:synuse	4 [PC]	2	3 (function)
Int:synuse	1 [PC]	-	-
Uniq:type	1	0 ("te" = Adp)	7
Uniq:synuse	1 [PC]	-	-
Punc:type	15	18	*varies*
ALL:spelling	11	-	-

Conflicts of Interest

Closed classes

As stated, WOTAN-2 is meant to be as compatible as possible with ANS-97, EAGLES, CELEX and AMAZON. Also, for the existing WOTAN-1 tagging to be of any use, some measure of compatibility between WOTAN-2 and WOTAN-1 would be beneficial as well. As all of these classification systems developed more or less independently, it is clear that they are not even completely mutually compatible. Wherever there are differences in classification, choices had to be made for WOTAN-2.

The incompatibilities are most pronounced in the closed classes. Here, however, there was a clear fundamental choice: use the classification from the ANS. The ANS provides full lists of items and criteria which can be used directly, i.e. we get an instant manual, and can be seen as representing a kind of Dutch descriptive linguistic consensus. CELEX is not a viable alternative as it admittedly does not describe the closed classes as well as the open ones. Furthermore, by using ANS, AMAZON is covered as it also aims at ANS compatibility and EAGLES conformance is reasonable well catered for as the differences mostly lie in the more vaguely described distinctions or ones not present for Dutch. The central distinctions of the ANS are present directly in WOTAN-2 as either single attributes or combined attributes. Some of the more peripheral ones are partially present, mostly when useful for AMAZON, e.g. the `prag` attribute for pronouns.

There are a few areas where WOTAN-2 does not follow the ANS, for various reasons. For example, because of tagset consistency, all adposition uses are grouped together under `Adp`, rather than spread out, e.g. particles to `Adv`. Because of recent linguistic insights, the R-pronomina (a group with both adverbial and pronominal properties: “er”, “hier”, “daar”, “waar”, “ergens”, “nergens”, “overal”) are given a special status, `Adv(type=R,…)`, although their major class remains adverb. Because of the fundamental decision to primarily tag form rather than function, all numerical forms are tagged `Num`, even those which are often classified as nouns, e.g. “miljoen”.

Open classes

For the open classes, the choice in situations where the different classifications clash must be taken on a much more pragmatic basis. The ANS may well provide criteria, but it does not provide exhaustive lists. For this we have to work with CELEX. Unless we want to invest in the enormous task of extending (or replacing) the open classes of the lexicon, using the CELEX classification is the only option. An example of an attribute where this leads to dissatisfaction is lexical verb complementation, where CELEX gives only a single attribute with three values (intransitive, transitive and reflexive), where we (like EAGLES) would prefer independent attributes for transitivity and reflexiveness. We may yet follow our preference, but this would mean checking some 12,000 verbs from CELEX alone.

There are several other distinctions which are (for now) left out of WOTAN-2 because they would mean too much work. Another lexicon-related example is countability, which practically everybody would like to see tagged. An example of a distinction which is determined by the context is adjective inflection, where now only the form is tagged, but not the reason for the form, e.g. +e is used both before plural and neutral singular forms, and WOTAN-2 tags `infl=e` rather than `numgen=plu` or `numgen=singn`. Again, we may yet introduce these distinctions if we think the added value justifies the effort.

Upgrading the Eindhoven Corpus

General approach

Because of the redefinition of closed classes and the changes in criteria for open classes, it is impossible to create a simple translation table of WOTAN-1 tags to WOTAN-2 tags. Instead, we start with a normal lexicon lookup procedure and then use the WOTAN-1 tags as a kind of filter. Whenever there is ambiguity in the potential WOTAN-2 tags, and the WOTAN-1 tag is compatible with one or more of those, all incompatible tags are removed. Also, if an unknown open class token is encountered, we base the WOTAN-2 tag on the old one, where possible.

However much this filter helps, it is not sufficient by itself. Manual selection is unavoidable. One strategy we followed to make this more efficient is that we let the selectors work with a limited amount of tagging information. For specific tasks, e.g. the choice between auxiliary verb and copula, only the words and the question at hand are given. For more general tasks, the upgrade process goes through three levels of specificity: 1) only the potential part of PC attributes is used, 2) the contextual value is added, except for synuse and 3) the full tags are used. This avoids distraction of the selector and also keeps ambiguity (and file size) in check.

Resources for disambiguation

The main resource for ambiguity reduction is the WOTAN-1 tagging. Some tag parts can be used directly, e.g. major wordclass and verb form. However, direct use is only possible if there is indeed a WOTAN-2 tag with the corresponding tag part. Not all filtering is this direct. More complex examples are the selection of `Adj(…,synuse=adv)` if the WOTAN-1 tag is `Adv(…)` or of the particle part of a separable verb for a WOTAN-1 `Adv(deel_v)`.

Another source of automatic disambiguation is the context. If there is no finite verb in the rest of the utterance, a subordinator can only be `subtype=withoutv`. A very effective contextual disambiguation is the removal of parts of separable verbs (for which the lexicon is extremely productive) after the linked part has been removed on the basis of the WOTAN-1 tagging.

Finally, there is manual disambiguation. There are a few systematic ambiguities which have to be dealt with manually, e.g. distinctions which are not present in WOTAN-1. The most explainable of these were done by students using the mentioned single-choice mechanism, viz. auxiliary vs copula (18,500 cases), interrogative vs relative vs independent relative vs exclamatory (7,100 cases), coordinator subtype (5,000 cases), subordinators with or without finite verb (2,900 cases) and a few smaller sets. Apart from these systematic choices, there are innumerable unsystematic choices to be made all through the disambiguation process.

Actual ambiguity and its reduction

The table below gives an impression of the ambiguity level during the upgrade process. The bottom line shows the current state for the whole corpus. The newspaper subcorpus has been completed, i.e. all remaining ambiguity has been removed, the unknown tokens correctly tagged and a number of errors corrected.

Stage	Tokens with 1 tag	Tokens with 2 tags	Tokens with 3-5 tags	Tokens with 6-10 tags	Tokens with >10 tags	Average number tags/token
After lexicon (no contextual values)	362,671	132,563	186,406	66,917	5,514	2.45
After level 1 disambiguation	685,182	43,330	24,991	367	1	1.14
Contextual values added (no synuse)	584,085	123,173	45,385	1,212	16	1.32
After level 2 disambiguation so far	745,041	6,172	2,477	180	1	1.02