Technical Reports on Language Technology
Report No 59, 20020
http://www.njas.helsinki.fi/salama

Disambiguation strategy of English text[1]

Arvi Hurskainen
Department of World Cultures, Box 59
FIN-00014 University of Helsinki, Finland
arvi.hurskainen@helsinki.fi

Abstract

The analysed English text contains exceptionally much ambiguity, compared with the analysis of many other languages. This is mainly due to the limited number of morphological features in the language. Although disambiguation is a separate operation in language technology, it cannot be detached from morphological analysis. In resolving disambiguation problems, we often need to correct or enhance the morphological lexicon. Therefore, the developer of the technology must know thoroughly the structure of the lexicon and be able to write all types of disambiguation rules. And, of course, the developer must have access to the development environments of both components. In this report I will discuss various disambiguation problems. The analysis component is based on finite-state methods and the disambiguation component uses Constraint Grammar rules.

Key Words: *morphological analysys, disambiguation, machine translation.*

1 Introduction

Due to the limited morphology of English language, the disambiguation is a major challenge. Languages with rich morphology offer possibilities for such language description, where most word-forms can be uniquely encoded on the basis of the word structure, and much less work will be left for disambiguation, where ambiguity is solved on the basis of the sentence structure. English is not such a language.

Furthermore, English users have a tendency to omit sentence boundary markers, which further aggrevates the disambiguation process.

The amount of ambiguity depends also on how fine-grained the analysis is. The general rule is that each new level of detail in analysis is likely to increse ambiguity. A low-level analysis system is not a solution, especially if the aim is machine translation, because the features, if left undescribed in analysis, must be addressed and resolved in a later phase of processing. Therefore, in most cases, it is advisable to make the morphological analysis as complete as possible, even with the risk of added ambiguity.

In this report I will discuss the strategy of designing a morphological disambiguation system for English. In the process of constructing the disambiguation system, corrections

[1] The report is issued under licence CC BY-NC

to the analysis system are often needed. Such close interaction results in the optimal structure of the whole system.

2 Using defaults in analysis

The need of disambiguation rules can be reduced so that we construct the morphological lexicon in such a way that the most likely interpretation will be printed as the first choice. This can be done by cotrolling the order, where the analyser processes various interpretations of the word. One must be, however, careful in changing the order of output, because a changed order may affect the behaviour of disambiguation rules drastically. For example, if verb interpretations and noun interpretations are in opposite order, the default readings change. This may require an entirely different approach in rule writing. Nevertheless, careful consideration of defaults before writing disambiguation rules reduces the need of rules.

3 Arrangement of disambiguation rules

Although no strict guidelines for writing disambiguation rules can be given, there are some guiding principles.

Rules can be arranged into sets so that reliable rules come first, and less reliable rules follow in the order of reliability. The least reliable rules come last.

In the Constraint Grammar rule system, rule sets can be separated into blocks using the reserved word CONSTRAINTS. Rules are not applied strictly in the order, where they are in the rule lexicon. The application order within one block is arbitrary, and the application order cannot be controlled by simply putting the rules into a certain order. However, all rules between two block names CONSTRAINTS will be applied first, and only then the control moves to the next rule block.

Examples of reliable rules that can be placed into the first rule block are in (1).

(1)

```
REMOVE V (-1 (DET));
REMOVE INFMARK (1 (SG3));
SELECT QUEST (*1 QUESTMARK BARRIER SNTB);
```

The first rule removes all verb readings, if the first word on the left is a determiner (DET).

The second rule removes the reading with infinitive marker, if the next word includes the third person singular tag (SG3).

The third rule selects the question reading, if somewhere in the sentence (usually in the end) is a question mark.

These rules do not do much, but because of simplicity and reliability they are useful.

Any number of such blocks can be constructed. However, it is often difficult to decide the optimal rule order. Rule writing proceeds normally so that the rule first applies to the context, where the need of rule was first found. Then, if the rule has wrong applications, further constraints are added.

Examples of such complex rules are in (2).

(2)

```
SELECT V + (PL3) (*-1 N + PL OR PRON + PL OR NUM BARRIER CLB OR N
OR PRON) (NOT 0 N OR A) (NOT -1 INF OR DET) (NOT -2 ("of"));
SELECT V + (PL3) (*-1 N BARRIER CLB LINK -1 CC LINK -1 N LINK NOT
*1 VFIN) (NOT -1 V);
```

It may also happen that the rule, initially written in some context, requires repeatedly more constraints. If this happens, it is likely that such a rule should be removed and another approach tried.

It is not needed that every word has only one reading after disambiguation. The requirement is that the first reading is the correct one. Other readings will be removed after disambiguation and only the first one is left.

There are cases, where reliable disambiguation is very difficult. In such cases we can mark certaing interpretations as preferable. Consider the example in (3).

(3)

```
"<*during>"
      "during" PREP CAP

"<that>"
      "that" CONJ REL
      "that" CONJ
      "that" PRON DEM SG

"<time>"
      "time" V vt PRES SG1
      "time" V vt PRES SG2/PL2
      "time" V vt PRES PL1
      "time" V vt PRES PL3
      "time" V vt INF
      "time" V vt IMP
      "time" PREFR N SG

"<he>"
      "he" PRON MALE SG3

"<slept>"
      "sleep" V vi PAST
      "sleep" V vi EN
      "slept" A

"<.>"
      "." **CLB
```

After running the disambiguation rules, we get the result as in (4).

(4)

```
"<*during>"
      "during" PREP CAP
"<that>"  S:142/1
      "that" PRON DEM SG
"<time>"  S:321/3, 244/2, 125/1
      "time" PREFR N SG
"<he>"
      "he" PRON MALE SG3
"<slept>"  S:155/3
      "sleep" V vi PAST
"<.>"
      "." **CLB
```

We take a look at how the word *time* was disambiguated. Three rules have applied.
The first one is rule number 125.

```
REMOVE V + PERS1-2 (NOT *-1 PRON + PERS1-2 BARRIER CLB) (NOT -1
INFMARK);
```

The rule removes the verb readings in first and second person singular and plural.

The second rule is the rule number 244.

```
REMOVE (IMP) (*1 EN BARRIER CLB) (NOT -1 SNTB) (NOT -2 SNTB);
```

The rule removes the imperative interpretation. After the application of these two rules, the reading for *time* looks like in (5).

(5)

```
"<time>"
      "time" V vt PRES SG1
      "time" V vt PRES PL1
      "time" V vt PRES PL3
      "time" V vt INF
      "time" PREFR N SG
```

We still have verb readings left, and the desired noun reading is the last one. In cases such as this we can make use of the special tag PREFR. The last rule block of the rule file has only one rule.

```
SELECT (PREFR);
```

This rule selects the reading, which has the tag PREFR. There are no constraints.
The rule is not fully reliable, but it works correctly in most cases. The operation of the rule can be considered as moving the reading into the first place in the cohort, whereby it is left as final reading, when other interpretations are removed.

4 Ambiguity of analysis result

The sentence in (6) contains three words, each of which can be interpreted as a verb or noun. In addition, the verb may have several separate interpretations.

(6)

```
"<*finals>"
      "final" CAP N PL

"<will>"
      "will" AUXV PRES SG3
      "will" V AUXMOD
      "will" AUXV PRES SG1
      "will" AUXV PRES SG2/PL2
      "will" AUXV PRES PL1
      "will" AUXV PRES PL3
      "will" AUXV INF
      "will" AUXV IMP
      "will" N SG

"<take>"
      "take" V vt PRES SG1
      "take" V vt PRES SG2/PL2
      "take" V vt PRES PL1
      "take" V vt PRES PL3
      "take" V vt INF
      "take" V vt IMP
      "take" N SG

"<place>"
      "place" V vt PRES SG1
      "place" V vt PRES SG2/PL2
      "place" V vt PRES PL1
      "place" V vt PRES PL3
      "place" V vt INF
      "place" V vt IMP
      "place" N SG

"<tomorrow>"
      "tomorrow" N SG

"<.>"
      "." **CLB
```

When we disambiguate the sentence, the result is as in (7). The result also displays the rule numbers (actually they are line numbers) in the rule file, which are responsible for the disambiguation of each word.

(7)

```
"<*finals>"
```

```
      "final" CAP N PL
"<will>"  S:106/1
      "will" AUXV PRES PL3
"<take>"  S:118/1
      "take" V vt INF
"<place>"  S:233/2, 125/1
      "place" N SG
"<tomorrow>"
      "tomorrow" N SG
"<.>"
      "." **CLB
```

Let us take a look at these rules in detail. First, we have the rule, which is on line 106 in the rule file (8).

(8)

```
SELECT V + (PL3) (-1 PRON + PL OR N + PL) (NOT -1 ("news") OR GEN)
(NOT 0 ("back")) (NOT -2 ("of")) (NOT *-1 V BARRIER CLB);
```

The rule runs: Select the verb reading with third person plural. The conditions are: (a) In the first cohort to the left must be a pronoun in plural form or a noun in plural form. (b) In the first cohort to the left there should not be the word *news* or a genitive form. (c) The target word should not be *back*. (d) The second word to the left should not be *of*. (e) On the left there should not be a verb. Do not scan beyond clause boundary.

Note that if the tag is within parentheses, it is a real tag in alaysis result. The tag without parentheses is a set name, which includes one or more members.

Actually, the rule defines the conditions for a finite verb. The verb can be the only verb in the sentence, or an auxiliary verb, followed by the main verb, as is the case in our example sentence.

The following word *take* is disambiguated as an infinitife form of a verb. The rule is in (9).

(9)

```
SELECT (INF) (-1 ("help") OR AUXV OR INFMARK) (NOT -1 BE);
```

The rule runs: Select the infinitive form, if the first word to the left is *help*, or it is an auxiliary verb, or the infinitive marker *to*. The first word to the left should not be the verb *be*.

It is a simple rule and as such fairly secure. However, it applies only to a small number of cases, where infinitive must be selected.

Next, we will see how the word *place* was disambiguated. Two rules were applied to this word. The rules are in (10).

(10)

```
REMOVE V + PERS1-2 (NOT *-1 PRON + PERS1-2 BARRIER CLB);
REMOVE V (-1C V) (NOT -1 AUXV OR ("help") OR ("get")) (NOT 0 ING);
```

The first rule removes the verbs, which have the tag of the first or second person, singular or plural. The condition is that somewhere on the left there must be a pronoun of the first or second person (*I*, *you* or *we*).

The second rule is more covering. It removes all verb readings. The conditions are: The first word to the left should be uniquely a verb. However, it should not be an auxiliary verb, or the verb *help* or *get*. The word should not be in gerund form.

5 Words with three POS interpretations

In English, the gerund form of verb may also be a noun or adjective. In the current system, the morphological analyser was so constructed that gerund verb forms also produce noun and adjective interpretation. Consider the example in (11).

(11)

```
"<following>"
      "follow" V vt vi ING
      "following" N SG
      "following" A
```

All three interpretations are included in the sentences in (12).

(12)

```
"<*he>"
      "he" PRON CAP MALE SG3

"<was>"
      "be" AUXV PAST SG3
      "be" AUXV PAST SG1

"<following>"
      "follow" V vt vi ING
      "following" N SG
      "following" A

"<them>"
      "they" PRON PL3 ACC

"<.>"
      "." **CLB

"<*he>"
      "he" PRON CAP MALE SG3

"<had>"
      "have" AUXV PAST
      "have" AUXV EN

"<a>"
      "a" DET INDEF
```

```
"<following>"
      "follow" V vt vi ING
      "following" N SG
      "following" A

"<.>"
      "." **CLB

"<*the>"
      "the" DET CAP DEF

"<following>"
      "follow" V vt vi ING
      "following" N SG
      "following" A

"<example>"
      "example" N SG

"<explains>"
      "explain" V vt PRES SG3

"<it>"
      "it" PRON SG3
      "it" PRON SG3 ACC

"<.>"
      "." **CLB
```

The sentences are disambiguated in (13).

(13)

```
"<*he>"
      "he" PRON CAP MALE SG3
"<was>"  S:125/1
      "be" AUXV PAST SG3
"<following>"
      "follow" V vt vi ING
      "following" N SG
      "following" A
"<them>"
      "they" PRON PL3 ACC
"<.>"
      "." **CLB
"<*he>"
      "he" PRON CAP MALE SG3
"<had>"  S:155/1
      "have" AUXV PAST
"<a>"
      "a" DET INDEF
```

```
"<following>"  S:149/1, 103/1
      "following" N SG
"<.>"
      "." **CLB
"<*the>"
      "the" DET CAP DEF
"<following>"  S:108/1
      "following" A
"<example>"
      "example" N SG
"<explains>"
      "explain" V vt PRES SG3
"<it>"  S:217/2
      "it" PRON SG3 ACC
"<.>"
      "." **CLB
```

In the first example, the word *following* was not disambiguated. Here we rely on the default, that is, the first reading is selected if no rule applies. When the first reading is correct, no disambiguation is needed.

This is also an example of how the output of the analyzer should be designed. The word *following* is most often a verb. Therefore, this interpretation must be produced first, and only after that other interpretations follow.

In the second sentence, the word *reading* has been subject to two rules. The first rule (No. 103) runs:

```
REMOVE V (-1 (DET));
```

The verb readings are removed, if a determiner precedes.

The second rule (No. 149) runs:

```
SELECT N (-1 DET) (NOT 1C N);
```

The noun reading is selected, if the first word on the left is a determiner. The first word on the right should not be uniquely a noun.

In the third sentence, the word *following* is interpreted as an adjective. The rule No. 108 runs:

```
SELECT A (-1 DET) (1C N OR NUM) (NOT 0 ("number") OR ("voting"));
```

The adjective reading is selected, if the first word on the left is a determiner. The first word on the right should be uniquely a noun or number. The word should not be *number* or *voting*.

In all three examples we found different readings for the word *following*. For all gerund forms, the analysis system produces three POS readins, verb, noun and adjective. All three do not occur in text for all verbs. Therefore, there is sometimes a need to use a shortcut for avoiding complex rules. Consider the example in (14).

(14)

```
"<*you>"
      "you" PRON CAP SG2/PL2
      "you" PRON CAP SG2/PL2 ACC

"<are>"
      "be" AUXV PRES

"<amazing>"
      "amaze" V vt ING
      "amazing" N SG
      "amazing" A
      "amazing" PREFR A

"<.>"
      "." **CLB
```

The word *amazing* has, in addition to the three interpretations, also an interpretation for adjective with the tag PREFR. The interpretation with PREFR is listed directly into the adjective lexicon. Because the word *amazing* occurs almost exclusively as an adjective, it can be selected directly (15). For other similar adjectives, normal rule writing might be necessary.

(15)

```
"<*you>"  S:241/2
      "you" PRON CAP SG2/PL2
"<are>"
      "be" AUXV PRES
"<amazing>"  S:321/3
      "amazing" PREFR A
"<.>"
      "." **CLB
```

If the added tag PREFR for the word *amazing* is missing in the lexicon, the disambiguation would be as in (16).

(16)

```
"<*you>"  S:242/2
      "you" PRON CAP SG2/PL2
"<are>"
      "be" AUXV PRES
"<amazing>"
      "amaze" V vt ING
      "amazing" N SG
      "amazing" A
"<.>"
      "." **CLB
```

The disambiguation system would rely on the default interpretation, that is, the word *amazing* is interpreted as verb. And when that interpretation is the first one, no rules are needed, and the default interpretation is selected.

The tag PREFR can help also in more complex disambiguation cases, not only as the last resort when disambiguation cannot be resolved with context sensitive rules. Consider the example in (17).

(17)

```
"<the>"
      "the" DET DEF

"<patient>"
      "patient" PREFR N SG
      "patient" A

"<record>"
      "record" V vt PRES SG1
      "record" V vt PRES SG2/PL2
      "record" V vt PRES PL1
      "record" V vt PRES PL3
      "record" V vt INF
      "record" V vt IMP
      "record" N SG

"<installation>"
      "installation" N SG
```

The word *patient* in this context would normally be interpreted as an adjective. Now, when the noun reading has the tag PREFR, it can be referred to in rules for avoiding wrong choices.

6 The granularity of analysis

The morphological analyzer can be designed with various degrees of granularity. If we want to develop a translation system, a maximum degree of granularity is preferable. If we leave the granularity to low level in analysis, we must add the needed information later in the process. Consider the analysis result of the forms of the verb *want* (18).

(18)

```
"<want>"
      "want" V vt vi PRES SG1
      "want" V vt vi PRES SG2/PL2
      "want" V vt vi PRES PL1
      "want" V vt vi PRES PL3
      "want" V vt vi INF
      "want" V vt vi IMP
      "want" N SG
```

```
"<wants>"
      "want" V vt vi PRES SG3
      "want" N PL

"<wanted>"
      "want" V vt vi PAST
      "want" V vt vi EN
      "wanted" A

"<wanting>"
      "want" V vt vi ING
      "wanting" N SG
      "wanting" A
```

The first form *want* has separate analyses for the present tense in all persons except for the third person singular. It also has analyses as infinitive and imperative, and also as noun.

The second form *wants* is interpreted as present third person singular, and as noun plural.

The third form *wanted* is interpreted as past tense and participial, and also as adjective. No reference to person is given.

The fourth form *wanting* is interpreted as gerund, noun, and adjective.

We see that the degree of granularity in forms *want* and *wanted* is different. In the first case we can select also the person of the verb form. In the latter case we cannot do that, because person tags are not there. Especially we would need them when translating the past tense form.

There are two solutions for displaying the required information. Solution one is to add the tags in the CG environment. Solution two is to add granularity to the analysis, such as in (19).

(19)

```
"<wanted>"
      "want" V vt vi PAST SG1
      "want" V vt vi PAST SG2
      "want" V vt vi PAST SG3
      "want" V vt vi PAST PL1
      "want" V vt vi PAST PL2
      "want" V vt vi PAST PL3
      "want" V vt vi EN
      "wanted" A
```

A shorter method is to use underspecified analysis, as in (20), and after analysis the underspecified reading is converted to the form as in (19).

(20)

```
"<wanted>"
      "want" V vt vi PAST SG1/SG2/SG3/PL1/PL2/PL3
      "want" V vt vi EN
      "wanted" A
```

It is difficult to say which method is most economical. Inflection tags are needed in any case. If high granylarity is in the morphological analysis, correct choices can be done in morphological disambiguation. If granularity is low, the inflection tags must be added later, and also there we need the same CG environment as in disambiguation.

For the rest of verb forms, infinitive, imperative and gerund, we do not need person tags.

7 MWE isolation in resolving ambiguity

There are cases when disambiguation seems almost impossible. This happens, when there are several consecutive words, each of which has a verb and noun analysis. Consider the example in (21).

(21)

```
"<*the>"
      "*the" DET CAP DEF

"<impact>"
      "impact" V vt PRES SG1
      "impact" V vt PRES SG2/PL2
      "impact" V vt PRES PL1
      "impact" V vt PRES PL3
      "impact" V vt INF
      "impact" V vt IMP
      "impact" N SG

"<of>"
      "of" PREP

"<the>"
      "the" DET DEF

"<new>"
      "new" A

"<*u*s>"
      "u*s" PROPN

"<sanctions>"
      "sanction" V vt PRES SG3
      "sanction" N PL

"<act>"
      "act" V vt vi PRES SG1
      "act" V vt vi PRES SG2/PL2
      "act" V vt vi PRES PL1
      "act" V vt vi PRES PL3
      "act" V vt vi INF
      "act" V vt vi IMP
```

```
        "act" N SG

"<remains>"
        "remain" V vi PRES SG3
        "remain" N PL

"<unclear>"
        "unclear" A

"<.>"
        "." **CLB
```

Each of the words *sanctions*, *act*, and *remains* has a verb and noun interpretation.

We can consider such interpretations as, (a) impact (N) sanctions (V), (b) sanctions (N) act (V), (c) act (N) remains (V), and (d) act (N) remains (N). It is difficult to define the context, because almost every word is ambiguous. In such cases one could consider MWE isolation for part of the ambiguous words. The obvious alternative is the cluster *sanctions act*. Now the sentence can be disambiguated (22).

(22)

```
"<*the>"
        "*the" DET CAP DEF
"<impact>"  S:104/1
        "impact" N SG
"<of>"
        "of" PREP
"<the>"
        "the" DET DEF
"<new>"
        "new" A
"<*u*s>"
        "u*s" PROPN
"<sanctions_act>"
        "sanction_act" N SG
"<remais>"  S:166/1
        "remain" V vi PRES SG3
"<unclear>"
        "unclear" A
"<.>"
        "." **CLB
```

The disambiguation required only one simple rule.

```
SELECT (PRES SG3) (-1 N + SG) (NOT 1 V) (NOT *-1 V BARRIER CLB)
(NOT 0 ("model"));
```

The selection was based mainly on the previous word, which is a MWE noun singular.

8 Conclusion

The high degree of word level ambiguity in English requires careful planning in designing the morhological analyzer and in formulating disambiguation rules. The first requirement is that the analyzer produces all needed forms. However, this is not enough. We must also consider the optimal order of readings, so that the need for rules would be minimal. The clustering of rules into sections helps in separating reliable rules from less reliable ones. We also must consider when we use description with high granularity and when we rely on adding needed tags later.