Morphological annotation of Old and Middle Hungarian corpora

Attila Novák[1,2] **György Orosz**[2] **Nóra Wenszky**[2]
[1]Research Institute for Linguistics, Hungarian Academy of Sciences
Benczúr u. 33., Budapest, Hungary
[2]MTA-PPKE Natural Language Research Group
Faculty of Information Technology, Pázmány Péter Catholic University
Práter u. 50/a, Budapest, Hungary
{novak.attila,oroszgy}@itk.ppke.hu, wenszkynora@gmail.com

Abstract

In our paper, we present a computational morphology for Old and Middle Hungarian used in two research projects that aim at creating morphologically annotated corpora of Old and Middle Hungarian. In addition, we present the web-based disambiguation tool used in the semi-automatic disambiguation of the annotations and the structured corpus query tool that has a unique but very useful feature of making corrections to the annotation in the query results possible.

1 Introduction

One of the aims of two parallel OTKA projects of the Research Institute for Linguistics of the Hungarian Academy of Sciences[1] is to create morphologically analyzed and searchable corpora of texts from the Old Hungarian and Middle Hungarian period. In the course of the projects, the Humor Hungarian morphological analyzer (Novák, 2003; Prószéky and Novák, 2005) was extended to be capable of analyzing words containing morphological constructions, suffix allomorphs, suffix morphemes, paradigms or stems that were used in Old and Middle Hungarian but no longer exist in present-day Hungarian. In the sections below, we describe how the morphological analyzer was adapted to the task, the problems we encountered and how they were solved. We also present the automatic and the manual disambiguation system used for the morphosyntactic annotation of texts and the corpus manager with the help of which the annotated corpora can be searched and maintained.

[1]*Hungarian historical generative syntax* [OTKA NK78074], and *Morphologically analysed corpus of Old and Middle Hungarian texts representative of informal language use* [OTKA 81189]

2 Preprocessing

The overwhelming majority of extant texts from the Old Hungarian period are codices, mainly containing texts translated from Latin. The texts selected for the Corpus of Informal Language Use, however, are much closer to spoken language: minutes taken at court trials, such as witch trials, and letters sent by noblemen and serfs. In the case of the latter corpus, metadata belonging to the texts are also of primary importance, as these make the corpus fit for historical-sociolinguistic research.

2.1 Digitization

All the texts selected for our corpora were originally hand-written. However, the basis for the digitized version was always a printed edition of the texts published earlier. The printed texts were scanned and converted to a character stream using OCR. This was not a trivial task, especially in the case of Old Hungarian texts, owing to the extensive use of unusual characters and diacritics. In the lack of an orthographic norm, each text applied a different set of characters; moreover, the printed publications used different fonts. Thus the only way to get acceptable results was to retrain the OCR program[2] for each text from scratch since the out-of-the-box Hungarian language and glyph models of the software did not fit any of our texts. Subsequently, all the automatically recognized documents had to be manually checked and corrected, but even so, this workflow proved to be much faster than attempting to type in the texts.

2.2 Normalization

The next step of preprocessing was normalization, i.e. making the texts uniform regarding their orthography and phonology. Normalization, which

[2]We used FineReader, which makes full customization of glyph models possible, including the total exclusion of out-of-the-box models.

Proceedings of the 7th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities, pages 43–48,
Sofia, Bulgaria, August 8 2013.

was done manually, in our case meant modernization to present-day orthography. Note that this also implies differences in tokenization into individual words between the original and the normalized version. During this process, which also included segmentation of the texts into clauses, certain phonological dialectal variations were neutralized.

Morphological variation, however, was left untouched: no extinct morphemes were replaced by their present day counterparts. We also retained extinct allomorphs unless the variation was purely phonological. In the case of potential irresolvable ambiguity, the ambiguity was preserved as well, even if it was due to the vagueness of the orthography of the era.

An example of this is the non-consistent marking of vowel length. The definite and indefinite 3rd person singular imperfect of the frequently used word *mond* 'say' was *mondá* $\sim$ *monda* respectively, but accents are often missing from the texts. Furthermore, in many texts in the corpus, these two forms were used with a clearly different distribution from their present day counterparts *mondta* $\sim$ *mondott*. Therefore, in many cases, neither the orthography, nor the usage was consistent enough to decide unambiguously how a certain appearance of *monda* should be annotated concerning definiteness.

Another example of inherent ambiguity is a dialectal variant of possessive marking, which is very frequent in these corpora and often neutralizes singular and plural possessed forms. For example, *cselekedetinek* could both mean 'of his/her deed' or 'of his/her deeds', which in many cases cannot be disambiguated based on the context even for human annotators. Such ambiguous cases were annotated as inherently ambiguous regarding number/definiteness etc.

2.3 Jakab's databases

Some of the Old Hungarian codices (Jókai (Jakab, 2002), Guary (Jakab and Kiss, 1994), Apor (Jakab and Kiss, 1997), and Festetics (Jakab and Kiss, 2001)) were not digitized using the OCR technique described above, as these were available in the form of historical linguistic databases, created by Jakab László and his colleagues between 1978 and 2002. However, the re-creation of the original texts out of these lexical databases was a difficult task. The first problem was that, in the databases, the locus of word token occurrences only identified codex page, column and line number, but there was no information concerning the order of words within a line. The databases also contain morphological analyses, but they were encoded in a hard-to-read numerical format, which occasionally was incorrect and often incomplete. Furthermore, the categorization was in many respects incompatible with our system. However, finally we managed to re-create the original texts. First the order of words was manually restored and incomplete and erroneous analyses were fixed. Missing lemmas were added to the lexicon of the adapted computational morphology, and the normalized version of the texts was generated using the morphology as a word form generator. Finally, the normalized texts were reanalyzed to get analyses compatible with the annotation scheme applied to the other texts in the corpora.

3 The morphological analyzer

The digitized and normalized texts have been analyzed with an extended version of the Humor analyzer for Hungarian. The lexicon of lemmas and the affix inventory of the program have been augmented with items that have disappeared from the language but are present in the historical corpora. Just the affix inventory had to be supplemented with 50 new affixes (not counting their allomorphs).

Certain affixes have not disappeared, but their productivity has diminished compared to the Old Hungarian era. Although words with these morphemes are still present in the language, they are generally lexicalized items, often with a changed meaning. An example of such a suffix is *–At*, which used to be a fully productive nomen actionis suffix. Today, this function belongs to the suffix *–Ás*. The (now lexicalized) words, however, that end in *–At* mark the (tangible) result of an action (i.e. nomen acti) in present-day standard Hungarian, as in *falazat* 'wall' vs. *falazás* 'building a wall'.

One factor that made adaptation of the morphological model difficult was that there are no reliable accounts on the changes of paradigms. Data concerning which affix allomorphs could be attached to which stem allomorphs had to be extracted from the texts themselves. Certain morphological constructions that had already disappeared by the end of the Old Hungarian era were

rather rare (such as some participle forms) and often some items in these rare subparadigms have alternative analyses. This made the formal description of these paradigms rather difficult.

However, the most time consuming task was the enlargement of the stem inventory. Beside the addition of a number of new lemmas, the entries of several items already listed in the lexicon of the present-day analyzer had to be modified for our purposes. The causes were various: some roots now belong to another part of speech, or in some constructions they had to be analyzed differently from their present analysis.

Furthermore, the number of pronouns was considerably higher in the examined period than today. The description of their extensive and rather irregular paradigms was really challenging as some forms were underrepresented in the corpora.

Some enhancements of the morphological analyzer made during the corpus annotation projects were also applicable to the morphological description of standard modern Hungarian. One such modification was a new annotation scheme applied to time adverbials that are lexicalized suffixed (or unsuffixed) forms of nouns, like *reggel* 'morning/in the morning' or *nappal* 'daytime/in daytime', quite a few of which can be modified by adjectives when used adverbially, such as *fényes nappal* 'in broad daylight'. This latter fact sheds light on a double nature of these words that could be captured in an annotation of these forms as specially suffixed forms of nouns instead of atomic adverbs, an analysis that is compatible with X-bar theory (Jackendoff, 1977).

4 Disambiguation

With the exception of already analyzed sources (i.e. the ones recovered from the Jakab databases), the morphological annotation had to be disambiguated. The ambiguity rate of the output of the extended morphological analyzer on historical texts is higher than that for the standard Humor analyzer for present-day corpora (2.21 vs. 1.92[3] analyses/word with an identical (high) granularity of analyses). This is due to several factors: (i) the historical analyzer is less strict, (ii) there are several formally identical members of the enlarged verbal paradigms including massively ambiguous subparadigms like that of the passive and the factitive,[4] (iii) a lot of inherent ambiguities described above.

The workflow for disambiguation of morphosyntactic annotation was a semi-automatic process: an automatically pre-disambiguated version of each text was checked and corrected manually. For a very short time, we considered using the Jakab databases as a training corpus, but recovering them required so much development and manual labor and the analyses in them lacked so much distinction we wanted to make that we opted for creating the training data completely from scratch instead.

4.1 The manual disambiguation interface

To support the process of manual checking and the initial manual disambiguation of the training corpus a web-based interface was created using JavaScript and Ajax where disambiguation and normalization errors can be corrected very effectively. The system presents the document to the user using an interlinear annotation format that is easy and natural to read. An alternative analysis can be chosen from a pop-up menu containing a list of analyses applicable to the word that appears when the mouse cursor is placed over the problematic word. Note that the list only contains grammatically relevant tags and lemmas for the word returned by the morphological analyzer. This is very important, since, due to the agglutinating nature of Hungarian, there are thousands of possible tags (see Figure 1).

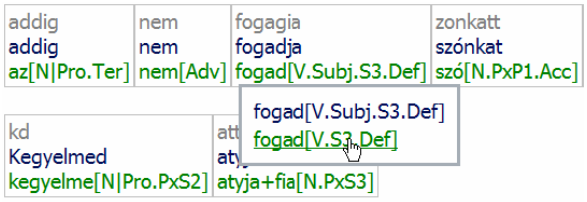

Figure 1: The web-based disambiguation interface

The original and the normalized word forms as well as the analyses can also be edited by clicking them, and an immediate reanalysis by the morphological analyzer running on the web server can be initiated by double clicking the word. We use Ajax technology to update only the part of the page belonging to the given token, so the update is immediate. Afterwards, a new analysis can be selected from the updated pop-up menu.

[3] measured on newswire text

[4] This ambiguity is absent from modern standard Hungarian because the passive is not used any more.

As there is an inherent difference between the original and normalized tokenization, and because, even after thorough proofreading of the normalized version, there may remain tokenization errors in the texts, it is important that tokens and clauses can also be split and joined using the disambiguation interface.

The automatic annotation system was created in a way that makes it possible that details of the annotation scheme be modified in the course of work. One such modification was e.g. the change to the annotation of time adverbs mentioned in Section 3 above. The modified annotation can be applied to texts analyzed and disambiguated prior to the modification relatively easily. This is achieved by the fact that, in the course of reanalysis, the program chooses the analysis most similar to the previously selected analysis (based on a letter trigram similarity measure). Nevertheless, the system highlights all tokens the reanalysis of which resulted in a change of annotation, so that these spots can be easily checked manually. For changes in the annotation scheme where the simple similarity-based heuristic could not be expected to yield an appropriate result (e.g. when we decided to use a more detailed analysis of derived verb forms as before), a more sophisticated method was devised to update the annotations: old analyses were replaced using automatically generated regular expressions.

4.2 Automatic disambiguation

While the first few documents were disambiguated completely manually using the web-based tool, we soon started to train and use a tagger for predisambiguation applying the tagger incrementally, trained on an increasing number of disambiguated and checked text. First the HMM-based trigram tagger HunPos (Halácsy et al., 2007) was used. HunPos is not capable of lemmatization, but we used a straightforward method to get a full analysis: we applied reanalysis to the text annotated only by the tags assigned by HunPos using the automatic similarity-based ranking of the analyses. This approach yielded quite good results, but one problem with it was that the similarity-based ranking always prefers shorter lemmas, which was not appropriate for handling the case of a frequent lemma ambiguity for verbs with one of the lemma candidates ending in an *–ik* suffix and the other lacking a suffix (such as *dolgozik* 'work' vs. *(fel)dolgoz* 'process'). Always selecting the *–ik*-less variant is not a good bet in the case of many frequent words in this ambiguity class.

Recently, we replaced HunPos with another HMM-based trigram tagger, PurePos (Orosz and Novák, 2012), that has many nice extra features. It can process morphologically analyzed ambiguous input and/or use an integrated analyzer constraining possible analyses to those proposed by the analyzer or read from the input. This boosts the precision of the tagger dramatically in the case of languages like Hungarian and small training corpora. The fact that PurePos can be fed analyzed input makes it easy to combine with constraint-based tools that can further improve the accuracy of the tagging by handling long distance agreement phenomena not covered by the trigram model or simply removing impossible tag sequences from the search space of the tool.

PurePos can perform lemmatization, even for words unknown to the morphological analyzer (and not annotated on the input) learning a suffix-based lemmatization model from the training corpus along with a similar suffix-based tag guessing model, thus it assigns a full morphological analysis to each token. It is also capable of generating an n-best list of annotations for the input sentence when using beam search instead of the default Viterbi decoding algorithm.

4.3 Disambiguation performance

We performed an evaluation of the accuracy of PurePos on an 84000-word manually checked part of the historical corpus using five-fold cross-validation with a training corpus of about 67000 words and a test corpus of about 17000 words in each round. The ratio of words unknown to the MA in this corpus is rather low: 0.32%.

The average accuracy of tagging, lemmatization and full annotation for different versions of the tagger are shown in Table 1. In addition to token accuracy, we also present sentence accuracy values in the table. Note that, in contrast to the usual way of evaluating taggers, these values were calculated excluding the always unambiguous punctuation tokens from the evaluation. The baseline tagger uses no morphological information at all. Its current lemmatization implementation uses suffix guessing in all cases (even for words seen in the training corpus) and selects the most frequent lemma, which is obviously not an ideal

solution.

The disambiguator using morphology performs significantly better. Its clause-level accuracy is 81.50%, which means that only every fifth clause contains a tagging error. The tag set we used in the corpus differentiates constructions which are not generally differentiated at the tag level in Hungarian corpora, e.g. deictic pronouns (*ebben* 'in this') vs. deictic pre-determiners (*ebben a házban* 'in this house'). Many of these can only be disambiguated using long-distance dependencies, i.e. information often not available to the trigram tagger. Combination of the tagger with a constraint-based tool (see e.g. Hulden and Francom (2012)) would presumably improve accuracy significantly.

In the rightmost column, we listed a theoretical upper limit of the performance of the current trigram tagger implementation using 5-best output and an ideal oracle that can select the best annotation.

		baseline	morph	5-best+o
token	Tag	90.17%	96.44%	98.97%
	Lem.	91.52%	98.19%	99.11%
	Full	87.29%	95.90%	98.53%
clause	Tag	62.48%	83.81%	93.99%
	Full	54.68%	81.50%	91.47%

Table 1: Disambiguation performance of the tagger

5 Searching the corpus

The web-based tool we created as a corpus query interface does not only make it possible to search for different grammatical constructions in the texts, but it is also an effective correction tool. Errors discovered in the annotation or the text appearing in the "results" box can immediately be corrected and the corrected text and annotation is recorded in the database. Naturally, this latter functionality of the corpus manager is only available to expert users having the necessary privileges.

A fast and effective way of correcting errors in the annotation is to search for presumably incorrect structures and to correct the truly problematic ones at once. The corrected corpus can be exported after this procedure and the tagger can be retrained on it.

The database used for the corpus manager is based on the Emdros corpus manager (Petersen, 2004). In addition to queries formulated using MQL, the query language of Emdros, either typed in at the query box or assembled using controls of the query interface, advanced users can use a custom-made corpus-specific query language (MEQL), which makes a much more compact formulation of queries possible than MQL. It is e.g. extremely simple to locate a specific locus in the corpus: one simply needs to type in the sequence of words one is looking for. Queries formulated in MEQL are automatically converted to MQL queries by the query processor.

The search engine makes it possible to search inside sentences, clauses, or texts containing grammatical constructions and/or tagged with metadata matching the criteria specified in the query. Units longer than a sentence can also be searched for. The context displayed by default for each hit is the enclosing sentence with focus words highlighted. Clauses may be non-continuous: this is often the case for embedded subordinate clauses, but the corpus also contains many injected parenthetical coordinate clauses and many examples where the topic of a subordinate clause precedes its main clause with the net effect of the subordinate clause being interrupted by the main clause. The query example in Figure 2 shows a sentence containing several clauses with gaps: the clauses enclosed in angle brackets are wedged between the topic and comment part of the clauses which they interrupt. Emdros is capable of representing these interrupted clauses as single linguistic objects with the interrupting clause not being considered part of the interrupted one.

6 Conclusion

In our paper, we described the most important steps of the creation of a morphological annotation framework for the analysis of Old and Middle Hungarian extant texts consisting of a morphological analyzer, an automatic disambiguation tool and an intuitive web-based manual disambiguation tool. Certain problems arising during this process were discussed together with their solution. We also presented our corpus manager, which serves both as a structured corpus query tool and as a correction tool.

The morphological analyzer is used for the annotation of the constantly growing Old and Middle Hungarian corpora. Part of these corpora are already searchable by the public. The Old Hun-

Figure 2: The query interface

garian Corpus is available at http://rmk.nytud.hu, while the analyzed part of the Historical Corpus of Informal Language Use can be searched at http://tmk.nytud.hu.

Acknowledgments

Research reported in this paper was supported by the research project grants OTKA NK78074 and OTKA 81189. In addition, we gratefully acknowledge support by the grants TÁMOP-4.2.1./B-11/2/KMR-2011-002 and TÁMOP-4.2.2./B-10/1-2010-0014. We would also like to thank anonymous reviewers of the paper for their helpful comments and suggestions.

References

Péter Halácsy, András Kornai, and Csaba Oravecz. 2007. HunPos: an open source trigram tagger. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 209–212, Stroudsburg, PA, USA. Association for Computational Linguistics.

Mans Hulden and Jerid Francom. 2012. Boosting statistical tagger accuracy with simple rule-based grammars. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Mehmet Uğur Doğan, Bente Maegaard, Joseph Mariani, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey. European Language Resources Association (ELRA).

Ray Jackendoff. 1977. *X-bar-Syntax: A Study of Phrase Structure*. Linguistic Inquiry Monograph 2. MIT Press, Cambridge, MA.

László Jakab and Antal Kiss. 1994. *A Guarÿ-kódex ábécérendes adattára*. Számítógépes nyelvtörténeti adattár. Debreceni Egyetem, Debrecen.

László Jakab and Antal Kiss. 1997. *Az Apor-kódex ábécérendes adattára*. Számítógépes nyelvtörténeti adattár. Debreceni Egyetem, Debrecen.

László Jakab and Antal Kiss. 2001. *A Festetics-kódex ábécérendes adattára*. Számítógépes nyelvtörténeti adattár. Debreceni Egyetem, Debrecen.

László Jakab. 2002. *A Jókai-kódex mint nyelvi emlék: szótárszerü feldolgozásban*. Számítógépes Nyelvtörténeti Adattár. Debreceni Egyetem, Debrecen.

Attila Novák. 2003. Milyen a jó Humor? [What is good Humor like?]. In *I. Magyar Számítógépes Nyelvészeti Konferencia*, pages 138–144, Szeged. SZTE.

György Orosz and Attila Novák. 2012. PurePos – an open source morphological disambiguator. In *Proceedings of the 9th International Workshop on Natural Language Processing and Cognitive Science.*, Wroclaw, Poland.

Ulrik Petersen. 2004. Emdros — a text database engine for analyzed or annotated text. In *In: Proceedings of COLING 2004. (2004) 1190–1193*.

Gábor Prószéky and Balázs Kis. 1999. A unification-based approach to morpho-syntactic parsing of agglutinative and other (highly) inflectional languages. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, ACL '99, pages 261–268, Stroudsburg, PA, USA. Association for Computational Linguistics.

Gábor Prószéky and Attila Novák. 2005. Computational Morphologies for Small Uralic Languages. In *Inquiries into Words, Constraints and Contexts.*, pages 150–157, Stanford, California.