Lene Antonsen, Saara Huhmarniemi, Trond Trosterud







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 Xerox compiler xfst.

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 freeform 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
classics confedence
  4909 Classe MVA
   stranslattanes
     eth anti-large instillegge/ork
     eth anticlong-"fin" ananoe'tra
   e/translations/
  keenanthook
keen class="F000 G00050"/k
   e/perior block-
  stantclass="t.syllut.c" dispting="so
graphics="yes" soggi="t" rise="2"/s-
stiplect class="90" EUT/s
   shook name/1507/5
shook name/1607017/5
     whook names "stigative
   e/sources-
```

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

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

water class-this/dichtof spari-115-The system-internal representation of monni states it is a bisvllabic irecssage lianti_intrinsel change i v d. stem, which triggers i > -c/manages á change in illative. The user types the No. of Contract mount and experiments you and use they and gets feedback server could. You'd always by it the office 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 QA 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 (Vasta)



The 322 logged Sahka errors are distributed along the following lines:

Alartic Sygn		Jan 10 Types	
and the second of	100	manager for Young	-
saddle weeks	20	are assess the Mann	
and the second second	-86	manage branch	1
and the limited by		and the second second second	
weavy histories	24	and the second	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 Sami 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

feedback

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

answei

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

command the figure of arting of 1.2. With a common depth of the factor and if the tree, where a gas

MAY (Square) print [11] TABORT (Topology) IT SI (4 INT) LIB (42 OUL LIB (40T 42 DIN C) US-400 CB Eng EMACHE S-ELECTROS);

JACON "guitor" Fron Intern Sg Acc "guitor" Fron Rel Sg Acc "guitor" Fron Rel Sg Acc "guitor" Fron Intern Sg Sen "Actual official" "Thereby all telling of the "Second of Second Performance From The space of the Person of the Person of the Space of the Person of the Table 1 Section For FIR Table 1 Section 1 Sect "Medito" Oil, generalização di "Medita" Press Fin Ser III Paral Francisco Para Para Ser Pilode di Press Mar Ser IV. "Maray III. Vita Ins. For Evil Pilodes V Add To The Court Person Digit Alan Tenan Tenan Berman Berm A in an against This was et ? Sign Heat Party S This was et? N. Sign Lee "Plyment" 6 Sp Sen PoSpS "Plyment" 6 Sp Sen PoSpS "Townert" 6 Sp Sen Assgd CONTRACTOR -dieder "gehief" Peon Tattern Sg Acc Agreemissing-TIT "gehter" Pron Onbern Sg Gen Agreemissing-TIL "startist" "Lettric" H Sg CLL 'nun'i Pron Pers Dull You "bidiot" V TV End Fra Dul nun'i Fron Pers Sol Sen TTV N ACR Sq Acc sahka" QDL gasa bidjat TV nun'i Pron Pers Dull Non-"bidget" V TV End Pre Dul. "dat" From Pere SgS Acc "dat" From Den Sg Acc "dat" From Pere Sg3 Sen "dat" From Den Sg Ger

"hivset" M Sy Lot Ssio target

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).

7	10000	200	1	-
March 5	60000	- C. C. C. C.	10000	100.0
Tarkers.		425-02	100,000	100
The second	90000	20.0	3.000	200
March C.	100.00	100	2004	480.00
Service.	100	200	644	100.0
Promise .	3.65	100		1.00.07
Total	7000	1.6 (250)	200	100,400

Evaluating Sahka errors



Navigation

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 student's name, place where she lives and for example the type of her car and use this information in tailored utterances.

Navigating inside the dialogue is implemented in

```
Mar (Salla Barget) TREET MY 8500 ± 010

16 (M: Qui Bascher Grandwar (196 M:-1 (N:10))

100 ±1 ( Quita*))(NOT 6 NOTHERS) ;
with types/question" seem/grow hidjet DF1
stand-date talogist must hidje as TFV-hard-
oist terrast-falsesed" times promote talogist-
craticalizer gill 11 set on Description without a American
politic politic services and provided without a American
with harpets' delegate" hints" generaled pet benedit."
stant. Soil, gotte des bettes these stant.
```

```
Company that are partir that processing only

    DECTION DATE OF THE STATE OF THE PROPERTY OF THE STATE OF
```

WV Place Children DWMT Ray (*) G SCL 1880 6 (*) and constraints And All

```
act type-"sucscien" name-"HomoldJamelyou's
  strateMar teamte den TestPolieste
estit turget- pengil Tris-Dougougeutstusdestigvong//s
estit turget- childi Tris-New yould briedle jethoolightisiske
  volt turget= abit* first-"Dogonwork.odule"/s
volt turget="defoil" first-"Dogonwork.odule"/s
```