Practicing syntax in spoken interaction: Automatic detection of syntactical errors in non-native utterances

Helmer Strik, Janneke van de Loo, Joost van Doremalen, Catia Cucchiarini

Department of Linguistics, Radboud University Nijmegen, The Netherlands

{h.strik, j.vandoremalen, c.cucchiarini}@let.ru.nl, JannekevandeLoo@student.ru.nl

Abstract

In the current paper we present a new method, called SynPOS: Syntactic analysis using POS-tags. SynPOS is applied to a corpus of spoken human-machine interactions. The results show that language learners of Dutch often make syntactical errors, that there are many different types of syntactical errors, and that their frequencies vary a lot. This information can be used next to select errors and develop exercises for CALL systems.

Index Terms: syntactic analysis, syntactical errors, part-of-speech, POS tags, ASR-based CALL

1. Introduction

Within the framework of Computer Assisted Language Learning (CALL) numerous systems have been developed for practicing grammar (morphology and syntax) in a foreign or second language. In the majority of these systems the learner's output is provided in the written modality, by means of a keyboard and/or a mouse (clicking, drag & drop, etc.). Although this way of practicing may be successful for learning the grammar of the target language, it is questionable whether the knowledge thus acquired really contributes to speaking the target language more correctly. Two important questions may be raised in this respect. First, according to some researchers this type of explicit knowledge about grammar is essentially different from the implicit knowledge of a language that is acquired from usage, rather than from rules and drills, and that is required for communicative competence (for a brief overview, see [4]). Second, it is not clear whether knowledge acquired in one modality (written) generalizes to other modalities (spoken). Research so far indicates that this is not the case [3].

For these reasons, it is very interesting to develop CALL systems that can handle non-native speech and that make it possible to practice grammar and to receive feedback while speaking in the target language. However, as far as we know, grammar has not yet been systematically addressed in ASR-based CALL systems that analyze L2 learners' speech production. Exceptions are Lee and Seneff [5], in which an approach for automatic grammar correction is presented, and the DISCO (Development and Integration of Speech technology into Courseware for language learning) project [9] which is aimed at realizing a CALL system that makes use of automatic speech recognition (ASR) for assessing speech of learners of Dutch as a second language and for providing corrective feedback on pronunciation and grammar. In the FASOP (Feedback on Syntax in Oral Proficiency) project [11] we will use the latter system to study the effect of providing different types of feedback on the acquisition of syntax in oral proficiency.

Although an ASR-based CALL system for practicing grammar may seem particularly appealing, developing a good system is far from trivial, mainly because automatic speech recognition of non-native speech is still problematic and thus only limited tasks can be used. For instance, as prompts one often uses written utterances that have to be read aloud or spoken utterances that have to be repeated. While such exercises can be useful for practicing pronunciation, they are not appropriate for practicing grammar.

Given the limitations of speech technology, the question then is how grammar can be practiced in such CALL systems. In the current paper the focus is on syntax. In order to practice syntax, we need to know what should be practiced, what the exercises should look like, and then we need to develop the technology to automatically handle these exercises. The final goal of the current line of research is a method that makes it possible to develop an ASR-based CALL system for practicing grammar.

In the Dutch-CAPT project [2, 10] we faced similar problems regarding pronunciation in Dutch as L2 and we adopted the following procedure: make an inventory of the errors, list criteria for selecting the errors, use them to select errors, and finally develop a system, i.e. the exercises to practice these aspects and the technology to handle these exercises (detect the errors and give feedback about them). In the current paper we explore the possibilities of using a similar procedure for syntax.

In the case of pronunciation, one can go through an utterance from the beginning to the end and determine for every sound whether it is pronounced correctly or not. In the case of syntax, the issues are more complex. For instance, it is not possible to simply go through an utterance from the beginning to the end and determine for every word whether it is correct or not. In fact, it is not straightforward what kind of method should be used to analyze non-native speech data, to make the inventory of errors, to select errors, and to generate the system (exercises and technology). We present a new method for automatically generating an inventory of (syntactical) errors made by non-native speakers by analyzing utterances from a corpus of non-native speech. The method makes use of part-of-speech (POS) tags to label the words in each utterance, and an algorithm that matches words in two utterances: the (correct) target utterance and the (possibly erroneous) realization of the utterance. In section 2 we describe this method together with the non-native speech material we used. The results are presented in section 3 and discussed in section 4.

2. Material and method

2.1. Material

The non-native speech material for the present experiments was taken from the JASMIN speech corpus [1]. Recordings were made for speakers with many different mother tongues who had relatively low proficiency levels, namely A1, A2 and

B1 of the Common European Framework (CEF). For the experiments reported on in this paper we used the spontaneous speech material.

Orthographic transcriptions were manually created and include (dis-)fluency phenomena such as filled pauses, restarts and repetitions. Grammatical errors were manually annotated. Furthermore, the annotators also entered the corresponding correct target utterance (see the examples presented below). For every utterance containing an error we thus have the realization and the corresponding correct target utterance.

The total number of utterances containing at least 1 error is 954. For the time being we selected only the 589 utterances (with 4150 words in the target utterances) that contain only 1 syntactical error. Note that in addition, the utterances often contain other errors, e.g. regarding morphology, pronunciation of sounds and prosody, disfluencies, etc.

2.2. Method

The general method for analyzing the non-native utterance on grammatical errors is called SynPOS: Syntactical analysis using POS-tags. It consists of the following four stages, carried out for each pair of utterances (target & realization):

- (1) Add POS-tags
- (2) Align words in the utterances
- (3) Match words in the utterances
- (4) Make an error list

Stages (1) and (2) are interchangeable, but are listed in this order because stages (2) + (3) together are for matching words for each pair of utterances (target & realization). The four stages are described in more detail in the following sections.

2.2.1. Add POS-tags

TADPOLE is a modular memory-based morphosyntactic tagger, analyzer and dependency parser for Dutch. TADPOLE is an acronym of 'TAgger, Dependency Parser, and mOrphoLogical analyzEr' [6, 8]. For the current research we only use the output of the part-of-speech (POS) tagger and the information about the lemmas. The POS-tags used are listed in Table 1. The first column contains the Dutch acronym, as obtained with TADPOLE, and the second column an English acronym and short description. An example of a realized utterance, its corresponding target, and the POS-tags of both utterances are provided in Figures 1 and 2.

2.2.2. Align words in the utterances

The program SCLITE is a tool for scoring and evaluating the output of automatic speech recognition (ASR) systems. SCLITE is part of the NIST SCTK Scoring Toolkit [7, 13]. The program SCLITE is generally used to compare the output of the ASR system to the correct target text. In our case, SCLITE is used to align the words (without using the POS-tags) for each pair of utterances. An example of the output for a pair of utterances is provided in Figure 2 (see the lines target, realization & SCLITE).

SCLITE results in an alignment of the two corresponding utterances, containing information on deletions (Del), Insertions (Ins), and Substitutions (Sub) (see Figure 2). However, this is not enough for our goals, as will become clear below. For instance, in some cases a combination of an insertion in one utterance and a deletion in the other utterance is a transposition (Tp). Therefore, some extra matching steps are needed, as described in the next section.

2.2.3. Match words in the utterances

Below a short description is presented of the different steps. The effect of these steps is illustrated in the example in Figure 2. First, position numbers are added to the words in the target, and if words are matched in the following steps position numbers from the target are copied to the realization.

*** step a. Match equal words aligned by SCLITE**

For words that match exactly (same position and form), copy the position number of the target to the realization. Obviously, the match is not yet complete, and therefore extra steps are needed.

*** step b. Match other equal words (except ART)**

In step b words (except words with the POS-tag ART) with the same form but on other positions are matched.

*** step c. Match words with equal lemmas (except ART)**

In step c words (except words with the POS-tag ART) with the same lemma are matched. For this step we use the lemmas obtained with TADPOLE (see section 2.2.1).

Steps b & c are not carried out for words with the POS-tag ART. The reason is that many utterances contain multiple articles, and non-native speakers make a lot of errors regarding articles (see Table 1). Treating articles in the same way as words with other POS-tags would result in many erroneous results. For instance, in the example in Figure 2, look at the two occurrences of the word “de”, which obviously should not be matched. They have the same form, and thus would be matched in step b; and they also have the same lemma and thus would be matched in step c. Matching of articles is resolved in the next steps.

*** step d. Match words with small Levenshtein distance**

Sometimes the orthographic representations of two words that should be matched differ slightly. The reason could be a typo, a pronunciation error, which in some cases is coded in the ‘orthographic’ representation, a morphological error, etc. To resolve these issues we match words for which the Levenshtein distance divided by the length of the longest word is smaller than or equal to 1/3. This is only done for pairs of words for which the length of the longest word is at least 4. Note that in this step also the POS-tag of the realization of the word “ZWIMBAD” (i.e. WW), is replaced by the correct POS-tag of the matching word “ZWEMBAD” (i.e. N) of the target utterance.

*** step e. Match words with equal POS-tags in matching post-word context**

Match words with same POS-tag and matching post-word context, i.e. of the following two words in the target utterance at least one of them should have been matched to one of the two following words in the realization.

*** step f. Match words in matching (surrounding and post-word) contexts**

In this final step, words are matched (see Figure 1) if

- either both surrounding (left & right) words match,
- or both following words match

target	en	TEN	derde	wil	ik	…
POStag	CON	PREP	NUM	VERB	PRON	…
lemma	en	ten	drie	willen	ik	…
pos.nr.	0	1	2	3	4	…
real.	en	DE	derde	wil	ik	…
POStag	CON	ART	NUM	VERB	PRON	
lemma	en	de	drie	willen	ik	…
pos.nr.	0	--	2	3	4	…
step f:	0	1	2	3	4	…

Figure 1: *Example illustrating the effect of step f.*

target	omdat	ik	ALTIJD	met	DE	bus	naar	HET	ZWEMBAD	GA
	because	I	always	with	the	bus	to	the	pool	go
POStag	CON	PRON	ADV	PREP	ART	N	PREP	ART	N	VERB
lemma	omdat	ik	altijd	met	de	bus	naar	het	zwembad	gaan
pos.nr.	0	1	2	3	4	5	6	7	8	9
real.	omdat	ik	GAAT	met	**	bus	naar	DE	ZWIMBAD	ALTIJD
	because	I	goes	with	--	bus	to	the	pool	always
SCLITE	=	=	Sub	=	Del	=	=	Sub	Sub	Sub
POStag	CON	PRON	VERB	PREP	-	N	PREP	ART	VERB	ADV
lemma	omdat	ik	gaan	met	-	bus	naar	de	zwimbad	altijd
pos.nr.										
step a	0	1	--	3	--	5	6	--	--	--
step b	0	1	--	3	--	5	6	--	--	2
step c	0	1	9	3	--	5	6	--	--	2
step d	0	1	9	3	--	5	6	--	8 (N)	2
step e	0	1	9	3	--	5	6	7	8 (N)	2
final	0	1	9	3	--	5	6	7	8 (N)	2
SynPOS	=	=	Tp+Sub	=	Del	=	=	Sub	Sub	Tp

Figure 2: *Made up example of a pair of utterances illustrating the method: the annotations and the effect of the various steps. SynPOS finds substitutions (Sub), deletions (Del), insertions (Ins), and transpositions (Tp). For further explanation see text.*

Table 1. *Absolute frequency and relative frequency (%) of syntactical errors. The columns contain the frequencies on Del, Sub, Tp, & Ins, the rows the frequencies for the different POS-tags.*

Dutch acronym	English acronym and description	Total	Del	Sub	Tp	Ins
		4150	399 (9.6%)	302 (7.3%)	212 (5.1%)	125 (3.0%)
LID	ART - article	350	**170 (48.6%)**	*20 (5.7%)*	1 (0.3%)	18 (5.1%)
VNW	PRON – pronoun	884	**133 (15.0%)**	**62 (7.0%)**	*32 (3.6%)*	18 (2.0%)
VZ	PREP – preposition	384	**38 (9.9%)**	**42 (10.9%)**	6 (1.6%)	**32 (8.3%)**
VG	CON - conjunction	198	10 (5.1%)	4 (2.0%)	1 (0.5%)	14 (7.1%)
WW	VERB - verb	853	*36 (4.2%)*	**81 (9.5%)**	**102 (12.0%)**	28 (3.3%)
BW	ADV - adverb	375	5 (1.3%)	*20 (5.3%)*	*27 (7.2%)*	3 (0.8%)
N	N - noun	608	5 (0.8%)	**47 (7.7%)**	19 (3.1%)	6 (1.0%)
ADJ	ADJ - adjective	358	2 (0.6%)	**62 (17.3%)**	*22 (6.1%)*	4 (1.1%)
TSW	INT - interjection	8	-	-	-	2 (25.0%)
SPEC	SPEC - special token	84	-	4 (4.8%)	2 (2.4%)	-
TW	NUM - numeral	48	-	-	-	-

2.2.4. Make an error list

After all the steps described above have been carried out the errors are annotated (see the row SynPOS), and a report with the results is generated. Some results are presented in the next section.

3. Results

An overview of the results obtained with our SynPOS method is presented in Table 1. For the syntactical errors we present both the absolute frequencies (the number of occurrences) and the relative frequencies (which were obtained by dividing the absolute frequencies by the number of occurrences of the POS-tags listed in the column 'Total').

The order of the results in Table 1 is as follows:

1. First in the columns: decreasing number of absolute and relative frequency, i.e. Del, Sub, Tp, and Ins.

2. Next in the rows: decreasing number of relative frequency (%) in the column Del.

It can be observed in Table 1 that many errors are found by our method: 399 (9.6%) deletions, 302 (7.3%) substitutions, and 212 (5.1%) transpositions; thus in total 913 (21.9%) of the words in the target are changed. In addition, 125 (3.0%) insertions were found. There are also many different types of errors, i.e. 35 in Table 1 (35 cells in Table 1 have a value larger than 0). Not all of these types of syntactical errors occur equally often. Deletion of articles occurs most often, both in terms of absolute and relative frequency; almost half of the articles are not realized.

These results can be useful for selecting syntactical errors for CALL systems. Frequency is obviously an important criterion, both absolute and relative frequency. Absolute and relative frequency can be combined, e.g., by simply multiplying their numbers. As an example, the values for which the product of these two numbers is larger than 2 are listed in bold in Table 1, and those for which the product is in

between 1 and 2 are in Italic. Of course, besides frequency other criteria could be used for selecting syntactical errors.

4. Discussion and conclusions

In the previous sections we have presented a new method, called SynPOS, to analyze syntactical errors in speaking performance for the purpose of developing CALL exercises for practicing syntax in Dutch L2 spoken interaction. SynPOS yields clear and plausible results that are in line with previous findings, especially with respect to the frequent syntactical errors we found. It seems therefore that SynPOS can be employed to analyze corpora to identify syntactical errors together with quantitative information. These results can then be used to select syntactical errors, and subsequently to develop a system for practicing the more problematic L2 syntactical phenomena. For example, the quantitative information can be employed to develop a language model (LM) for the ASR with different probabilities for the options (paths) present in the language model. In the current research the method is applied to Dutch utterances. However, the proposed method can also be applied to other languages, if POS taggers exist for those languages.

The next thing we are going to study is finding patterns in the results, patterns that generalize from our current data to other data, and thus can be used for system development. For deletions and substitutions (the largest classes) the situation is probably straightforward: the position of these words in the target utterances is known, and these words can simply be deleted or substituted. In the LM of the ASR we can then add extra arcs (paths), possibly with the corresponding probabilities. However, in the case of insertions and transpositions we have to find patterns that make clear where the words could appear (given the syntactical errors that non-natives make). Maybe the information we have at the moment is not rich enough to make this possible to a sufficient degree. If that turns out to be the case, we will consider gathering extra information. An obvious alternative would be to use a syntactic parser, e.g. for Dutch the Alpino parser [12]. A disadvantage of using a syntactic parser is that its output may contain more errors than the output of a POS-tagger, even for the correct target utterance. In any case, given that at the moment there probably is no method that can correctly analyze utterances spoken by non-natives that contain errors, it is probably best to use a correct target and its analysis as a reference, as we did in the current method with POS-tags.

In the first three stages of this method some errors are made. We manually checked tags and lemmas of 50 pairs of utterances. The 50 target utterances contained 394 words in total, out of which 15 words (4%) received an incorrect POS-tag from TADPOLE. Of these 15 words, 10 belong to two classes that were often tagged incorrectly, i.e. (1) "het weer" (the weather) which should be tagged as 'ART N', and (2) some adjectives tagged as adverbs. Often, when the POS-tag is incorrect, the lemma is also incorrect; for the words with correct POS-tag the lemma was generally correct as well. For the POS-tags and lemmas we could use other resources, but they probably will contain other errors, and it is not likely that the net gain will be very large. Furthermore, the alignments produced by SCLITE are not always optimal. SCLITE offers some possibilities to improve the alignment, for instance by using Levenshtein distance. For the present experiments we used the standard 'basic' version of SCLITE. However, the alignment errors that can be resolved in this way probably are already resolved in our stage 3. Finally, for all 589 target-realization pairs, for which the target utterances contain 4150 words, only 12 matching errors were found, i.e. for only 2.0% of the utterance and 0.29% of the words. Consequently, the number of errors made by SynPOS is small, and some of the errors made in stages 1 and 2 are resolved in stage 3. Still, there might be room for some improvement, but a more thorough analysis requires a larger corpus, and it is not likely that this will result in substantial changes in the analysis results, especially not in the frequent syntactical errors found.

We could also use more fine-grained POS-tags, for instance within the class of pronouns we could discern personal pronoun, demonstrative pronoun, etc. For the analysis this is not necessary, but it may be useful for finding patterns in the results. However, for finding patterns the biggest gain can probably be obtained by using a syntactic parser, as was already mentioned above.

We intend to study these issues in future research. We will also look at utterances containing more than 1 syntactical error. Finally, we will use the information obtained with SynPOS to develop and test ASR-based CALL exercises to train syntax in spoken language in the projects DISCO [9] and FASOP [11]: we will select syntactical errors, develop exercises to train these aspects, develop the technology to handle the spoken replies automatically, analyze them, and provide feedback, and finally compare and test the effect of providing feedback in different ways.

5. References

References to papers and URLs, listed in alphabetical order.

[1] Cucchiarini, C., Driesen, J., Van Hamme, H. and Sanders, E. (2008) "Recording speech of children, non-natives and elderly people for HLT applications: the JASMIN-CGN corpus", Proceedings of LREC-2008.

[2] Cucchiarini, C., Neri, A. and Strik, H. (2009) "Oral Proficiency Training in Dutch L2: the Contribution of ASR-based Corrective Feedback", Speech Communication, pp. 853-863, , Volume 51, Issue 10, October 2009.

[3] De Jong, N. (2005), "Can second language grammar be learned through listening? An Experimental Study", Studies in Second Language Acquisition, 27, 205–234, 2007.

[4] Ellis, N.C., and Bogart, P.S.H. (2007), "Speech and LanguageTechnology in Education: the perspective from SLA research and practice", Proceedings ISCA ITRW SLaTE, Farmington PA.

[5] Lee, J., and Seneff, S. (2006) "Automatic grammar correction for second-language learners", Proceedings of Interspeech 2006.

[6] Van den Bosch, A., Busser, G.J., Daelemans, W. and Canisius, S. (2007) "An efficient memory-based morphosyntactic tagger and parser for Dutch", in F. van Eynde, P. Dirix, I. Schuurman, and V. Vandeghinste (Eds.), , pp. 99-114, Selected Papers of the 17th Computational Linguistics in the Netherlands Meeting, Leuven, Belgium, 2007.

[7] ftp://jaguar.ncsl.nist.gov/current_docs/sctk/doc/sclite.htm

[8] http://ilk.uvt.nl/tadpole/

[9] http://lands.let.ru.nl/~strik/research/DISCO

[10] http://lands.let.ru.nl/~strik/research/Dutch-CAPT/

[11] http://lands.let.ru.nl/~strik/research/FASOP.html

[12] http://www.let.rug.nl/vannoord/alp/Alpino/

[13] http://www.nist.gov/itl/iad/mig/tools.cfm