Improving feedback on L2 misspellings – an FST approach

Lene Antonsen – University of Tromsø – lene.antonsen@uit.no

Workshop: NLP for computer assisted language learning, SLTC 2012

L2 MISSPELLINGS

In the learner's production there will be both **errors of performance**, which characteristically will be unsystematic, and errors of competence, which will be systematic.

The **errors of competence** can be divided into two groups:
1) Those which are morphologically irrelevant, but still systematic, like for North Saami writing *a* instead of *á* in the stem. The North Saami orthography is radically different from Norwegian and Swedish, and is a challenge for the language learners.

2) Those which are morphologically relevant, e.g. skipping the monophthongizing going from the nominative form *viessu* 'house' to the illative form *vissui* 'to the house' (which gives the erroneous form *viessui*), or choosing the wrong suffix. Also these are systematic errors, and they are possible to predict and give to the analyser.

FEEDBACK

Feedback to the L2 should support and facilitate learning, and the error should be seen as a chance of getting the language learner to not only to correct the word or phrase, but to understand the reason for the misconception.

If the misspelling is an error of performance, it will be sufficient to make the student aware of it. But if it is an error of competence, the student needs a correction, and if it is a metalinguistic comment, it is crucial that the feedback is given according to what the student thinks he has written, and at his own level of competence. This is the challenge with real word errors. The student will be confused when getting feedback on the syntax instead of the misspelling, e.g. feedback on using an infinite form instead of a finite form, when the student believes he has written a finite form.

SPELL CHECKER

Most spell checkers are generic and made with L1 users in mind. Testing demonstrates that the North Saami spell checker is not sufficient for L2 writers:

A relatively big part of their misspellings are **real word errors** and are not identified.

In a study monitoring 563 errors, the generating and ranking of candidates of the **non-word errors,** was not good enough for 32.3 % of the cases. The main reason is that the average edit distance for the L2 misspellings was as high as 1.54. A similar annotated corpus of L1-errors gave an average edit distance of 1.26. Another reason is that the phonetic rules ranking errors do not suit L2 writers, who often are not sure of how a word should be pronounced.

THREE WAYS of ENRICHING the FST with SYSTEMATIC MISSPELLINGS

Lexical transducer (lexc)

Suffixes are added and some vowel and consonant changes are done in the lexical transducer. One may make erroneous paths for adding wrong suffixes, e.g. the incorrect suffix *-i* instead of *-ii* for nominals with trisyllabic stem. The erroneous path is marked with an error tag `IllErr` in the upper level (here: to the right):

```
"<hivssegi>"  "hivsset" N Sg Ill IllErr
"<hivssegii>" "hivsset" N Sg Ill
```
'to the toilet.N'

Some suprasegmental processes are taken care of in the phonological transducer triggered by a dummy symbol in the lexical transducer. The erroneous path is made without this dummy, e.g. inflections with strong grade for the consonant centre when there should have been weak grade. The error tag in the upper level is `CGErr`:

```
"<áhkku>" "áhkku" N Sg Nom
"<áhkku>" "áhkku" N Sg Acc CGErr
"<áhku>"  "áhkku" N Sg Acc
```
'grandmother.N'

Concatenating transducers

There is also a special transducer for lowercase initial letter in place names, which is concatenated to the main transducer after the first compilation process. All forms have the tag `LowercaseErr` at the upper level, and this gives the following analysis of the misspelling *lundas* and the target form *Lundas*:

```
"<lundas>" "Lund" N Prop Plc Sg LocLowercaseErr
"<Lundas>" "Lund" N Sg Loc
```
'in Lund'

Phonological transducer (twol)

The phonological transducer changes letters under specific conditions, like when it changes the consonant centre, if it is followed by one of more vowels and the dummy WeG:

```
hkk -> hk, rj -> rjj, ... || _ Vow* WeG ;
```

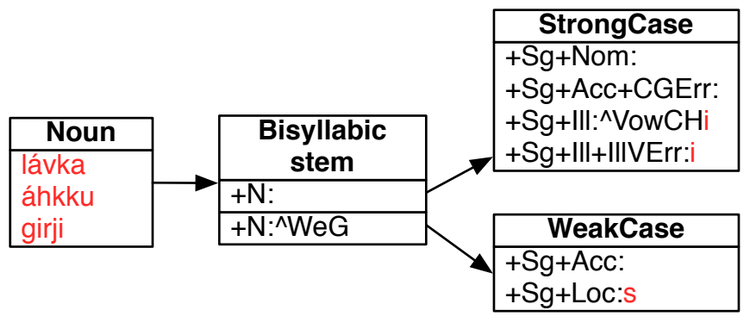
Some misspellings are generated by first adding a path with error tags to both upper and lower level in lexc, and then removing the error tag from the lower level under special conditions in twolc. The analyses with error tag in both levels are then removed from the output of the FST, by means of regex-rules.

The erroneous path can be a rule that changes letters generally, from a letter with a diacritic mark to a letter without, like changing *á* into *a*. The path with the error tag `AErr` remains in the upper level only if the change happens. This gives the following analysis of the misspelling *barru*.

```
"<barru>" "bárru" N Sg Nom AErr
"<bárru>" "bárru" N Sg Nom
```
'wave.N'

Other rules change letters under special conditions, like diphthong simplification, and the erroneous path with error tag `DiphErr` will remain only if the diphthong simplification does not happen. This gives this analysis of the misspelling *viessui*.

```
"<viessui>" "viessu" N Sg Ill DiphErr
"<vissui>" "viessu" N Sg Ill
```
'to the house.N'



The lexical transducer is adding both suffixes and dummies for the phonological transducer to the stem. The dummies here are WeG for consonant centre in weak grade, and VowCH for vowel change. The erroneous paths without the dummies are marked with error tags: `+CGErr` and `+IllVErr`. This automat gives analyses as these (left: correct, right: erroneous):

lávka+N+Sg+Ill	lávkii	lávká+N+Sg+Ill+IllVErr	lávkai
girji+N+Sg+Ill	girjái	girji+N+Sg+Nom+IllVErr	girjii
áhkku+N+Sg+Acc	áhku	áhkku+N+Sg+Acc+CGErr	áhkku
lávká+N+Sg+Acc	lávkka	lávká+N+Sg+Acc+CGErr	lávka

The following erroneous paths have been added to the FST, and the analyser thus knows both the target form and the grammatical word of the misspelled word:

error tag	err.form	targetform
Lowercase (place names)	londonis	Londonis 'London.SgLoc'
AErr (general rule)	manna	mánná 'child.SgNom'
AiErr (verbs)	boahtan	boahtán 'come.V.PrfPrc'
CGErr (nouns)	skuvlas	skuvllas 'school.SgLoc'
DiphErr (nouns)	viessui	vissui 'house.SgIll'
IllVErr (nouns)	skuvlai	skuvlii 'house.SgIll'
IllErr (nouns)	hivssegi	hivssegii 'toilet.SgIll'

EVALUATION

As testbench was used an existing ICALL-program, which accepts free-input, and has L2 learners as its target group (*http://oahpa.no/davvi/*). By using an error-FST as morphological analyser instead of a normative FST, it was, to some extent, possible to recognize the student's intended word, and, by means of Constraint Grammar-rules, trigger metalinguistic feedback as help for the student, like: "X misses diphthong simplification" with a short explanation of the morphological process.

Testcorpus: 2705 logged question-answer pairs were parsed with the normal and the error-FST, respectively, and then parsed with the CG-rules. With the error-FST the number of analyses increases with 12.1 %, from 2.26 to 2.54 pr wordform, before disambiguation.

Misspellings found in the testcorpus	Norm.FST.		Err.FST	
Target form not recognised	871	91.9%	563	56.0%
Target form recognized	77	8.1%	443	44.0%
Total	948	100%	1006	100%

By parsing the test corpus with the normal FST, the target form for only 8.1 % of the misspellings was recognised. They were recognised by means of special GC-rules for systematic real word errors. By parsing the test corpus with the error-FST, the target form for 44.0 % of the misspellings were recognised regardless of whether they were real word errors or non-word errors. The precision and recall for the system did not decline using the error-FST.

The analysis also recognised combinations of the erroneous forms, e.g. the word *fallejohkas* is recognised as a misspelling of the target form *Fállejogas* despite of an edit distance of 4:

```
"<fallejohkas>" "Fállejohka" N Prop Plc Sg Loc LowercaseErr CGErr AErr
"<Fállejogas>"  "Fállejohka" N Prop Sg Loc
```
'in Fállejohka'

All the extra paths make the error-FST almost ten times as big as the normal FST. The compilation time increases to 667 %. It is, however, possible to make the error-FST smaller by removing rare dynamic compounding and derivation paths, which are not likely to occur in the language of L2-students.

	Normal FST	Error FST
size	41.5 Mb	398.8 Mb
	100%	959%
states	497,632	4,739,590
arcs	1,062,995	10,297,121

CONCLUSION

Adding grammatical misspellings to the finite state transducer gives promising results:

- It makes the syntactic analyser able to recognise systematic misspellings, both real word errors and non-word errors, even if the edit distance is as big as 4.
- Even though the number of analyses per wordform increases, it does not ruin the disambiguation in a restricted ICALL program. In fact, by means of the erroneous forms some of the students' errors are reclassified from syntactic or semantic errors to misspellings, and the student gets a feedback according to what he thinks he has written.
- The error tags made it possible not only to recognise the target form for 44 % of the misspellings in the test corpus, but also to give tutorial feedback on the nature of the error to the student.
- The erroneous forms also make it possible to ignore misspellings in favour of giving feedback on syntax.
- The size of the error-FST expands exponentially, but it can be trimmed for L2 users.

ACKNOWLEDGEMENTS AND REFERENCES

Thank to my supervisor, professor Trond Trosterud, for discussions and valuable input.

Lene Antonsen, Saara Huhmarniemi and Trond Trosterud. 2009a. Constraint Grammar in Dialogue Systems. Proceedings of the 17th Nordic Conference of Computational Linguistics NEALT Proceeding Series. Volum 8. p. 13–21. Odense, Denmark

Lene Antonsen, Saara Huhmarniemi and Trond Trosterud. 2009b. Interactive pedagogical programs based on constraint grammar. Proceedings of the 17th Nordic Conference of Computational Linguistics NEALT Proceeding Series Volum 4. Odense, Denmark.

Kenneth V. Beesley and Lauri Karttunen. 2003. Finite State Morphology CSLI publications in Computational Linguistics. USA.

Eckhard Bick. 2006. A Constraint Grammar Based Spellchecker for Danish with a Special Focus on Dyslexics. (ed.) Suominen, Mickael et al. A Man of Measure – Festschrift in Honour of Fred Karlsson, p. 387-396. The Linguistic Association of Finland.

S. P. Corder 1967. The significance of learner's errors. International Review of Applied Linguistics 5, p 161– 169.

Carl James. 1998. Errors in language learning an use: exploring error analysis. Longman. USA.

Fred Karlsson, Atro Voutilainen, Juha Heikkilä and Arto Anttila. 1995. Constraint grammar: a language-independent system for parsing unrestricted text. Mouton de Gruyter.

Kimmo Koskenniemi. 1983. Two-Level Morphology: A General Computational Model for Word-Form Recognition and Production.. Publications of the Department of General Linguistics, University of Helsinki, No. 11.

Anne Rimrott and Trude Heift. 2008a. Evaluating automatic detection of misspellings in German. Language Learning & Technology 12(3). p 73–92.

Trond Trosterud and Heli Uibo. 2005. Consonant Gradation in Estonian and Sami: Two-Level Solution. (Eds) Antti Arppe et al. Inquiries into Words, Constraints and Contexts. p. 136–150.