Interactive pedagogical programs based on constraint grammar

Lene Antonsen, Saara Huhmarniemi, Trond Trosterua





Abstract

We here present a set of interactive parser-based CALL programs for North Sámi. The programs are based on a finite state morphological analyser and a constraint grammar parser which is used for syntactic analysis and navigating in the dialogues. The analysers provide effective and reliable handling of a wide variety of user input. In addition, relaxation of the grammatical analysis of the user input enables locating grammatical errors and reacting to the errors with appropriate feedback messages



Background and pedagogical motivation

The pedagogical programs in OAHPA! are based upon three pre-existing language technology resources developed at the University of Tromsø: a morphological analyser/ generator, a CG parser for North Sámi and a number word generator compiled with the

The main goal of the development of OAHPA! was to develop a language tutoring system going beyond simple multiple-choice questions or string matching algorithms, with free-form dialogues and sophisticated error analysis. Immediate error feedback and advice about morphology and grammar were seen as important requirements for the program.

Pedagogical lexicon

The OAHPA! programs share a set of common resources; a pedagogical lexicon and a morphological generator that is used for generating the different word forms that appear in the programs. The dialectal variation is taken into account in the lexicon as well as in the morphology. The lexical entry for monni "egg" is given below:

```
eartrys
elemnomornie/lenno
    eth anti-targe "not" seggesters
eth anti-targe "fin" anuna of tra-
  e/translationsy
    view class="F000 Oxcoon"/v
   e/semantics-
  estem class="tisyllub.c" digethorg="so
   gradation="yes" saggi="\" rise="2"/s-
sticlect class="401 KJ"/s
   -book name-"som" />-
```

The morphological properties of words are used when making a detailed feedback on morphological errors. If the user does not inflect the lemma correctly, she can ask for hints about the inflection, and try once more, instead of getting the correct answer straight away.

Training numbers



The Numra game offers training on number expressions, either from number symbols to linguistic representation or vice versa. As our only program, Numra offers training for four different Sámi languages, Kildin, Inari, North, Lule and South Sámi. We hope to extend our coverage to more languages for the other

Morphological feedback

The feedback messages are determined by the combination of morphological features in the lexicon and the inflection task at hand The morphological specification below gives a rule stating that there is a vowel change in illative singular for bisyllabic nouns that end with the vowel i. The corresponding feedback message instructs the user to remember the vowel change

vation of con-Thiese Labrief I speci-1175 The system-internal representation of monni states it is a bisvllabic irecssage lianti_intrinsel change i v d. stem, which triggers i > -c/heepage: á change in illative. No. of Contract mount and experiently you and use have and gets feedback from the machine. A correct answer gets green colour.

Handling dialectical variation

When generating sentences or providing the correct answers for the user we allow only normative forms in the chosen dialect. On the other hand, the live analyser used for the analysis of the user input accepts all correct dialect variants of the same grammatical word. We compile one normative but variation-tolerant transducer for analysing the input, and one strict one for each dialect for sentence generation.

In the source code, forms are marked as missing in certain dialects (the default being that all forms occur in all dialects). Below is an example of dialectal variation in the comparative inflection. The resulting transducers give stuorát for the KJ dialect and stuorit for the GG one, of the adjective stuoris "big".

•A+Comp::35-84b BUStern ; / M07-K3 •A+Comp::356-36b BUStern ; / M07-G0

Sentence generation in the OA game Vasta

One of the main goals of the programs in OAHPA! is to practice language in natural settings with variation in the tasks. In order to provide variation in programs that involve sentential context we implemented a sentence generator. The sentence generator is used in the morphology in sentential context program (Morfa-C), and for generating questions to the QA drill



Evaluation

The overall evaluation shows that the students answer correctly slightly half of the time. By far the most popular program is the basic morphological drill (but the interactive programs have been logged for a couple of days only).

Program	Element.	"A" to mag	T	-
March 5	60000	- T-	1000	10.0
Tariban.	100000	42742	90000	107.0
The second	900000	2.012	4.80%	
March Co.	10000	160	3000	480.0
Sales	100	200	644.0	10.0
Period	3.65	1800	10.0	10.0
Test	7797	1.5 (200	7.00	375,484

The 322 logged Sahka errors are distributed along the following lines

Allertic byger		And to Type	
and the second of	100	manager in Young	
sadic proces	20	are constant for Money	
manage 2000 ages	-86	maken kenan	
and the final by the		and the second second second	
Service Audition	24	writing month	7

Conclusion

By using a sloppy version of the syntactical analyser for North Sámi, combined with a set of error-detection rules, we have been able to build a flexible CALL resource. The programs are modular, and the modules may be improved by adding more materials -- words, tasks, dialogues, levels, words from textbooks. The CG parser framework was originally chosen as parser framework for Sámi due to its extraordinary results for free-text parsing. The present project has shown that CG is well fit for making pedagogical dialogue systems as well.

CG-parser in live analysis in the interactive programs Vasta and Sahka

rather let the student formulate her own answer. To a certain question one may give many kinds of acceptable answers. In Sámi one may change word order, and also add many kinds of particles.

We use a ruleset file which disambiguates the student's input only to a certain extent, because there will probably be grammatical and orthographic errors. The last part of the file consists of rules for giving feedback to the student's grammatical errors, and rules for navigating to the correct next question of in the dialogue, due to the student's answer

The system question and student answer are analysed together, delimited by the boundary marker ^qst. They first get a morphological analysis (left), and are then disambiguated, and, if possible. assigned an error tag or a navigation tag.

logical

analysis

(sme-norm.fst)

machine

Schematical view of the process

Analysis

lookup2cg

disambiguating.

interpretation

ped-sme.cg3

processing error detection,



Grammar feedback

The system may give feedback, as shown here Here comes an explanation to the example &grm-missing-Ill



```
WY Characteristic Court flag (* 0.20), 180 f. Phanton back Ac. N.
                             auth twocyfeachdighf nancyffloraeldaghawaris.
                                 stectabler boards don feet?oftesta-
                               rails targets young" I not "Dougrapout a school going "As well targets of his "I not "Bound you no creat gar wheel _chille" or will targets white "I not "Dougrapout adult" "As well targets default" I not "Dougrapout adult" "As well targets default" I the "Dougrapout adult "As
```

County over Indian service (years) from full by the (years) from full by the (years) from Indian by the SECURIES

Taking Silvery 101

Security

The Control of the Control TV NAME OF STREET "Martin" "Associati () il gostajniojen.jiV Hot R Prop Ptv. by 111 hazar Prop Ptv. by 111 West in Prop. Ptc. Sci. III. Trible Pro Por Posto.

Hand R Is to the Aug.

Hand R Is to the Aug.

The CG-rules disambiguate and add grammar-error-tag (&grm-missing-Ill) and navigation-tag (&dia-hivsset):

```
Tailetel Pror Orders Sq Act Sgre-risking-311
    "Service" of Services of Services Servi
                               "man" Proc Hors (A.C. Name)
  weight
widge," a ny and see tot.
"Man" Proc Fork Sgs for
"476-"
                             "IN" IT MAY SO NOT
"viget" or garablejet videle-hveret
                               "man" Proc Back Sup Name
  "vidja." v iv ird see un
"vidja." v iv ird see un
                               The make spice
                             "YOURSEL" HING LOC
```

Navigation



Navigating inside the dialogue is implemented in CG-rules. The user input is tagged during analysis with information on whether the answer is interpreted as affirmative or negative. In addition, a special tag indicates whether the sentence contains some information that should be stored for the following questions or utterances. The program is thus able to store simple information such as the example the type of her car and use this information in tailored utterances.

In the example to the left the question is "In which room do we put the TV?" One of the alternatives assigned because of the lemma hivsset ("WC"). The answer will be "That is not a good idea. Make a new try." The CG-rule:

SAF Carlo Historia DANCE (N. 15 (6 Openaloudjet.TV)) (C. (historia) ENGINE ENGRY)

There are alternative links in the dialogue, one of them is due to the &dia-hivsset tag:

```
with hyperflyandism's sensity on hidde 1975
shock that belongs, now hid a no Whylands
out, toront-hidden it there are unitable of
controller get it belon Desired without offering
and harped "def and." Limbs" generated july beautiful of an action while the switch these victors.
```

Every question has its own unique id, which is used in navigating between questions. In addition, the CG-rules may be tailored for specific questions, like in the rule above.

Age-tags are assigned with help of regex-rules to the answer to the question "How old are you?". The tags function as links for moving to the next dialogue branch tailored to