CG-kurs Romssas 2011

Linda Wiechetek Divvun/Giellatekno / Romssa universitehta

11.-14. oktober 2011

Course program

Bures boahtin!

- Tuesday 11.10.2011
- introduction to the Constraint Grammar formalism
- Wednesday 12.10.2011
- Making Constraint Grammar rules: disambiguation, syntactic analysis
- Thursday 13.10.2011
- Making a grammar checker
- Friday 14.10.2011
- More Constraint Grammar, Apertium, dependencies, planing the future

Why we do the seminar?

- objective of making a grammar checker for Sámi
- within the Constraint Grammar formalism, which we have been using to make a syntactic analyzer, certain modules within the machine translation programs and the pedagogical programs
- teaching the Constraint Grammar formalism to those who are going to contribute to the grammar checker
- and others who are interested

What is Constraint Grammar?

- Fred Karlsson started in the 90s
- used for a large variety of languages, often with an F-Score over 99%
- statistical system often do not achieve that high F-Scores
- the formalism usually takes lemmatized, morphologically analyzed text as an input
- outputs syntactially analyzed and disambiguated text (one reading only)

General facts about Constraint Grammar

- Constraint Grammar is written by linguists
- linguistic intuitions and text examples help the linguist to write linguistic rules
- those rules attach tags to each word (syntactic function, dependency relation, error tags, semantic roles etc.)
- other rules add, remove, select and substitute a reading
- the rules specify a context in which a certain tag is chosen
- a condition, they can specify both positive and negative conditions

General facts about Constraint Grammar

- in the analysis first each word is put in a single line
- below it we see the possible analyses, the regular case is that there is more than one
- there can be 10 or 20 different analyses
- it depends a bit on how general the rules are
- this is mostly done by MAP rules
- after that the different analyses of each word are disambiguated
- that means a particular reading can be selected by a very accurate and specified rule
- alternatively we can step by step remove the readings that cannot be the right ones
- it is difficult to say which is the better way to do it

Constraint Grammar analysis

```
"<Mon>"
	"mun" Pron Pers Sg1 Nom @SUBJ> #1->2
"<lean>"
	"leat" <mv> V IV Ind Prs Sg1 @FMV #2->0
"<okta>"
	"okta" Num Sg Nom @>N #3->4
"<sápmelaš>"
	"sápmelaš" N Sg Nom @<SPRED #4->2
"<.>"
	"." CLB #5->2
```

Constraint Grammar analysis - Some of the tags

- @SUBJ> - subject to the left of its head (the finite verb)
- @FMV - finite main verb (no arrow necessary)
- @>N - premodifier of a noun (here not its own category is specified, but that of the item it modifies)
- @<SPRED - subject predicate to the right of its head

Constraint Grammar analysis - Philosophy of tags

- the finite verb is the head of the sentence: @+FAUXV @+FMAINV (@FMV @FAUX)
- arrow head: information about the dependency mother
- . . . where it is left out it is a verb
- @SUBJ> = @SUBJ>V
- @<SPRED = @V>SPRED
- arrow base: information about the dependency daughter
- . . . where it is left out it can be any category
- @>N - premodifier of a NP (can be an adjective, a demonstrative, a numeral)
- @N< - postmodifier of a NP
- specifying @ART>N, @ADJ>N and @DET>N is not necessary, the morphological tags already say that

Can you identify the following tags?

- `@>ADVL`

Can you identify the following tags?

- @>ADVL
- modifier of an adverbial MAN dávjá
- @<ADVL

Can you identify the following tags?

- @>ADVL
- modifier of an adverbial MAN dávjá
- @<ADVL
- adverbial to the right of the finite verb mun viegan JOĐÁNIT
- @ADVL<

Can you identify the following tags?

- @>ADVL
- modifier of an adverbial MAN dávjá
- @<ADVL
- adverbial to the right of the finite verb mun viegan JOĐÁNIT
- @ADVL<
- Complement of an adverbial. guktii VAHKUS
- @ADVL>

Can you identify the following tags?

- @>ADVL
- modifier of an adverbial MAN dávjá
- @<ADVL
- adverbial to the right of the finite verb mun viegan JOĐÁNIT
- @ADVL<
- Complement of an adverbial. guktii VAHKUS
- @ADVL>
- Adverbial to the left of the finite verb JOĐÁNIT mun viegan

Redundancy in tags

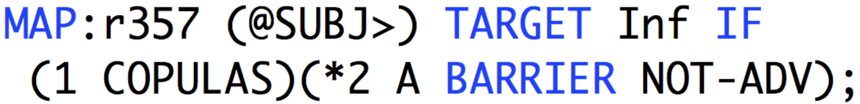
muitalus (sámiid birra) .... lea somá.
@SUBJ> @>P @N< @FMV @SPRED<
@SUBJ>FMV @N>P @N<P @FMV @SPRED<FMV
birra viesu
@P<
@P<N

More tags

Norgga Sámiid
@>N

```
Man dávjá don vuola jugat?  |
@>ADVL @ADVL> @SUBJ> @OBJ> @FMV |
Mun liikon dutnje
@SUBJ> @FMV @<ADVL
guktii vahkus
@<ADVL @ADVL<
Dutnje mun liikon.
@ADVL> @SUBJ> @FMV
```

Constraint Grammar Rules

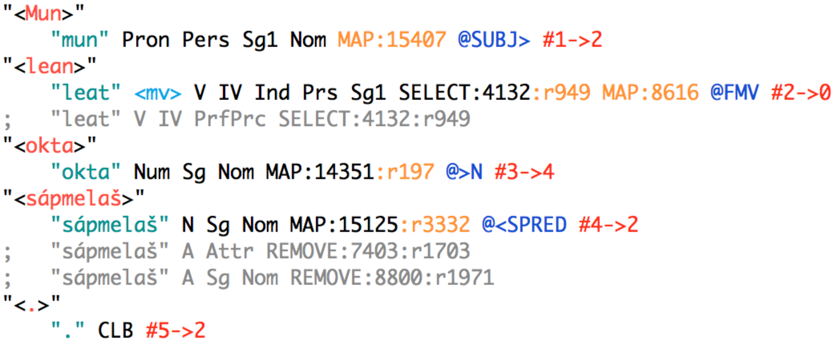


```
MAP:r357 (@SUBJ>) TARGET Inf IF
 (1 COPULAS)(*2 A BARRIER NOT-ADV);
```

Rule types

- the basic rule types for making a shallow grammar are:
- MAP, REMOVE, SELECT, SUBSTITUTE, ADD
- MAP rules annotate syntactic tags to lemmata (can be many, causes ambiguity)
- REMOVE rules take away a whole line below the lemma (making up a possible lemma)
- SELECT rules select one of the mappings and by the discards the other ones
- ADD rules add a mapping
- SUBSTITUTE substitute one mapping by another one

Constraint Grammar analysis with rules applied



```
"<Mun>"
    "mun" Pron Pers Sg1 Nom MAP:15407 @SUBJ> #1->2
"<lean>"
    "leat" <mv> V IV Ind Prs Sg1 SELECT:4132:r949 MAP:8616 @FMV #2->0
;   "leat" V IV PrfPrc SELECT:4132:r949
"<okta>"
    "okta" Num Sg Nom MAP:14351:r197 @>N #3->4
"<sápmelaš>"
    "sápmelaš" N Sg Nom MAP:15125:r3332 @<SPRED #4->2
;   "sápmelaš" A Attr REMOVE:7403:r1703
;   "sápmelaš" A Sg Nom REMOVE:8800:r1971
"<.>"
    "." CLB #5->2
```

Context

- the contexts are specified by numbers
- left contexts: -1 (one position to the left)
- right contexts: 1 (one position to the right)
- *1 at least one position to the right until the first occurence (unbounded context)
- **1 at least one position to the right until whichever occurrence

Context

- What is the difference between (**1 Adv LINK 1 Inf) and (*1 Adv LINK 1 Inf)?
- MAP X TARGET Y IF (*1 Adv LINK 1 Inf)

Context

- What is the difference between (**1 Adv LINK 1 Inf) and (*1 Adv LINK 1 Inf)?
- MAP X TARGET Y IF (*1 Adv LINK 1 Inf)
- X is mapped to Y if the first Adv it finds has an infinitive to its right
- MAP X TARGET Y IF (**1 Adv LINK 1 Inf)
- X is mapped to Y also if the first Adv does not have an infinitve to its right, but if the second/third/etc. does

Context - safe reading

C stands for the safe reading, it means that the word has to have been disambiguated with only the desired reading left

Context - safe reading

- what is the difference between (1 COPULAS) and (1C COPULAS)
- IF (1 COPULAS)

Context - safe reading

- what is the difference between (1 COPULAS) and (1C COPULAS)
- IF (1 COPULAS)
- one of the readings of the word is copula, it can have more readings
- IF (1C COPULAS)

Context - safe reading

- what is the difference between (1 COPULAS) and (1C COPULAS)
- IF (1 COPULAS)
- one of the readings of the word is copula, it can have more readings
- IF (1C COPULAS)
- the only reading that is possible is copula

IFF-rules: SELECT and REMOVE in one

- if the condition is matched, it works as a SELECT-rule
- if the condition is not matched, it works as a REMOVE-rule

IFF targetset (-1* ("someword")) ;

South Sámi IFF-rule

```
IFF:JisPcle ("jis" Pcle) (-1 N OR Pron OR TIME) (NEGATE -1
("buerie") OR ("luste") OR ("seamma") OR ("juktie") OR ("nuelie")
OR ("gaevhtie") OR ("dovne")) ;
```

jis = gis
jis = jos

NOT and NEGATE

- NOT negates (only) the immediate context conditions (to which the adjacent position marker applies)
- NEGATE is used to ”open a negation bracket”, where the remaining (LINK’ed) contexts are negated as a whole

Exercise: NOT and NEGATE

- (0 @SUBJ> LINK NOT 0 @OBJ> LINK 0 HUMAN)

Exercise: NOT and NEGATE

- (0 @SUBJ> LINK NOT 0 @OBJ> LINK 0 HUMAN)
- applies if X is a subject and human, but does not have an object reading
- (0 @SUBJ> LINK NEGATE 0 @OBJ> LINK 0 HUMAN)

Exercise: NOT and NEGATE

- (0 @SUBJ> LINK NOT 0 @OBJ> LINK 0 HUMAN)
- applies if X is a subject and human, but does not have an object reading
- (0 @SUBJ> LINK NEGATE 0 @OBJ> LINK 0 HUMAN)
- applies if X is a subject and human, but does not have an object reading and belongs to the HUMAN set

Refer to absolute positions in the sentence @

- (@) in front of a position number means absolute context
- e.g. @1 for the first token/cohort, @2 for the second, and @-2 for the second-but-last token/cohort in the sentence

AND, OR, +

- OR

AND, OR, +

- OR
- SELECT:r604 Num IF (*-1 ("diibmu") OR ("biellu") BARRIER S-BOUNDARY)(0 ("okci") OR ("vihtta"));
- +

AND, OR, +

- OR
- SELECT:r604 Num IF (*-1 ("diibmu") OR ("biellu") BARRIER S-BOUNDARY)(0 ("okci") OR ("vihtta"));
- +
- concatenation in one and the same reading line
- REMOVE Nom (0 HUMAN + Gen)(*1 BODY BARRIER NPNH) ;
- as old Vislcg AND

AND, OR, +

- OR
- SELECT:r604 Num IF (*-1 ("diibmu") OR ("biellu") BARRIER S-BOUNDARY)(0 ("okci") OR ("vihtta"));
- +
- concatenation in one and the same reading line
- REMOVE Nom (0 HUMAN + Gen)(*1 BODY BARRIER NPNH) ;
- as old Vislcg AND
- intersection, both tags in the same cohort, but not necessarily in the same reading
- has been deprecated in favour of the equivalent LINK 0

AND, OR, +

- OR
- SELECT:r604 Num IF (*-1 ("diibmu") OR ("biellu") BARRIER S-BOUNDARY)(0 ("okci") OR ("vihtta"));
- +
- concatenation in one and the same reading line
- REMOVE Nom (0 HUMAN + Gen)(*1 BODY BARRIER NPNH) ;
- as old Vislcg AND
- intersection, both tags in the same cohort, but not necessarily in the same reading
- has been deprecated in favour of the equivalent LINK 0
- REMOVE:r783 ("siessi") IF (0 ("siessá") LINK 0 Px);

Cohort

everything that belongs to the analysis of a wordform:

sápmelaš
sápmelaš sápmelaš+A+Sg+Nom
sápmelaš sápmelaš+A+Attr
sápmelaš sápmelaš+N+Sg+Nom

Rule types: MAP

```
MAP (@SUBJ> @ACC>) TARGET N OR (PERS NOM) IF (NOT *-1
NON-PRE-N) (1C VFIN) ;
```

Rule types: REMOVE

Removes a reading from a cohort unless this reading is the last surviving reading.

```
REMOVE:r2198 (N Acc) IF (-1 Adv)(0 Gen)(1 (Pron Indef) LINK 1
N);
```

Rule types: SELECT

Selects a reading, selection is equivalent to a removal of all other readings

```
SELECT:r2089 Gen IF (0 ANIMATE)(1 ("dárbu"));
```

ADD rule

```
ADD (@>N) TARGET PrfPrc IF
(*-1 PrfPrc BARRIER NPNHA LINK
0 V-NOT-AUX LINK NOT 0 Actio)
(*1C N BARRIER NOT-ADJ);
```

Rule types: ADD

- an ADD rule can be used if a MAP rule has already mapped all possible readings to a form
- and we want an extra reading for a specific context
- we use ADD rule in sme for the following:

Make ADD-rules

- without looking: what kind of readings could "diibmu" in nominative case get, e.g. "Diibmu lea okta."

Make ADD-rules

- without looking: what kind of readings could "diibmu" in nominative case get, e.g. "Diibmu lea okta."
- now, add an adverbial reading to "diibmu" nominative if it is in the following sentence "Mun boađán diibmu guokte."

Make ADD-rules

- without looking: what kind of readings could "diibmu" in nominative case get, e.g. "Diibmu lea okta."
- now, add an adverbial reading to "diibmu" nominative if it is in the following sentence "Mun boađán diibmu guokte."
- ADD (@ADVL>) TARGET Nom IF (0 ("diibmu") OR ("biellu") OR ("dbm") OR ("dii") OR ("bie"))(1 Num)(*1 FMAINV BARRIER S-BOUNDARY2);

Make ADD-rules

- without looking: what kind of readings could "diibmu" in nominative case get, e.g. "Diibmu lea okta."
- now, add an adverbial reading to "diibmu" nominative if it is in the following sentence "Mun boađán diibmu guokte."
- ADD (@ADVL>) TARGET Nom IF (0 ("diibmu") OR ("biellu") OR ("dbm") OR ("dii") OR ("bie"))(1 Num)(*1 FMAINV BARRIER S-BOUNDARY2);
- make a rule for South Sámi

Rule types: SUBSTITUTE

When do we use substitute? - dependency grammar

Rule

SUBSTITUTE (Plc) (Sur) TARGET Prop IF (-1 (Mal Attr) OR (Fem Attr))(NEGATE 0 Sur);
Linda England.

Rule

SUBSTITUTE (Plc) (Sur) TARGET Prop IF (-2 (Mal Attr) OR (Fem Attr))(-1 ABBR);
Linda B. England.

WEDNESDAY

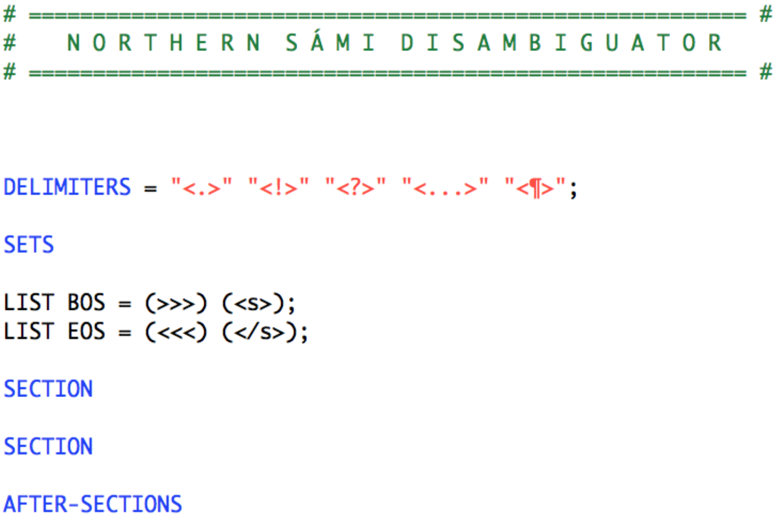
IFF-rules (Ole Brumm "Ja takk begge deler")

- jis = gis Pcle De gis Linda bodii LInda gis nu logai
- jis = jos CC Mon boadán jos Linda boahtá
- IFF:JisPcle ("jis" Pcle) (-1 N OR Pron OR TIME)(NEGATE -1 ("buerie") OR ("luste") OR ("seamma") OR ("juktie") OR ("nuelie")
- if all contextual tests are satisfied it will behave as a SELECT rule
- if the tests are not satisfied it will behave as a REMOVE rule

Structure of the grammar

- A vislcg3 rules file consists typically of the following sections:
- DELIMITERS (1 line, defines sentence boundaries)
- SETS (1 or more sections of set definitions, compiled as one)
- CORRECTIONS (1 section of correction rules, replacing tags anywhere in a reading)
- MAPPINGS (1 section of mapping rules, adding tags at the end of a reading line)
- CONSTRAINTS (1 or more sections of REMOVE or SELECT rules)

structure of the Constraint Grammar analyser



```
# ======================================================== #
#   N O R T H E R N  S Á M I  D I S A M B I G U A T O R
# ======================================================== #


DELIMITERS = "<.>" "<!>" "<?>" "<...>" "<¶>";

SETS

LIST BOS = (>>>) (<s>);
LIST EOS = (<<<) (</s>);

SECTION

SECTION

AFTER-SECTIONS
```

existing CGs for Sámi - how many rules

- **sme-dis.rle** - 765 MAP, 975 REMOVE, 1919 SELECT, 43 SUBSTITUTE, 11 ADD - 3713
- **sma-dis.rle** - 20 MAP, 92 REMOVE, 70 SELECT, 2 SUBSTITUTE (4.10.2011) - 184
- **smi-dep.rle** - 117 SUBSTITUTE, 207 SETPARENT, 0 SETCHILD - 324

Making a whole grammar

- rule ordering: some tips
- put safe rules before not so safe rules

Safe rules

```
SELECT:r949 (V Sg1) IF (*-1 MUN BARRIER Nom OR V-SG1 OR S-
BOUNDARY LINK NOT *-1 V-SG1 BARRIER NOT-ADV-PCLE)(NEGATE 0
PrfPrc LINK *-1 REALCOPULAS BARRIER S-BOUNDARY OR CLB)
(NEGATE 0 PrfPrc LINK *-1 V-SG1 BARRIER NOT-ADV-PCLE);
 ## Dego mun dás vuollelis mottiin ovdamearkkain
čájehan,...

SELECT:r950 (V Sg1) IF (*1 MUN BARRIER NOT-ADV-PCLE OR
CLB);
 ## Man guhká ferten mun gierdat din?
```

If you need to check the previous/following sentence: Span Left, Span Right

- In order to continue a context search across window boundaries, use Span Left (<) and Span Right (>) as a pre- or postfix for the position block,
- e.g. <*-1 (left) or >*1 (right).
- Using 'W' instead of the arrows will allow a span to search in both directions.
- As a default, the span covers 2 windows left and 2 windows right of the focus window, but the number can be set arbitrarily with the –num-windows command line flag.
- (*-1 »> LINK -1W ("<:>") LINK -1 VQUOTE)
- will check if the preceding sentence ends in a colon, after a quoting verb, making the second sentences a quotation "object" of the first

Larger frames than sentences

- Make a rule for a recognizing leat as 1st plural if you in a sentence to the left of it find the personal pronoun "mii"
- other ideas? - pro drop, ellipser etc.

Sets

- LIST = @A @B ;
- SET = LIST LIST ;
- SET = N - LIST ;

Sets

```
SET TIME = MANNU OR VAHKKU OR BEAIVI OR AIGI ;
SET NOTIME = N - TIME ;
```

Sets

- sets as to-be-unified variables, prefixing $$ before the set name
- All occurrences of such a set in a given rule will be unified to mean the same set member, and the rule operation will only apply if the set does have a member that satisfies all occurrences of the set in both target and contexts at the same time
- Therefore, if a $$-set only occurs in contexts (and not in the target), the KEEPORDER option should be used.
- unification of semantic roles (agent, patient, theme and location):
- LIST ROLE = §AG §PAT §TH §LOC ;
- SELECT $$ROLE (-1 KC) (-2C $$ROLE) ;
- not possible in mapping rules: MAP $$ROLE TARGET (N) IF (-1 KC) (-2 N + $$ROLE) ;

Sets

```
MAP (@COMP-CS<) TARGET $$ADVLCASE
IF (-1 ("go" CS) LINK -1 $
$ADVLCASE)(1 COMMA LINK 1 VFIN) ;
    ## Juohkehaš guhte earrána
áhkástis mange eará sivas go
fuorrávuođas, dagaha su rihkkut
náittosdili.
```

Sets

- set union: OR or | , e.g. set1 OR set2 OR (tag3) OR (N F S)
- concatenation (cartesian product): + , e.g. set1 + set2, yields all possible combinations of the 2 sets' elements.
- Thus, a concatenation of LIST set1 = V and LIST set2 = INF GER PCP covers all non-finite verb forms: (V INF) (V GER) (V PCP).
- negation (match set difference): - , e.g. set1 but not set2, means set1 as long as the reading in question does not contain elements from set2. Thus, rather than as just a removal of set2 elements form the set1 list (i.e. defining list difference, as used in Tapanainen's cg2), vislcg3 interprets the minus operation as a kind of NOT condition, so the presence of a set2 element in a reading will block and override the presence of a set1 reading. Thus, (N) - (P) means non-plural nouns. If needed for

Sets

- The + and - operators have precedence over OR.
- failfast: The ˆ symbol can be used in both set operations (e.g. A ˆ C OR B) and set definitions (LIST = a b ˆc)
- A set or list element prefixed by ˆ will block instantiation of the entire set if matched in a given reading, even if other elements of the set would otherwise make the set compatible with the cohort line in question.
- Note that all set operators, as well as the parenthesis convention for creating sets on-the-fly, can be used in targets and context conditions of rules.

Sets

- tag inversion: ! (exclamation mark) is used as a tag-prefix and means ”all but ..” or ”but not”, much like the ^{f}ail fast prefix.
- ! is used in tag strings and context parentheses
- ^{i}s used in set definitions or set operations
- (V !PAS), for instance, matches all verb forms that are not passive
- magic set, (*), to denote ”everything”.
- e.g. LINK -1 (*) LINK 1 ... to include the 0 position in an unbounded search (useful in complex vp's).
- negate a set, e.g. (*) - (N) for all tokens that are not nouns
- (*) - N is formally equivalent to (!N), but faster for the compiler to match

Sets

- Make a set of what can be between a noun and a demonstrative
- Make a set of what can be between a mainverb and its adverbial argument

BARRIER

- BARRIER is used right after an unbounded context (e.g. *-1 context)
- barrier context blocks the preceding context search
- e.g. (*1 VFIN BARRIER CLB) looks for a finite verb (VFIN) anywhere to the right (*1), but only if there is no interfering clause boundary (CLB) in between
- BARRIER keyword can be used in careful mode, too (CBARRIER), where only unambiguous readings will block the search
- For NEGATEd contexts, CBARRIER is recommended

BARRIER - @CNP or @CVP?

- Sokkis sii seailluhit diŋggaid, muhto go lávus leat ollu olbmot, de sáhttet soggái maid nohkkat.

BARRIER - @CNP or @CVP?

- Sokkis sii seailluhit diŋggaid, muhto go lávus leat ollu olbmot, de sáhttet soggái maid nohkkat.
- REMOVE (@CNP) (*-1 VFIN BARRIER S-BOUNDARY)(0 CC)(1 CS) ;
- Sokkis sii seailluhit diŋggaid, muhto go lávus leat ollu olbmot, de sáhttet soggái maid nohkkat.
- E-boasta lea hui álkes ja jođánis vuohki sáddet reivviid birra máilmmi.

BARRIER - @CNP or @CVP?

- Sokkis sii seailluhit diŋggaid, muhto go lávus leat ollu olbmot, de sáhttet soggái maid nohkkat.
- REMOVE (@CNP) (*-1 VFIN BARRIER S-BOUNDARY)(0 CC)(1 CS) ;
- Sokkis sii seailluhit diŋggaid, muhto go lávus leat ollu olbmot, de sáhttet soggái maid nohkkat.
- E-boasta lea hui álkes ja jođánis vuohki sáddet reivviid birra máilmmi.
- REMOVE (@CVP) IF (-1C A)(1C A LINK *1 N BARRIER NPNH);

Make a set of rules for V Sg3

- Geas ležžet bealjit, son gullos!
- ...oažžu son atnit dušše dan čázis mii sutnje lea mearriduvvon.
- Sii manne vissui.

Make a set of rules for V Sg3

- Geas ležžet bealjit, son gullos!
- ...oažžu son atnit duššе dan čázis mii sutnje lea mearriduvvon.
- Sii manne vissui.
- manne mannet+V+IV+Ind+Prs+Sg3
- Biret-guovttos Juffáin leaba čeahpit duddjot.

Make a set of rules for V Sg3

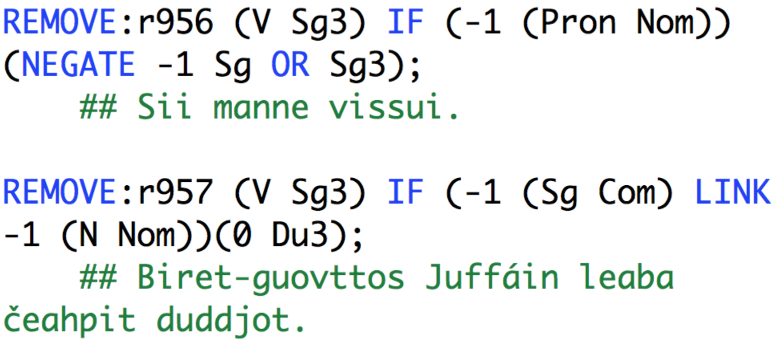
- Geas ležžet bealjit, son gullos!
- ...oažžu son atnit duššе dan čázis mii sutnje lea mearriduvvon.
- Sii manne vissui.
- manne mannet+V+IV+Ind+Prs+Sg3
- Biret-guovttos Juffáin leaba čeahpit duddjot.
- leat+V+IV+Ind+Prs+Sg3+Foc/ba

A set of rules for V Sg3

```
SELECT:r954 (V Sg3) IF (*-1 SON BARRIER Nom
OR V-SG3 OR CS OR PUNCT LINK NOT *-1 V-SG3
BARRIER NOT-ADV-PCLE)(NEGATE 0 Adv LINK *1
V-SG3);
    ## Geas ležžet bealjit, son gullos!

SELECT:r955 (V Sg3) IF (*1 SON BARRIER NOT-
ADV-PCLE OR go OR CLB)(NEGATE 0 CS)(NEGATE 0
Adv LINK *1 (V Sg3) BARRIER SV-BOUNDARY);
    ## ...oažžu son atnit dušše dan čázis
mii sutnje lea mearriduvvon.
```

A set of rules for V Sg3



```
REMOVE:r956 (V Sg3) IF (-1 (Pron Nom))
(NEGATE -1 Sg OR Sg3);
    ## Sii manne vissui.

REMOVE:r957 (V Sg3) IF (-1 (Sg Com) LINK
-1 (N Nom))(0 Du3);
    ## Biret-guovttos Juffáin leaba
čeahpit duddjot.
```

BARRIER

- SET NPNH = WORD - PRE-NP-HEAD OR ABBR ;
- SET NPNHA = WORD - PRE-NP-HEAD - Adv ;
- SET PRE-NP-HEAD = (Prop Attr) OR (Prop @>N) OR (A Attr) OR (ABBR Attr) OR ("buorre") OR (Pron Pers Gen) OR (N Gen) OR Num OR Cmpnd OR CC OR (Pron Dem) OR (Pron Refl Gen) OR (Pron Indef) OR (PrfPrc @>N) OR PrsPrc OR (A Ord) ;

BARRIER

- SET S-BOUNDARY = (Pron Interr) OR (Pron Rel) OR ("muhto") OR ("de" CC) OR MO OR ("") OR (":") OR ("-") OR ("–") OR CS ;
- SET S-BOUNDARY1 = (Pron Interr) OR (Pron Rel) OR ("muhto") OR ("de" CC) OR MO OR ("") OR (":") OR ("-") OR ("–") ;
- SET S-BOUNDARY2 = (Pron Interr) OR (Pron Rel) OR ("muhto") OR ("de" CC) OR MO OR ("") OR (":") OR ("-") OR ("–") OR (@CVP) ;

Regular expressions

- sometimes it is useful when a set contains many productive compounds
- the regular expression has an r after it
- ".*" stands for any amount of any character
- ".*ize"r to match certain transitive verbs in English
- to mark case insensitivity the letter i is used
- it is possible to use both "i" and "r"

Use of regular expressions in the North Sámi analyzer

```
LIST VAHKKU-TIME = ("[0-9]*#-jahki"r) "[0-9]*-[0-9]*-#jahki"
"[0-9]*-[0-9]*-#lohku" ... ;
LIST AIGI = "áigi" (".*#áigi"r) ;
LIST VOLUME = "cl" "lihtar" (".*#lihtter"r) "dl" "lihtter" "ml" ;
```

Use of regular expressions in the South Sámi analyzer

```
LIST CONCRETE-ROUTE = "baalka#raejkien" "byjje#raejkien"
dielhtie#raejkieh" "geajnoe#raejkien" "njaelmie#raejkien"
"okse#raejkien" "bïegke#raejkien" "johke#raejkien"
"rosse#raejkien" "sjaedtie#raejkien" "vaeljie#raejkien"
"voemesje#raejkien" "valte#raejkiem" "johke#raejkiem"
"laath#raejkiem" "guhkies" ;
LIST CONCRETE-ROUTE = (".*#raejkien"r) (".*#raejkiem"r);
```

Regular expressions practise

- How do you express that any compound of "skuvla" should be considered

Regular expressions practise

- How do you express that any compound of "skuvla" should be considered
- (".*skuvla"r)

Regular expressions practise

- How do you express that any compound of "skuvla" should be considered
- (".*skuvla"r)
- How do you express that both "Ipmil" and "ipmil" and "IPMIL" should be considered

Regular expressions practise

- How do you express that any compound of "skuvla" should be considered
- (".*skuvla"r)
- How do you express that both "Ipmil" and "ipmil" and "IPMIL" should be considered
- ("ipmil"i)

Testing existing rules for South Sámi

- MAP (@OBJ>) TARGET Acc (NOT 0 TIME OR ROUTE)(*1 MAINV + TV BARRIER S-BOUNDARY OR COMMA) ;
- MAP (@<OBJ) TARGET Acc (*-1 MAINV + TV BARRIER S-BOUNDARY OR COMMA)(NOT 0 TIME OR ROUTE) ;
- MAP (@OBJ>) TARGET Acc (NOT 1 EOS) ; #"guhkies" A Sg Acc @OBJ> MAP:952
- MAP (@<OBJ) TARGET Acc (NOT -1 BOS) ; #"guhkies" A Sg Acc @<OBJ MAP:954
- MAP (@OBJ>) TARGET (Pl Nom) (NOT 0 TIME OR ROUTE)(*1 MAINV + TV BARRIER S-BOUNDARY OR COMMA) ;
- MAP (@<OBJ) TARGET (Pl Nom) (*-1 MAINV + TV BARRIER S-BOUNDARY OR COMMA)(NOT 0 TIME OR ROUTE) ;

Specifying the context of rules - what can be inbetween - long-distance relationships

- linking a main clause to a subclause, the following things can disturb the flow
  - relative clauses, they can also start with an Adv (gosa, gos)
  - embeded subclauses

Fixing existing rules - step by step

- in the following sentence: Beana gal jorgala, go lea dakkár mii máhttá ja gille.
- the X does not get recognized
- why?
- many times the dependency rules do not get applied (correctly) because something in the syntax is wrong
- echo "Beana gal jorgala, go lea dakkár mii máhttá ja gille." | preprocess – /gtsvn/gt/sme/abbr=bin/abbr.txt | lookup -flags mbTT -utf8 /gtsvn/gt/sme/bin/sme.fst | lookup2cg | vislcg3 -g /gtsvn/gt/sme/src/sme-dis.rle -t | vislcg3 -g /gtsvn/gt/smi/src/smi-dep.rle

Thursday

Making a grammar checker

Danish Constraint Grammar based spellchecker OrdRet

- special focus on dyslexics
- multi-stage approach: data-driven error lists, phonetic similarity measures, traditional letter matching at the word and chunk level, and CG rules at the contextual level
- ordinary CG parser (DanGram) to choose between alternative correction suggestions + error types are CG-mapped on existing, but contextually wrong words
- OrdRet finds 68% of errors and achieves ranking-weighted F-Scores of around 49 for this genre
- word lists are not enough

Danish Constraint Grammar based spellchecker OrdRet

- linguistic core:
  - (a) word based spell checking and similarity matching,
  - (b) morphological analysis of words, compounding and correction suggestions,
  - (c) syntax based disambiguation of all possible readings, and
  - (d) context-based mapping of error types and correction suggestions
- DanGram removes all readings but one
- OrdRet re-appends all other suggestions as number 2.3... etc. according to their original weights and user preferences as to list length
- apart from the DanGram tagger-parser, OrdRet also uses a dedicated errordriven Constraint Grammar (ca. 800 rules) to resolve correction ambiguity,
- error-CG adds information: suggestions @-tags (@inf, @vfin, @neu, @pl)

Eckhard's examples

- green mistakes *
- red mistakes **
- Hun har en opfattelse af at kvinde* (@pl) er bedre til det merster** (R:meste). (no indefinite singular non-mass nouns without prenominals)
- Han kan ikke hører* (@inf) dig. (auxiliary verb context)
- Han ønsker ikke og** (@:at) forstyrre. (infinitive right, verb with infinitivevalency left)
- Min søster er syge plejerske* (@comp). (dictionary lookup)

Eckhard’s examples

- Hun besøgte barndoms* (@comp-) veninden. (indefinite singular noun in the genitive, immediately preceding definite noun)
- Glasset var fuld* (@sc-neu). (subject agreement of subject predicative)
- Jeg er træt* (@headstop) jeg vil hjem ... (syntactic indicators for sentence separation)
- Det har vært** (R:været) en lang dag. (‘været’ V wins over ‘vært N’ after auxiliary)

Estonian Grammar Checker

- How to collect comma errors?
- online news portals, internet forums
- Plan: first versions of bachelor theses
- Labels @OK and @ERR attached to conjunction words and finite verbs

Kevin

- put false friends into the lexicon
- CG-rules, but difficult to get good precision
- the more mistakes in a sentence, the bigger chance that the disambiguation is wrong
- open disambiguation up for missing commata etc. can cause right commata to be analyzed wrong
- errorcorpus: newspaper, essays (often too many mistakes)
- make very open rules first
- get the corpus through

Kevin

- pick the sentences that got the errortag
- structure them into good and bad ones
- make a testing set for this mistake
- ADD (&errortag) prep-that-takes-gen IF (p NOM);
- drive corpus through
- grep for &errortag
- pick 100 random sentences and order into wrong and good ones

Kevin

- useful: have a method to go back to the original sentence from the CG-format
- leif | morf | dis | synt | gramkontroll | reformat
- get out a long sentence with leif&errortag
- MAPPING-PREFIX = & ;
- ADD (&errortag &dummy) possibly-wrong-sentence
- REMOVE (&errortag) IF exception
- &dummy-tagg since the last reading cannot be removed
- how many errortags? - around 100

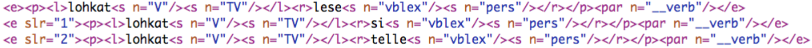
Friday

Lexical selection in Apertium

- apertium-sme-nob.sme-nob.lex
- first time used for apertium-sme-smj.sme-smj.lex
- Francis and me found words that cannot be directly translated, but the translation depends on the context
- Constraint Grammar is perfect to define context rules
- apertium-sme-nob.sme-nob.lex - 63 rules

Bilingual dictionary

bilingual dictionary specifies the possible equivalents of a word, numbered



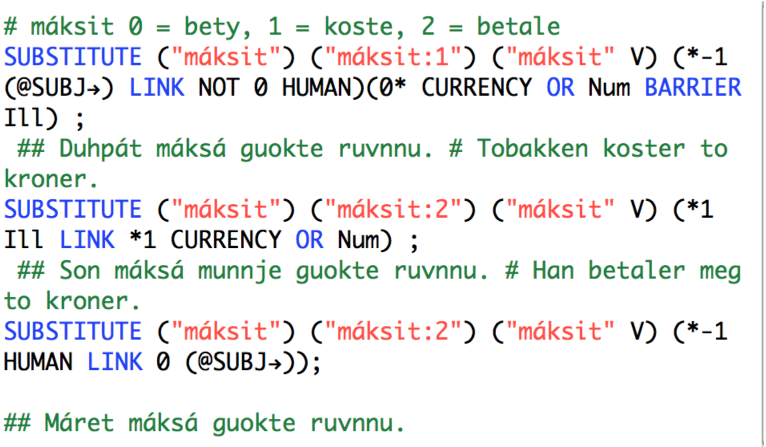
```
<e><p><l>lohkat<s n="V"/><s n="TV"/></l><r>lese<s n="vblex"/><s n="pers"/></r></p><par n="__verb"/></e>
<e slr="1"><p><l>lohkat<s n="V"/><s n="TV"/></l><r>si<s n="vblex"/><s n="pers"/></r></p><par n="__verb"/></e>
<e slr="2"><p><l>lohkat<s n="V"/><s n="TV"/></l><r>telle<s n="vblex"/><s n="pers"/></r></p><par n="__verb"/></e>
```

SUBSTITUTE Rule

```
# lohkat 0 = lese, 1 = si, 2 = telle
SUBSTITUTE ("lohkat") ("lohkat:1") ("lohkat" V)
((1 ("ahte") OR (Refl Acc) OR (Refl Loc) OR
PrfPrc) OR (*1 Nom OR Ess OR Ill BARRIER NPNH))  ;
## Ovddeš bargi Yle Sámi Radios, Ánne Risten
Juuso, lohká ahte Gárasavvonis livčče eará latnja
leamaš Yle Sámi radio doaimmahussii.
```

SUBSTITUTE Rule

bilingual dictionary specifies the possible equivalents of a word, numbered



```
# máksit 0 = bety, 1 = koste, 2 = betale
SUBSTITUTE ("máksit") ("máksit:1") ("máksit" V) (*-1
(@SUBJ→) LINK NOT 0 HUMAN)(0* CURRENCY OR Num BARRIER
Ill) ;
 ## Duhpát máksá guokte ruvnnu. # Tobakken koster to
kroner.
SUBSTITUTE ("máksit") ("máksit:2") ("máksit" V) (*1
Ill LINK *1 CURRENCY OR Num) ;
 ## Son máksá munnje guokte ruvnnu. # Han betaler meg
to kroner.
SUBSTITUTE ("máksit") ("máksit:2") ("máksit" V) (*-1
HUMAN LINK 0 (@SUBJ→));

## Máret máksá guokte ruvnnu.
```

Lexical selection in Apertium - Mistakes in the rules

- lohkat
- Son lohká máddin Sámis lea sámit garrasabbot deddon dahje vealahuvvon go davvin.
- - Sii áigot šiltet buot stáhta-, filkkagieldda- ja gielddageainnuid golmma gillii, doppe gos lea lunddolaš cegget dákkár šilttaid, namalassii guovlluide, gos orrot dáid gielaid geavaheaddjit, lohka Gabrielsen.
- orrut
- – Dieđusge vásihit sámi mánát nugo Anáris beaivválaččat ahte sidjiide lea hástalus beassat geavahit sámegielmáhtu iešguđet doaimmaid oktavuođas, muhto sii orrot goit čeahpit hutkat vugiid movt ávkkástallat sin sámegielmáhtuin, čilge son.
- http://paste.pocoo.org/raw/491882/

Dependency rules

- main difference is that we distinguish between main and subclauses - new tags
- SUBSTITUTE
- SETCHILD
- SETPARENT
- p (parent, mother)
- c (child, daughter)
- s (sibling)

Dependency format

- pointer node
- #2 -> 1
- #2 means "the second word in the sentence"
- ->1 means "and my mother in the sentence is word number N"
- for "to me", me would be marked with #2 -> 1

Dependency rule format

Rule

```
SETPARENT @HNOUN (NONE p (*)) TO (@0 (*));
```

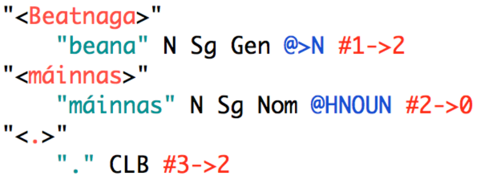
Dependency rule format

Rule

SETPARENT @HNOUN (NONE p (*)) TO (@0 (*));

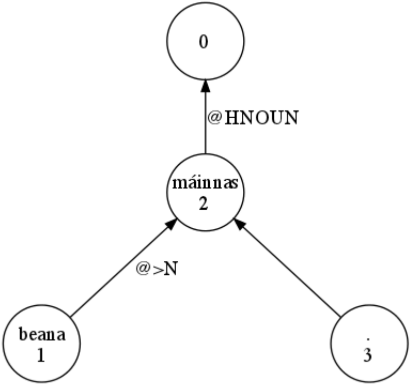
- set the mother of an item carrying the syntactic label @HNOUN to root @0
- for any set (the magic set *)
- all @HNOUN (headnouns) that do not have a parent of any kind
- NONE excludes all relations, p stands for parent, and the magic set * for all possible tags

Example sentence without a finite verb @HNOUN



```
"<Beatnaga>"
	"beana" N Sg Gen @>N #1->2
"<máinnas>"
	"máinnas" N Sg Nom @HNOUN #2->0
"<.>"
	"." CLB #3->2
```

Example sentence without a finite verb @HNOUN



Dependency rule format: options

- * (Deep scan) allows a child- or parent-test to continue searching along a straight line of descendants and ancestors, until the test condition is matched
- ’*p VFIN’, will find the finite verb in the parent verb chain, even if the subject or object itself is linked to a nonfinite main verb
- ALL or C requires a child- or sibling-relation to match all children or all siblings
- different from the ordinary C (= safe) option which applies to readings
- ’cC ADJ’ or ’ALL c ADJ’ means ’only adjectives as children’ – e.g. no articles orpp’s
- ’c (*) LINK 0C ADJ’ means ’any one daughter with an unambiguous adjectivereading

Dependency rule format: options

- NONE or NOT has the opposite effect of ALL - it means, that no child, or no sibling, may
- 'NONE c @>N' means that there is no premodifier child, i.e. that all children are not premodifiers
- (just one) daughter that does not match, the format is 'c (*) LINK NOT 0 @>N'
- S (Self) can be combined with c, p or s to look at the current target as well
- 'c @SUBJ LINK cS HUM' looks for a human subject np – where either the head noun (@SUBJ) itself is human, or where it has a modifier that is tagged as human

Dependency rule format

Link

SETPARENT:COMMA COMMA TO (*-1 @SCLV LINK *-1 @CVP) ;

This rule links the comma to a previous subjunction if a previous subclause-verb can be found it sets the head to @CVP, not to @SCLV other than in mapping rules the testing condition here is @SCLV

A couple of new tags for dependencies

- WHY NEW TAGS?
- WHAT ARE THE DIFFERENCES?
- subclauses

A couple of new tags for dependencies

- WHY NEW TAGS?
- WHAT ARE THE DIFFERENCES?
- subclauses
- are dependents of main clauses
- relative clauses

A couple of new tags for dependencies

- WHY NEW TAGS?
- WHAT ARE THE DIFFERENCES?
- subclauses
- are dependents of main clauses
- relative clauses
- are dependents of their antecedents (noun phrases)

How does a sublcause tree look like?

- the finite verb of the subclause goes to the main verb of the main clause

How does a sublcause tree look like?

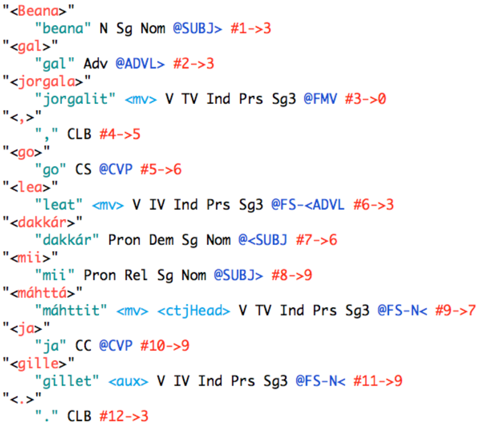
- the finite verb of the subclause goes to the main verb of the main clause
- the finite verb of the subclause can either be in object or adverbial function to the main clause

How does a sublcause tree look like?

- the finite verb of the subclause goes to the main verb of the main clause
- the finite verb of the subclause can either be in object or adverbial function to the main clause
- the subjunction goes to the finite verb of the sublause

How does a relative clause tree look like?

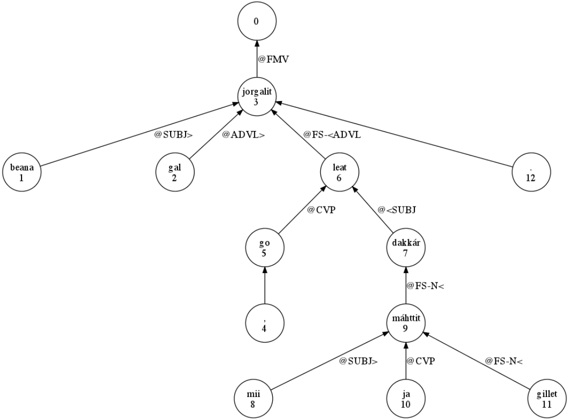
Beana gal jorgala, go lea dakkár mii máhttá ja gille.



```
"<Beana>"
	"beana" N Sg Nom @SUBJ> #1->3
"<gal>"
	"gal" Adv @ADVL> #2->3
"<jorgala>"
	"jorgalit" <mv> V TV Ind Prs Sg3 @FMV #3->0
"<,>"
	"," CLB #4->5
"<go>"
	"go" CS @CVP #5->6
"<lea>"
	"leat" <mv> V IV Ind Prs Sg3 @FS-<ADVL #6->3
"<dakkár>"
	"dakkár" Pron Dem Sg Nom @<SUBJ #7->6
"<mii>"
	"mii" Pron Rel Sg Nom @SUBJ> #8->9
"<máhttá>"
	"máhttit" <mv> <ctjHead> V TV Ind Prs Sg3 @FS-N< #9->7
"<ja>"
	"ja" CC @CVP #10->9
"<gille>"
	"gillet" <aux> V IV Ind Prs Sg3 @FS-N< #11->9
"<.>"
	"." CLB #12->3
```

How does a relative clause tree look like?

Beana gal jorgala, go lea dakkár mii máhttá ja gille.



How does a relative clause tree look like?

- the finite verb of the subclause goes to the antecedent of the relative pronoun
- the relative pronoun goes to the finite verb of the subclause

A couple of new tags for dependencies

There are three new types

- @FS- prefix stands for finites subclause
- @FS-N prefix stands for finite subclause modifying a noun phrase = relative clauses
- @ICL- prefix stands for infinite sublcauses

A couple of new tags for dependencies

- @FS-ADVL>
- @FS-<ADVL
- @FS-IAUX
- @FS-IMV
- @FS-OBJ
- @FS-OBJ>
- @FS-P<IMV
- @FS-SUBJ
- @FS-VFIN<

A couple of new tags for dependencies

- @FS-N<
- @FS-N<IAUX
- @FS-N<IMV

A couple of new tags for dependencies

- @ICL-OBJ
- @ICL-P<
- @ICL-SUBJ

Making dependency rules

Make dependency rules for relative sentences

- Muhto go dein ii leat rievttes isit, de šaddet dál biinnu vuolde leahkit jápmima rádjái — mii lea surges áššši, gii dan jurddaša ja ipmirda.
- Dat guđet suhttet ja álget viegahallat ja velá gilljot beatnagiigui -ja de bohccot ruhttet ja seahkanit ja de šaddá ođđa bargu, ja dan sivas dávjá suhtadit.
- Dat leat dat rokkit mat leat eatnama siste juohke sajis, gos dološ sápmelaččat leat orron.
- Mon lean okta sápmelaš, guhte lean bargan visot sámi bargguid ja mon dovddan visot sámi dili.

Making dependency rules

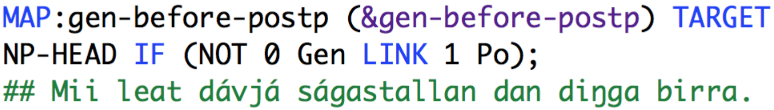
Make dependency rules for relative sentences

- Ja go dál ollejedje dan báikái, gosa lávejit vuosttaš beaivvi johtit, de sii luoitale.
- Ja de ferteje geafit čiehkat maid borre, ja go álggos ohppe veháš, de lassánii veháš geardde.
- Sii sárdnidedje ja dubmejedje deid visot, mat eai álgán čuovvut sin.
- Muhto go lassánedje olbmot, de bohte čáhppesbivttasolbmot fas dohko, gosa ledje sámit vuohččan ballán, ja dahke orohagaid jur dasa gos sámit ledje orrume, dan dihte go sii oidne, ahte das leai čáppa gieddi, maid ledje bohccot dutken, gožžan ja baikán — gos ledje sámit orron mánga olmmošbuolvva.
- Muhto go dakkár juoigit leat, guđet garrudit ja bániid gasket ja uhkidit goddit bohccuid ja velá isidanai, ja de dat leat ahkidat gullat.
- Ja dat uhkidedje juo álgit soahtái, muhto eai olbmot jáhkkán. Muhto sámit gal balle, geat oidne ahte dat risseje deid, geat eai álgán sin čuovvut.

Planning the future

- Where to get texts?
- Name of the grammar checker?
- interface
- what kind of rules

Making the first gramchk rule



```
MAP:gen-before-postp (&gen-before-postp) TARGET
NP-HEAD IF (NOT 0 Gen LINK 1 Po);
## Mii leat dávjá ságastallan dan diŋga birra.
```

GIITU!

Bibliography