Manual Morphological Annotation of Slovak Translation of Orwell's Novel 1984 – Methods and Findings

Radovan Garabík[1]
Lucia Gianitsová-Ološtiaková[2]

[1] Ľudovít Štúr Institute of Linguistics
Slovak Academy of Sciences
Bratislava, Slovakia
korpus@korpus.juls.savba.sk
http://korpus.juls.savba.sk

[2] University of St. Cyril and Methodius
Trnava, Slovakia
gianitsova@zoznam.sk

Abstract. Manual morphological text annotation is indisputably an important part of building a framework of NLP tools used in corpora construction. From 2004 to 2005, the complete text of Orwell's 1984 novel, some Slovak Wikipedia texts and some newspaper articles have been annotated. In the paper we present the methodology used in manual annotation and correction of annotated data, and the discussion of obtained results.

Manual morphological text annotation of the Slovak National Corpus is a part of an intense work made as a part of constructing a corpus. It represents another processing of corpus data, providing rich information about language and its usage. The importance of exact manually annotated data for subsequent computer processing of morphology is indisputable. For that reason during the years 2003 – 2005 a great attention has been given to this phase of corpus construction.

During the introductory phase (in 2003) after the first theoretical discussions[1] about morphological tagging a tagset described in [2] has been designed.

The second phase, a manual annotation (2004 – 2005) started after confrontation with a real text material, using the annotation rules described in [3]. From February 2004 to June 2005 a manual lemmatization and tagging have been carried out using the complete texts of the Orwell's novel *1984*, samples from *InZine* (internet magazine), *Wikipedia* (internet encyclopædia) and *SME* (daily newspaper). The annotation was done by students of the Faculty of Philosophy, Comenius University, Bratislava. The number of students varied from 2 at the beginning to 11 at the end. Though manual annotation is a time-consuming work, following texts containing 215 000 tokens have been annotated: Orwell's *1984* (102 000 tokens), Slovak *Wikipedia* (50 000 tokens), *SME* daily (about 21 000 tokens), internet magazine *InZine* (more than 42 000 tokens).

The paper deals with our experiences with manual annotation acquired during annotation of Orwell's novel *1984*. Attention is also paid to the description of some fundamental methods applied to correction and finalization of manual annotations.

The files being annotated are conforming to XML TEI XCES standard[4]. We have created a GUI program written in python-gtk, using ElementTree library to parse and modify the XML files[5], used to manually annotate the files, called *Anno*. The program displays list of words (tokens) in the file, and for each selected token a list of possible lemmas and tags. The user either selects a corresponding pair of lemma and tag (disambiguation), or if none are provided or suitable, he/she can enter or fix the lemma and tag directly.

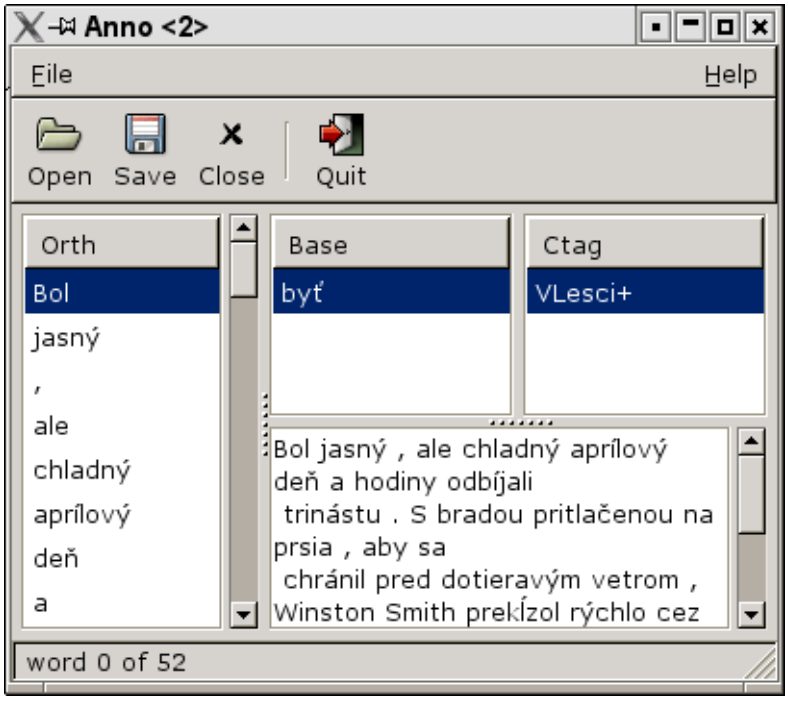

Fig. 1. Anotation tool Anno

At the beginning the annotation speed was about 80 tokens per hour, but after the annotation tool has been tuned according to the requirements of the annotators, a remarkable acceleration of the annotation (up to 200 – 250 tokens/hour) occurred. The possibility to work with automatically pre-tagged texts has been extremely advantageous. Pre-tagging has been done by using the morphological tagger described in [6], with combination with the TNT statistical tagger[7] trained on already annotated texts. Consequently, the annotators needed to focus their attention only at verification and correction of lemmas and tags. This significantly increased the accuracy of a manual annotation. At the beginning it was 84 per cent (partly due to changes being made in the tagset specification and annotation principles). After transition into a new annotation tool the accuracy of manual annotation increased to 92 per cent.

Each annotator was given a part of the novel 1984, at the beginning in chunks of text containing about 100 – 200 words, later expanded to 500 and more words. Completed parts have been gathered into a single-unit text, which became a subject of verification and unification of various token interpretations. During the first annotation phase (first 20 000 tokens) each annotator obtained one chunk of text, and then the whole text was checked by a linguist responsible for morphological annotation. Out of the corpus with 21 500 tokens there were 2 952 mistakes (14 per cent of the annotated text) detected. However, this method turned out not to be very effective because of its slow speed (1000 – 1200 tokens/day) and a high demand put on the linguist. Moreover, with increasing the text length the percentage of unspotted mistakes was increasing. Later, if was found out that 350 mistakes and incorrect interpretations (1.6 per cent of the annotated text) have not been detected.

However, later we used the method used when annotating the texts in the Prague Dependency Treebank – each file is annotated by two persons and results are automatically compared and subsequently checked and disambiguated only by one annotator[8]. Consequently, in the following phase the text samples have been given to two annotators and we focused on correcting just the differences in the annotation, with the use of command line utility `diffxces.py`, providing a diff(1)-like comparison of two XCES files.

1	Odkiaľsi	odkiaľsi	Dx
	Odkiaľsi	odkiaľsi	PD
17	Kávy	káva	SSfs2
	Kávy	káva	SSfs2x:r
21	sa	sa	R
	sa	sa	Z
47	akoby	akoby	OY
	akoby	akoby	O
85	byť	byť	VIe+
	byť	byť	VKe+
107	jedine	jedine	Dx
	jedine	jedine	T
124	na	na	Eu6
	na	na	Eu4
153	aj	aj	O
	aj	aj	T
160	*	*	#
	*	*	Z
186	väčšmi	väčšmi	Dx
	väčšmi	veľa	Dy

Table 1. Example of output of comparing two XCES files

Unfortunately, this method also turned out to be inconvenient, because often the detected errors had origin in an insufficient practical morphological skill of one of the annotators (it is represented by 1118 tokens out of 15 061 tokens and it makes 58 per cent of detected differences). We needed just to have the text annotated by a different annotator and then verify only his/her annotation. Moreover, a comparison of controversial cases and correction of those which really required it (from original 1921 differences only 803 ones required correction) has been a time-consuming work. Accordingly, results indicated the effectiveness of manual annotation up to 95 per cent. On the other hand, many mistakes have not been detected because annotators often made the same mistakes. These misinterpretations occurred especially in cases of part-of-speech homonymy – conjunctions and particles, adverbs and particles, in cases of wrong indication of homonymous nominative and accusative, genitive and accusative and similar grammar categories. In the sample the number of non-detected mistakes was 387, i. e. about 3 per cent of all the tokens. This kind of mistakes is typical of Slovak language grammar analysis, regardless of the linguistic level of the person doing the analysis – from elementary school pupils up to the university students. This reason reduced the annotation accuracy down to 92 per cent and was the main reason for the fact that a comparison of two annotations eliminates on average only 67 per cent out of all mistakes, the remaining 33 per cent is not detected.

Third phase of a final verification after various experiments started in January 2005. We made use of additional semi-automatized verification tools, to check out the annotated files. However, these tools have to be supplemented by a manual correction anyway. Each tool was designed to check out one specific class of mistakes. Our verification process included three phases:

1st phase, using tool named `checkxcestags.py`: removal of superfluous whitespace in lemmas (in the table below tokens number 267 and 2932), automatic checking of correct tag length and correct combination of characters in tags, e. g. a missing tag (token number 11637), an unknown tag (token number 7818), a missing tag for the level of adjectives (token number 1333), a missing tag for the congruence in gender of -l- participle (token number 503), an unknown tag for the category (token number 5856), a redundant tag (tokens number 126 and 3057), missing tags for categories (token number 2990), inappropriate gender for the given pronoun type, or person of verbs (tokens number 651 and 2500), wrong type of paradigm (token number 3332):

```
  126 nejakej         nejaký          PAfs6x          Bad length
  267 "                "              Z               Spaces in lemma/orth
  503 bola            byť             VLesc+          Bad length
  651 mi              ja              PPms3           Bad gender
 1333 tučné           tučný           AAfp4           Bad length
 1456 sú              byť             VKefp+          Bad number
 2500 zažili          zažiť           VLdpbm+         Bad gender
 2932 —                —              Z               Spaces in lemma/orth
 2990 niekoľko        niekoľko        PU              Bad length
 3057 deviatich       deväť           NUip2w          Bad length
 3332 ich             on              PPmp4           Bad gender
 5856 pohrá           pohrať VKmsc+                   Bad aspect
 6057 slúžia          slúžiť VKepci+                  Bad length
 7818 II              II              C}------------- Bad POS
11637 ,               ,               None            Not string
```

This method made it possible to eliminate some repeating mistakes and obvious incorrect interpretations of tagging manual. The tool is based on some general properties of certain grammar categories encoded in the tag. Out of all the mistakes, 28% were corrected in this phase.

2nd phase: We generated lists of unique triplets (token, lemma, tag) from the text, using tool named `cesstat-tab.py`. We then sorted the list either by lemma or by the tag, thus making it possible to easily spot any discrepancies. This phase decreased significantly tag assignment inconsistency, most notably mistakes with wrong indication of paradigm, gender, case or number. Overall, we corrected about 31 % of all the mistakes using this method.

Lemma	**Token**	**Tag**	**Correction**
akoby	Akoby	O	tag = OY
akoby	akoby	OY	
blízky	bližšie	AAfp1x	tag = AAfp1y
blízky	bližšie	AAns4y	
byť	bude	VBesc+	
byť	bude	VKesc+	tag = VBesc+
celý	celé	AAns4x	
celý	Celý	AAns4x	tag = AAis4x
čo	čo	PD	tag = PFns1
čo	čo	PFns1	
dav	dav	SSis4	
dav	dav	SSms1	tag = SSis1
do	do	Eu2	
do	do	Eu4	tag = Eu2
hľadieť	hľadelo	VLescn+	
hľadieť	hľadieť	VId+	tag = VIe+
indický	Indický	AAis4x	
indický	Indického	AAis2x:r	tag = AAis2x
iný	iných	AAmp2x	tag = PAmp2
iný	iný	PAms1	
katharine	Katharine	SSfs4:r	tag = SUfs4:r
katharine	Katharine	SUfs1:r	
každý	Každé	PAns4	
každý	každého	NAns2	tag = PAns2
nedefinovateľný	nedefinovateľnými	Gtip7x	tag = AAip7x
nedefinovateľný	nedefinovateľného	AAns2x	
niekoľko	niekoľko	NUns4	tag = PUns4
niekoľko	niekoľko	PUns4	
otvorený	otvorený	Gtis4x	
otvorený	otvorenou	AAfs7x	tag = Gtfs7x
predtým	predtým	Dx	
predtým	predtým	PD	tag = Dx
prsia	prsia	SSfp4	tag = SSnp4
prsia	prsia	SSnp1	
winston	Winstona	SSms2:r	
winston	Winstonom	SSms7	tag = SSms7:r

Table 2. List of triplets sorted by the lemma

Tag	Lemma	Token	Correction
AAmp1x	bezpečný	bezpeční	
AAmp1x	mladý	mladí	
AAmp1x	mladý	mladý	tag = AAms1x
Dx	celkom	celkom	
Dx	celok	celkom	lema = celkom
Eu6	v	V	
Eu6	v	vo	tag = Ev6
Gtfs1x	preťať	preťatá	lema = preťatý
PAis7	ktorý	ktorý	tag = PAis1
PAis7	nejaký	nejakým	
PAis7	niektorý	niektorým	
PAis7	nijaký	nijakým	
PFmp1	ten	tí	
PFmp1	ten	tých	tag = PFmp2
SSfs4	zákonnosť	zákonnosť	
SSfs4	záležitosť	záležitosť	
SSfs4	záležitosť	záležitosti	tag = SSfp4
SSms1	pán	pán	
SSms1	pán	pána	tag = SSms2
VIe-	necivieť	necivieť	
VIe-	neexistovať	neexistoval	tag = VLescm-
VIe-	nemyslieť	nemyslieť	
VKdpc+	pobiť	pobijú	
VKdpc+	podariť	podarí	tag = VKdsc+
VKdpc+	pokaziť	pokazia	
VKdsc+	vybrať	vyberie	
VKdsc+	vyčistiť	vyčistím	tag = VKdsa+
VKdsc+	vydobiť	vydobije	

Table 3. List of triplets sorted by the tag

After the final text sample (about 40 000 tokens) has been corrected, this verification method has been replaced by a method of generating only a list of those triplets (token, lemma, tag) that did not occur in previously corrected texts. A pair of tools have been used for this purpose. The first one, `make3.py`, makes a pickled list of all existing triplets from the XCES files given as parameters to the program, and subsequently the second program, check3.py loads the pickled list and prints the triplets that are present in a XCES file given as a parameter but that are not present in the pickled lists. The annotator then verifies only these suspicious triplets. This phase had significantly reduced the number of inconsistencies including token interpretations and some mistakes not detected during a routine check.

3rd phase: quick visual check of annotated text, using annotation tool.

In this phase the attention was focused upon mistakes where the correct grammar categories can be found out only taking into account context of the word (case, person,

part-of-speech homonymy). The speed of this checking was about 1500 – 2000 tokens per hour and remaining 41 per cent of all mistakes were removed.

After implementation of the presented verification model from January to June 2005 more than 102 000 tokens were checked and corrected, i. e. the whole Orwell's novel 1984. Currently, the Slovak National Corpus uses this methodology for verification of further manual morphology annotation. In our opinion, this system proved to be able to provide positive results and improved texts verification. Its advantages could be seen especially in an implementation of semi-automatised methods that interactively (along with the manual control) participate in detecting ambiguities of manual annotation.

Acknowledgements

Publication of this article has been a part of the grant Morphosyntactic research in the Slovak National Corpus, MŠ SR VEGA 1/3149/04

References

1. Forróová, M., Horák, A.: Morfologická anotácia korpusu. In: *Proceedings of the Conference Slovenčina na začiatku 21. storočia*. Prešov, Fakulta humanitných a prírodných vied PU (2004) 174–183
2. Forróová, M., Garabík, R., Gianitsová, L., Horák, A., Šimková M.: Návrh morfologického tagsetu SNK. In: *Proceedings of International Conference Slovko 2003 – Slovanské jazyky v počítačovom spracovaní*. Bratislava (2003). To be published.
3. Garabík, R., Gianitsová, L., Horák, A., Šimková M.: Tokenizácia, lematizácia a morfologická anotácia Slovenského národného korpusu. (Current version of May 4, 2004) SNK JÚĽŠ, Bratislava.
   http://korpus.juls.savba.sk/publikacie/Tagset-aktualny.pdf
4. Ide, N., Bonhome, P., Romary, L., XCES: An XML-based Encoding Standard for Linguistic Corpora. In: *Proceedings of the Second International Language Resources and Evaluation conference*. Paris, European Language Resources Association (2000)
5. Garabík, R.: Processing XML Text with Python and ElementTree – a Practical Experience. Bratislava, Ľ. Štúr Institute of Linguistics (2005). To be published.
6. Hajič, J., Hric, J., Kuboň, V.: Machine Translation of Very Close Languages. In: *Proceedings of the ANLP 2000*. Seattle, U.S.A. (2000)
7. Brants, T.: TnT – a statistical part-of-speech tagger. In: *Proceedings of the the ANLP 2000*. Seattle, U.S.A. (2000)
8. Hajič, J., Hladká, B., Pajas, P.: The Prague Dependency Treebank: Annotation Structure and Support. In: *Proceedings of the IRCS Workshop on Linguistic Databases*. Philadelphia, University of Pennsylvania (2001) 105–114