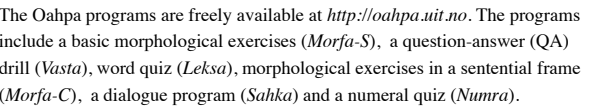


CAHPA!

<http://oahpa.uit.no>

We have made 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 CG-parser provides 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.



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 semantic class is used in the sentence generator for Vasta and Morfa-C. The lexical entry for *monni* "egg" is given to the right.

If the user does not inflect the lemma correctly in the morphological exercises, she can ask for hints about the inflection, and try once more, instead of getting the correct answer straight away.

```

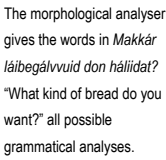
<div class="blog-title" style="text-align: center; border: 1px solid black; padding: 5px; margin-bottom: 10px;">
    <h1 style="margin: 0; font-size: 2em; color: #333;">My Blog Title</h1>
</div>
<div class="blog-content" style="border: 1px solid black; padding: 10px; margin-bottom: 10px;">
    <h2 style="margin: 0; font-size: 1.5em; color: #333;">Blog Post Title</h2>
</div>
<div class="blog-footer" style="border: 1px solid black; padding: 5px; text-align: center; margin-top: 10px;">
    <p style="margin: 0; font-size: 0.8em; color: #333;">Copyright © 2023. All rights reserved. <a href="#">Privacy Policy</a> <a href="#">Terms of Service</a></p>
</div>

```

A correct answer gets green colour as feedback.

The sentence generator in Morfa-C and Vasta is able to generate a virtually unlimited number of different tasks, and allows the student to use the programs over and over again.

Constraint grammar is a syntactic framework for choosing correct grammatical analysis of a given wordform, based upon the context it occurs within. Each rule removes or selects readings, or adds or removes a syntactic tag. Inappropriate analyses are removed, but the last analysis is never removed. CG thus always gives an analysis, and is therefore a very robust framework, well fit to handle potentially erroneous input.



The CG grammar then picks the correct analysis, and adds grammatical function and dependency structure.

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).

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

For Sahka we test *precision* (correctly identified errors/all
diagnosed errors) *recall* (correctly identified errors/all errors), and
accuracy (correct judgements/cases).

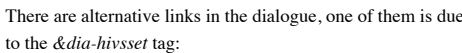
Precision = 0.8; Recall = 0.68; Accuracy = 0.82 (N=584)

By using 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.

Antonsen, Lene, Saara Huhmarniemi and Trond Trosterud 2009: Interactive pedagogical programs based on constraint grammar. Proceedings of the 17th Nordic Conference of Computational Linguistics. *Nealt Proceedings Series* 4. <http://dspace.utlib.ee/dspace/handle/10062/9206>

The CG-rules disambiguate the input, and the rule above adds a grammar-error-tag (&grm-missing-ill) to the input sentence (the analysis to the left). In the grammar feedback library, the tag in question looks up a message in the appropriate interface language (in this example, English), and the user is presented with the feedback *The answer should contain an illative*, as shown above.

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



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?”. Due to the tags the system choose the correct link for moving to the next dialogue branch tailored to the student's age.