The Workshop Programme

9:30- 9:35	Introduction
9:35-10:00	Penny Labropoulou, Harris Papageorgiou, Byron Georgantopoulos, Dimitra Tsagogeorga, Iason Demiros and Vassilios Antonopoulos: "*Integrating Language Technology in a web-enabled Cultural Heritage system*"
10:00-10:30	Oto Vale, Arnaldo Candido Junior, Marcelo Muniz, Clarissa Bengtson, Lívia Cucatto, Gladis Almeida, Abner Batista, Maria Cristina Parreira da Silva, Maria Tereza Biderman and Sandra Aluísio: "*Building a large dictionary of abbreviations for named entity recognition in Portuguese historical corpora*"
11:00-11:30	Eiríkur Rögnvaldsson and Sigrún Helgadóttir: "*Morphological tagging of Old Norse texts and its use in studying syntactic variation and change*"
11:30-12:00	Lars Borin and Markus Forsberg: "*Something Old, Something New: A Computational Morphological Description of Old Swedish*"
12:00-12:30	Dag Trygve Truslew Haug and Marius Larsen Jøhndal: "*Creating a Parallel Treebank of the Old Indo-European Bible Translations*"
12:30-12:55	Elena Grishina: "*Non-Standard Russian in Russian National Corpus (RNC)*"
12:55-13:20	Dana Dannélls: "*Generating tailored texts for museum exhibits*"
14:30-15:00	David Bamman and Gregory Crane: "*The Logic and Discovery of Textual Allusion*"
15:00-15:30	Dimitrios Vogiatzis, Galanis Dimitrios, Vangelis Karkaletsis and Ion Androutsopoulos: "*A Conversant Robotic Guide to Art Collections*"
15:30-16:00	René Witte, Thomas Gitzinger, Thomas Kappler and Ralf Krestel: "*A Semantic Wiki Approach to Cultural Heritage Data Management*"
16:30-17:15	Christoph Ringlstetter: "*Error Correction in OCR-ed Data*"- Invited Talk
17:15-17:30	Closing

Workshop Organisers

Caroline Sporleder (Co-Chair), Saarland University, Germany
Kiril Ribarov (Co-Chair), Charles University, Czech Republic
Antal van den Bosch, Tilburg University, The Netherlands
Milena P. Dobreva, HATII, University of Glasgow, Scotland
Matthew James Driscoll, Kobenhavns Universitet, Denmark
Claire Grover, University of Edinburgh, Scotland
Piroska Lendvai, Tilburg University, The Netherlands
Anke Luedeling, Humboldt-Universitat, Germany
Marco Passarotti, Universita Cattolica del Sacro Cuore, Italy

Workshop Programme Committee

Ion Androutsopoulos, Athens University of Economics and Business, Greece
Timothy Baldwin, University of Melbourne, Australia
David Bamman, Perseus, USA
David Birnbaum, University of Pittsburgh, USA
Antal van den Bosch, Tilburg University, The Netherlands
Andrea Bozzi, ILC-CNR, Pisa, Italy
Kate Byrne, University of Edinburgh, Scotland
Paul Clough, Sheffield University, UK
Greg Crane, Perseus, USA
Milena P. Dobreva, HATII, University of Glasgow, Scotland
Mick O'Donnell, Universidad Autonoma de Madrid, Spain
Matthew James Driscoll, Kobenhavns Universitet, Denmark
Franciska de Jong, University of Twente, The Netherlands
Claire Grover, University of Edinburgh, Scotland
Ben Hachey, University of Edinburgh, Scotland
Djoerd Hiemstra, University of Twente, The Netherlands
Dolores Iorizzo, Imperial College London, UK
Christer Johansson, University of Bergen, Norway
Piroska Lendvai, Tilburg University, The Netherlands
Anke Luedeling, Humboldt-Universitat, Germany
Roland Meyer, University of Regensburg, Germany
Maria Milosavljevic, University of Edinburgh, Scotland
Marie-Francine Moens, Katholieke Universiteit Leuven, Belgium
Marco Passarotti, Universita Cattolica del Sacro Cuore, Italy
Martin Reynaert, Tilburg University, The Netherlands
Kiril Ribarov, Charles University, Czech Republic
Maarten de Rijke, University of Amsterdam, The Netherlands
Peter Robinson, ITSEE, UK
Maria Simi, University of Pisa, Italy
Caroline Sporleder, Saarland University, Germany

Table of Contents

Author Index

The Logic and Discovery of Textual Allusion

David Bamman
The Perseus Project
Tufts University
Medford, MA
david.bamman@tufts.edu

Gregory Crane
The Perseus Project
Tufts University
Medford, MA
gregory.crane@tufts.edu

Abstract

We describe here a method for discovering imitative textual allusions in a large collection of Classical Latin poetry. In translating the logic of literary allusion into computational terms, we include not only traditional IR variables such as token similarity and n-grams, but also incorporate a comparison of syntactic structure as well. This provides a more robust search method for Classical languages since it accomodates their relatively free word order and rich inflection, and has the potential to improve fuzzy string searching in other languages as well.

1 Introduction

> Five score years ago, a great American, in whose symbolic shadow we stand today, signed the Emancipation Proclamation ...

Thus begins Martin Luther King Jr.'s "I Have a Dream" speech of 1963. While the actual text of the Gettysburg Address is not directly quoted here, it is elicited by means of an allusion: King's audience would immediately have recognized the parallels between his first four words and the "Four score and seven years ago" that began Lincoln's own speech. By opening with this phrase, King is aligning Lincoln's invocation of human equality with "the greatest demonstration for freedom in the history of our nation" for which he was then speaking.

While the term "allusion" is commonly applied to any reference to a person, place, or thing already known to the reader, we are using it here in the specific context of an *imitative textual allusion* – a passage in one text that refers to a passage in another. When Willy Loman calls each of his sons an "Adonis" in *Death of a Salesman*, there is no doubt that this is an allusion to a Classical myth, but it does not point to a definable referent in the record of written humanity (as King's allusion refers specifically to the first six words of the Gettysburg Address).

The discovery of these allusions is a crucial process for the analysis of texts. As others have pointed out,[1] allusions have two main functions: to express similarity between two passages, so that the latter can be interpreted in light of the former; and to simultaneously express their dissimilarity as well, in that the tradition they recall is revised.[2] Allusions of this specific variety are perhaps most widely known as a trope of modernist authors such as Eliot and Joyce, but they are common in the Classical world as well – most strongly in the Greek poetry of the Hellenistic era, in the Roman poetry of the republic and early empire and in New Testament texts (which allude to prophecies recorded in the Old Testament). Given the long history of Latin literature, we must also keep in the mind a text's *Nachleben* – how it has been received and appropriated by the generations that follow it.[3]

Uncovering allusions of this sort has long been the task of textual commentators, but we present

[1] For an overview of the function and interpretive significance of allusions, see Thomas (1986).

[2] Cf. Bloom (1973).

[3] Cicero, for example, was widely admired by Renaissance humanists after Petrarch and provided a model for textual imitation. Cf. Kristeller (1979).

a method here to automatically discover them in texts. Our approach has many similarities with research on text reuse (Clough et al., 2002), paraphrase and duplicate detection (Dolan et al., 2004), and locating textual reference (Takeda et al., 2003; Lee, 2007), but while these methods generally focus on string comparison and document structure, we include variables for considering the abstract structure of a sentence as well, as represented by its syntax. This enables a more robust search method since it is not restricted by word order or inflection. Our test corpus is a collection of Latin poetry, but the methods we describe are language independent.

2 Types of Textual Allusion

While others have categorized textual allusion into a number of types dependent on their function (e.g., Thomas (1986) distinguishes between "casual reference," "single reference," "self-reference," etc.), we are concerned only with a practical distinction in terms of the ease of locating them: an allusion is either direct (equivalent to a quotation) or indirect.

2.1 Direct reference

The most explicit and unambiguous type of allusion is direct reference in the form of a verbatim quotation. We see this form of allusion most often in the long afterlife of a text, as for instance in the reception of this line from Ovid's *Amores*.

(1) At si, quem mavis, Cephalum conplexa teneres / Clamares: **lente currite, noctis equi!** (Am. 1.13)[4]

While Ovid's line comes from the mouth of the mythic Aurora (dawn) pleading with her chariot to pull her more slowly across the sky to give her more time with her lover before returning to her husband, Christopher Marlowe sixteen centuries later appropriates it for Faust, who voices it in the final minutes before midnight in a plea to prolong his life.

(2) Stand still, you ever-moving spheres of heaven, That time may cease, and midnight never come: Fair Nature's eye, rise, rise again and make Perpetual day; or let this hour be but A year, a month, a week, a natural day, That Faustus may repent and save his soul! **O lente, lente, currite noctis equi!** (Act V, Scene 2)

And again, four centuries later, Vladimir Nabokov appropriates it for *Lolita* as his protagonist is chased along a highway.

(3) We were many times weaker than his splendid, lacquered machine, so that I did not even attempt to outspeed him. **O lente currite noctis equi!** O softly run, nightmares! (Nabokov 219)

Following Irwin (2001), we can distinguish an allusion from a mere quotation in the level of context required to understand it. A quotation is self-contained; an allusion calls forth the original context in which it's found. Direct allusions like these are easier to find than their adapted counterparts (it is essentially a simple string search) but they reside on the same continuum as the others.

2.2 Indirect reference

Most of what we would consider allusions involve some transformation of the referent text. An example of this can be found in the first line of the first poem of Ovid's *Amores*, an imitation (and revision) of the first line of Vergil's *Aeneid*.

(4) Arma gravi numero violentaque bella parabam / Edere (Am. 1.1-2)[5]

(5) Arma virumque cano (Aen. 1.1)[6]

Vergil's *Aeneid* is an epic poem focussed on the figure of Aeneas (an ancestor of the Romans), written in dactylic hexameter, the same "heavy" meter as Homer's epics the *Iliad* and *Odyssey*. Ovid, in contrast, is a love poet, and elicits Vergil's famous opening to motivate his genre (the line continues with Cupid stealing one of the line's metrical feet, leaving it an elegaic couplet, a common meter of Roman love poetry).

This type of common allusion clearly presents much more difficulty in being found: any variety of simple string search (either exact or fuzzy) will not be successful, since only two word forms – *arma* ("arms") and the enclitic *-que* ("and") – are common to both strings.

[4]"But if you held Cephalus in your arms, whom you prefer, you would shout 'run slowly, horses of the night!'"

[5]"I was planning to write about arms and violent wars in a heavy meter."

[6]"I sing of arms and the man."

3 The Logic of Allusion

Clearly we need to add new methods for establishing similarity between two lines beyond simple string matches. This begs the question, however, of how it is we know (as humans) that one passage in a text is an allusion to another. The ultimate criterion of course involves higher-order reason (an allusion must make interpretive sense) but we can identify a number of explicit surface variables that give notice to the presence of an allusion in the first place.

Identical words. A quotation is an allusion where the edit distance between two strings is effectively 0: i.e., all word forms in one span of text are identical with those in another. In sentences 4 and 5, only *arma* and *que* are the same, but they nevertheless provide a necessary anchor for establishing a link between the two passages. While *arma* in both examples here in is the same grammatical case (accusative), many times an alternation occurs as well (e.g., transforming a word from the accusative to the nominative case). We can therefore define "identical" to mean both token identity (*arma* = *arma*) and root form (lemma) identity (*ego* = *me*).

Word order. Syntax in projective languages like English is strongly tied to word order (an adjective, for example, generally modifies the noun that immediately follows it), but for non-configurational languages like Latin and Greek, word order is much more free, especially in the genre of poetry in which allusion is so common. For this reason we treat syntax as a separate variable (see below) and isolate word order as its own phenomenon. For our example above, word order is another cue to the presence of an allusion since both lines begin with the same word, *arma*.

Syntactic similarity. When considering syntax we begin to see the strongest parallels between the two passages. In both sentences, *arma* is involved in coordination as a direct object of a verb. While the head verbs differ (*edere* vs. *cano*) as does the other object involved in coordination (*bella* vs. *virum*), the two structures are syntactically identical.

Figures 1 and 2 present a syntactic tree of each sentence under the formalism of dependency grammar.[7] In both of these trees, the two direct objects of the verbs are headed by the coordinator *que* via the syntactic relation OBJ_CO, while the coordinator is headed by the verb via the relation COORD. While the words themselves vary, the structure is the same.

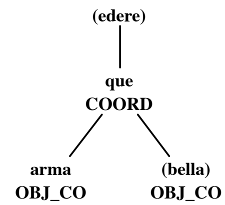


Figure 1: Dependency tree of *arma -que bella edere* ("to write about arms and wars").

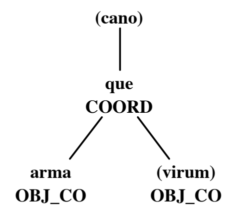


Figure 2: Dependency tree of *arma virumque cano* ("I sing of arms and the man").

Metrical and phonetic similarity The first lines of both of these poems are both written in dactylic hexameter, but the similarity between the two is much closer than that, since the first seven syllables of both lines are metrically identical – two dactyls followed by a stressed syllable and caesura. Additionally, the final long syllable before the caesura is the same in both sentences ("o"), eliciting a further phonetic similarity as well.

(6) Ārmă grăvī nŭmĕrō ‖ ...

(7) Ārmă vĭrūmquĕ cănō ‖ ...

Semantic similarity We can also note that on a semantic level, both of these passages are "about" similar things, at least in this first line (before the arrival of Cupid in Ovid) – in both lines, the author is communicating (via writing or singing) about war (*violenta bella*) and the instruments of war (*arma*).

[7] This is the structural representation of syntax as found in the Latin Dependency Treebank (Bamman and Crane, 2007) and the Prague Dependency Treebank of Czech (Hajič, 1998).

With semantic similarity we can also group another very important variable – cross-language semantic information in the form of translation equivalents. This is extremely important given the reception of these texts across cultures and distant eras. Classical Roman poets themselves are especially fond of borrowing from Homer and Hellenistic poets, but we see the same phenomenon in English as well – one only need to look at Milton's use of the *Aeneid* in *Paradise Lost* to see the level of appropriation, which in its simplest form approaches exact translations of fixed phrases, such as sentences 8 and 9 below, and in its more complex form also involves the host of other variables outlined above.

(8) The Moons resplendent Globe (PL 4.723)

(9) Lucentemque globum lunae (Aen. 6.725)

These five categories represent broad classes of similarity, but of course we must expect others on an ad hoc basis as well – in sentences 4 and 5 from above, we have the additional similarity that both passages come from the privileged first lines of both poems, suggesting a larger structural similarity. While these variables do not illuminate the interpretive significance of an allusion (we can leave that contentious task to critics), they do provide a means by which to discover them in the first place.

4 Discovering Allusions

Our task in automatically discovering allusions is to transform the variables listed above into ones that are computationally tractable. We need to be able to define the precise degree to which two passages are similar in order to quantitatively compare which pairs of passages are more similar to each other than others.

Information retrieval has produced a number of metrics for judging the similarity of two documents. The most widely used of these generally assign a relevance score based on some variation of tf/idf weighting: two documents are similar if they both contain words that occur less frequently in the collection of texts at large. The more uncommon words they share, the greater their similarity.

To establish the similarity between two sentences, we can use the cosine measure as a means of judging their vector similarity.

$$cos(\vec{s},\vec{t}) = \frac{\sum_{i=i}^{n} s_i t_i}{\sqrt{\sum_{i=i}^{n} {s_i}^2}\sqrt{\sum_{i=i}^{n} {t_i}^2}}$$

Here s_i is the tf/idf score for the term i in the source sentence s and t_i is the tf/idf score for that same term in the target comparsion sentence t. We measure each tf/idf by the following formula.

$$(1+\log(tf_{i,j}))\log\frac{N}{df_i}$$

Here tf_i = the count of term i in sentence j, N = the total number of sentences in the collection, and df_i = the number of sentences in that collection that contain the term i.

The closer this cosine is to 1, the more similar two sentences are. We will use this general framework to inform all of the following variables: the difference between them will be in what exactly constitutes a "term."

4.1 Identical words

Given Latin's rich inflection, we will define two variables for establishing identity between words, token similarity and lemma similarity.

Token similarity. Here we define *term* to be the overt (i.e., inflected) form of the word. This measure reflects a typical search engine query in that it compares two documents (here, sentences) based on how closely their words match each other. More common words between the two documents leads to a greater level of similarity.

Lemma similarity. Here we define *term* to be the uninflected lemma from which the token is derived. In this variable, *omnia vincit amor* ("love conquers all") is identical to *omnia vincuntur amore* ("all things are conquered by love") since the lemmas underlying both are *omnis1 vinco1 amor1*. A measure for lemma similarity addresses the fact that many allusions are not simple quotations – the words that constitute the reference are not bound to their original case as they were used in the target text, but are often given a different grammatical role in the allusion.

4.2 Word order

We can measure the explicit order of words (as distinct from their abstract syntax) with the use of n-grams – specifically bigrams and trigrams, which measure how frequently two or three words appear in linear order. Using the beginning and end of sentences as distinct words of their own (in order to measure when a word begins or ends a line), the phrase *omnia vincit amor* has 4 bigrams (*[start] omnia, omnia vincit, vincit amor, and amor [end]*) and three trigrams: (*[start] omnia vincit, omnia vincit amor, and vincit amor [end]*).

This will let us capture, for instance, that *arma virumque cano* is similar to *arma gravi numero* in that both begin with the bigram *[start] arma*. We can again account for Latin's rich inflection with the use of lemma bigrams and trigrams in addition to tokens. This results in four total word order variables: token bigram, token trigram, lemma bigram and lemma trigram.

4.3 Syntax

The two variables outlined so far form the backbone of information retrieval applications. By considering syntax, we can get beyond simple string resemblance metrics and begin to consider similarities in abstract structure as well.

With syntactic relations, we can specify the true syntactic distance between two phrases (as distinct from simple word order). Several measures of syntactic distance have recently been proposed: Spruit (2006) presents a method for classifying dialects based on previously human-curated variables (e.g., the presence of personal vs. reflexive pronouns etc.); Nerbonne and Wiersma (2006) approximate syntactic distance using part of speech trigrams, which works well for classifying different language groups (adults vs. child) in English (a language with strict word order); and Sanders (2007) measures distance using Sampson's (2000) leaf-ancestor paths, in which each word in a sentence is identified as its path from itself to the top of the syntactic tree (e.g., in a phrase structure grammar: "The"-Det-NP-S/"dog"-N-NP-S/"barks"-V-VP-S). Given Latin's non-projectivity, we have adopted this third measure and augmented it along three dimensions to make it suitable for a dependency grammar.

Figure 3 presents a syntactic tree annotated under a dependency-based grammar. Since dependency grammars do not have intermediate phrase structures such as NP or VP, we take our basic syntactic structure to be a child-parent relationship between words

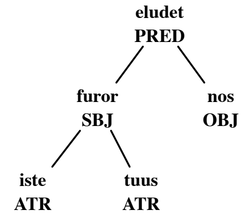


Figure 3: Dependency tree of *furor iste tuus nos eludet* ("that madness of yours will play with us"), Cicero, *In Catilinam 1.1*.

themselves. How we represent those words constitutes the first dimension:

- part of speech: adj:noun:verb
- token: iste:furor:eludet
- lemma: iste1:furor1:eludo1

The second dimension is the length of the path. While Sanders' metric identifies each word by its full path from itself to the top of the tree, we can use a number of intermediary paths to assert similarity as well. Since a full path from the word itself to the top of the tree is very unlikely to be repeated across sentences, we approximate it by considering only paths of lengths 2 and 3 (bigrams and trigrams): a path of length 2 would for instance be comprised of "adj:noun"/"iste:furor"/etc. while a path of length 3 would be comprised of "adj:noun:verb" (as above).

The third dimension is the presence or absence of the syntactic label. Dependency grammars differ from phrase structure grammars by providing an explicit relation between words (where phrase structure grammars often imply it by position – a subject, for example, is an NP that c-commands a VP). Using the syntactic labels specified in the Latin Dependency Treebank (Bamman and Crane, 2007), a labeled path would be comprised of "iste:ATR:furor:SBJ:eludet" for token trigrams, while an unlabeled path would leave this information out (as above).

These three dimensions provide 12 distinct syntactic variables for each word in a sentence, ranging from least explicit (unlabeled part of speech bigrams ["adj:noun"]) to most (labeled token trigrams ["iste:ATR:furor:SBJ:eludet"]). The most explicit variables will have the lowest inverse document frequencies and will therefore be the most informative for judging similarity if present, while the least explicit variables will still provide a back-off means to provide some similarity in the event of a more explicit non-match.

4.4 Metrical/phonetic similarity and semantic similarity

While we do not implement metrical/phonetic or semantic similarity measures in what follows, we can address the means by which we could do so in the future.

We can measure metrical and phonetic similarity in a manner similar to the term frequencies used in the variables above, by comparing the meter of two passages (this of course requires metrically annotated texts). Meter in this case can be seen as a language with two letters, *long*(_) and *short*(◡), and we can judge the similarity between two meters as a simple string comparison of that representation.

We can judge the semantic similarity between two words using either monolingual clustering techniques such as latent semantic analysis (which notes, for example, that an *apple* is semantically close to an *orange* since both appear often with words such as *eat* and *tree*) (Deerwester et al., 1990), or by cross-language translation equivalents (such as those induced in the course of parallel text alignment (Och and Ney, 2003)), which notes the frequency with which a word in one language (such as *oratio* in Latin) is translated by different terms (e.g., *speech* vs. *prayer*).

5 Evaluation

We evaluated the first three variable classes above (word identity, word order and syntax) on a collection of 14 texts from 5 Latin poets – Catullus (*Carmina*), Ovid (*Metamorphoses*, *Amores*, *Epistulae*, *Medicamina Faciei Femineae*, *Ars Amatoria*, *Remedia Amoris*), Vergil (*Aeneid*, *Eclogues*, *Georgics*), Propertius (*Elegies* I), and Horace (*Carmina*, *Satyrarum libri*, *De Arte Poetica liber*).

While the word identity and word order variables can be calculated on unstructured text, we need syntactically parsed data in order to measure syntactic similarity. To create this, we trained McDonald et al.'s dependency parser (McDonald et al., 2005) on the manually curated data in the Latin Dependency Treebank and used it to parse all of the texts in our collection.[8]

Author	Words	Sentences
Ovid	141,091	10,459
Vergil	97,495	6,553
Horace	35,136	2,345
Catullus	14,793	903
Propertius	4,867	366
	293,382	20,626

Table 1: Composition of the test corpus by author.

After finding the most similar sentences for each of the 20,626 sentences in our collection, we filtered the results to require a lower limit for sentence length in order to find meaningful pairs (short sentences such as *Quid est?* can be found across many authors and are not allusions even though they match exactly) and to avoid sentence pairs that are both found in the same immediate context (e.g., Catullus' poem 61, where a chorus of the same 7 words is exactly repeated 11 times throughout the poem).[9]

The results are encouraging: while a detailed quantitative evaluation must await the creation of a test corpus of canonical allusions, we can at least now provide a list of the closest matches for all sentences in our collection. For any given sentence, further research will of course be necessary to discern whether it represents a real allusion, but the highest scoring pairs in our experiment tend to be strong examples. Sentences 10 and 11, for instance, present one such pair from Ovid and Vergil with a similarity score of .173.

(10) **Innumeras urbes atque aurea tecta** videbis, / Quaeque suos dicas templa decere deos (Ov. Ep. 16)[10]

(11) Iam subeunt **Triviae lucos atque aurea tecta** (Verg., Aen. 6.13)[11]

[8] In a tenfold test on the treebank data itself, we measured the parser's unlabeled accuracy to be 64.99% and its labeled accuracy to be 54.34% (Bamman and Crane, 2008).

[9] *o Hymen Hymenaee io, o Hymen Hymenaee.*

[10] "You will see innumerable cities and golden roofs, and temples that you would say are fitting to their gods."

[11] "Already they enter Trivia's groves and golden roofs."

Sentences 12 and 13 likewise present a pair from Ovid and Catullus with a score of .141.

(12) **nulli illum iuvenes, nullae tetigere puellae** (Ov., Met. 3.353)[12]

(13) idem cum tenui carptus defloruit ungui / **nulli illum pueri, nullae optavere puellae** (Cat., Carm. 62)[13]

The strongest matches, however, came within authors, who often sample their own work in other contexts. This occurs most often by far in Vergil, where the re-appropriation involves exactly repeating complete sentences (9 instances), exactly repeating substantial sentence fragments (23 instances),[14] and more significant modifications.

Additionally, since our weights are based on preset variables, the process by which we come to the most similar match is transparent. Table 2 presents the term weights for several of the highest and lowest variables at play in establishing the similarity between sentences 12 and 13 above.

This table presents the clear importance of using syntax as a method for establishing the similarity between sentences – three of top four variables that have linked these two sentences to each other involve syntax (e.g., *nullae* depends on *puellae* in both sentences as an attribute).[15]

Our search for *loci similes* to our original allusion from above – Ovid's *Arma gravi numero violentaque bella parabam* – illustrates well the importance of bringing a variety of information to the search. The closest sentences to Ovid's original line all bear some similarity to it on both a lexical and syntactic level (as sentences 1 and 2 demonstrate below). Our target sentence of Vergil (*Arma virumque cano ...*), however, only shows up in 11th place on the list.

[12]"No youths, no girls touched him."

[13]"This same one withered when plucked by a slender nail; no boys, no girls hope for it."

[14]Here "substantial" means at least seven consecutive identical words.

[15]Note that the labeled syntactic bigram nullae:puella:ATR has the same tf/idf score as the unlabeled nullae:puellae since all instances of *nullae* depending on *puella* in our automatically parsed corpus do so via the relation ATR.

Variable	tf/idf
nullae:puellae:ATR	9.24
nullae:puellae	9.24
nulli/illum	9.24
p:SBJ_EXD_OBJ_CO:u:COORD:v	9.24
,/nullae	8.84
nullus1:puella1	8.55
nullus1:puella1:ATR	8.32
nullae	8.55
...	...
nulli	6.30
puellae	5.55
illum	5.34
a:n:ATR:v:SBJ	1.67

Table 2: Sample of variable contribution. Components separated by a colon represent syntactic relations; those with slashes are n-grams.

1. Arma procul currusque virum miratur inanes (.059) (Aen. 6.651)[16]

2. Quid tibi de turba narrem numeroque virorum (.042) (Ov., Ep. 16.183)[17]

11. Arma virumque cano, Troiae qui primus ab oris Italiam, fato profugus, Laviniaque venit litora, multum ille et terris iactatus et alto vi superum saevae memorem Iunonis ob iram (.025) (Aen. 1.1)[18]

This is understandable given the variables we have implemented – the first three sentences do indeed bear a closer similarity to the original without being diluted by extra words (since our cosine value normalizes for sentence length). We hope in the future to be able to include other important variables (such as metrical similarity) as well.

[16]"At a distance he marvels at the arms and the shadowy chariots of men."

[17]"What could I tell you of the crowd and the number of men?"

[18]"I sing of arms and the man, who first from the borders of Troy, exiled by fate, came to the Lavinian shores – much was he thrown about on land and sea by force of the gods on account of the mindful anger of cruel Juno."

6 Conclusion

Allusion is by nature an oblique art; its very essence – referring to something that the audience already knows – gives it the opportunity to be highly economical in its expression. Since even a single word or structure can refer to another text, we must leverage as many different varieties of information as we can in order to discover them, from lexical information to syntax and beyond. We have defined five different variable classes that contribute to the surface realization of allusion, and have implemented a system that includes three of those five. By considering the abstract structure of sentences, we are able to effectively search Latin without being encumbered by its flexible word order and rich inflectional morphology, which allows similar sentences to be expressed in a variety of ways. While we have designed this method for a collection of Classical texts, we expect that it can also be used to improve the robustness of searches in any language.

7 Acknowledgments

Grants from the Digital Library Initiative Phrase 2 (IIS-9817484), the National Science Foundation (BCS-0616521) and the Andrew W. Mellon Foundation (#40700635) provided support for this work. Thanks are due also to Meg Luthin and Molly Miller for their invaluable research assistance.

References

David Bamman and Gregory Crane. 2007. The Latin Dependency Treebank in a cultural heritage digital library. In *Proceedings of the Workshop on Language Technology for Cultural Heritage Data (LaTeCH 2007)*, pages 33–40, Prague.

David Bamman and Gregory Crane. 2008. Building a dynamic lexicon from a digital library. In *Proceedings of the 8th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL 2008)*.

Harold Bloom. 1973. *The Anxiety of Influence; A Theory of Poetry*. Oxford University Press, New York.

Paul Clough, Robert J. Gaizauskas, Scott S. L. Piao, and Yorick Wilks. 2002. METER: Measuring text reuse. In *Proceedings of the ACL*, pages 152–159.

Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407.

Bill Dolan, Chris Quirk, and Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: exploiting massively parallel news sources. In *COLING '04*. Association for Computational Linguistics.

Jan Hajič. 1998. Building a syntactically annotated corpus: The Prague Dependency Treebank. In Eva Hajičová, editor, *Issues of Valency and Meaning. Studies in Honor of Jarmila Panevová*, pages 12–19. Prague Karolinum, Charles University Press.

William Irwin. 2001. What is an allusion? *The Journal of Aesthetics and Art Criticism*, 59:287–297.

Paul Oskar Kristeller. 1979. *Renaissance Thought and Its Sources*. Columbia University Press, New York.

John Lee. 2007. A computational model of text reuse in ancient literary texts. In *Proceedings of the ACL*, pages 472–479, Prague, Czech Republic.

Ryan McDonald, Fernando Pereira, Kiril Ribarov, and Jan Hajič. 2005. Non-projective dependency parsing using spanning tree algorithms. In *Proceedings of the HLT-EMNLP*, pages 523–530.

Vladimir Nabokov. 1991. *The Annotated Lolita. Edited, with preface, introduction, and notes by Alfred Appel, Jr.* Vintage Books, New York.

John Nerbonne and Wybo Wiersma. 2006. A measure of aggregate syntactic distance. In *Proceedings of the Workshop on Linguistic Distances*, pages 82–90.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Geoffrey Sampson. 2000. A proposal for improving the measurement of parse accuracy. *International Journal of Corpus Linguistics*, 5(1):53–68.

Nathan C. Sanders. 2007. Measuring syntactic difference in British English. In *Proceedings of the ACL2007 Student Research Workshop*, pages 1–6.

Marco Rene Spruit. 2006. Measuring syntactic variation in Dutch dialects. *Literary and Linguistic Computing*, 21:493–506.

Masayuki Takeda, Tomoko Fukuda, Ichiro Nanri, Mayumi Yamasaki, and Koichi Tamari. 2003. Discovering instances of poetic allusion from anthologies of classical Japanese poems. *Theoretical Computer Science*, 292(2):497–524.

R. F. Thomas. 1986. Vergil's *Georgics* and the art of reference. *Harvard Studies in Classical Philology*, 90:171–98.

Something Old, Something New: A Computational Morphological Description of Old Swedish

Lars Borin, Markus Forsberg

Språkbanken, Department of Swedish Language
University of Gothenburg
Box 200, SE-405 30 Gothenburg, Sweden
lars.borin@svenska.gu.se, markus.forsberg@gu.se

Abstract

We present a computational morphological description of Old Swedish implemented in Functional Morphology. The objective of the work is concrete – connecting word forms in real text to entries in electronic dictionaries, for use in an online reading aid for students learning Old Swedish. The challenge we face is to find an appropriate model of Old Swedish able to deal with the rich linguistic variation in real texts, so that word forms appearing in real texts can be connected to the idealized citation forms of the dictionaries.

1. Background

1.1. Motivation

Languages with a long written tradition have accumulated over the centuries a rich cultural heritage in the form of texts from different periods in the history of the language community. In these texts, we find information on many aspects of the origins and history of our culture.

Since languages change over time, older texts can be difficult or impossible to understand without special training. Indeed, the oldest extant texts in many European languages must in fact be translated in order to be accessible to a modern reader. In Sweden, the training of professionals who can keep this aspect of our cultural heritage alive by conveying the content of, e.g., the oldest Swedish legal and religious texts (13th–15th c.) to a modern audience, is the province of the academic discipline of Swedish Language and Linguistics at Swedish universities.

Consequently, an important component of all Swedish Language university curricula in Sweden is the study of the older stages of the language. For obvious reasons, any form of any language older than a bit over a century will be accessible only in writing. Hence, our main source of historical Swedish language data is in the form of old texts, and courses in the history of the Swedish language, in comparative Scandinavian and in Old Swedish, all contain – in time-honored fashion – a module where students are required to read a certain amount of texts. In equally time-honored fashion, students have two main tools for working with Old Swedish texts: dictionaries and grammar books.

There are excellent editions of Old Swedish texts available in book form, as well as good grammatical descriptions (Noreen, 1904; Wessén, 1969; Wessén, 1971; Wessén, 1965; Pettersson, 2005) and reference dictionaries (section 1.4). Both text collections and reference works such as dictionaries and grammars tend to be out-of-print works, which does not in any way detract from their usefulness or their accuracy, but which does present some practical problems. Students are confined to using reference copies held in departmental libraries, but often departments see themselves forced to restrict access because of the excessive wear this causes.

However, historical texts are increasingly available also in digital form on the internet. Two sites with extensive collections of Old Swedish texts are *Språkbanken* (the Swedish Language Bank; <http://spraakbanken.gu.se>) at the University of Gothenburg and *Fornsvensk textbank* (the Old Swedish Text Bank; <http://www.nordlund.lu.se/Fornsvenska/Fsv%20Folder/index.html>) at Lund University. The standard reference dictionaries of Old Swedish have also been digitized (by Språkbanken) and are available for word lookup via a web interface <http://spraakbanken.gu.se/fsvldb/> and in full-text for research purposes by special agreement.

1.2. Toward a Solution

Our project aims to aid online reading of Old Swedish texts (cf. Nerbonne and Smit (1996)) by providing access to automatic morphological analysis and linking that to available lexical resources.

For our goals, we need morphological analysis, i.e. an analysis module which returns, crucially, a lemma as part of its result, since the lemma is necessary in order to access the online dictionaries. This makes the methodology of our project different from that of, e.g., Rayson et al. (2007), where a POS tagger trained on Modern English was adapted to deal with the Early Modern English of Shakespeare's time. It is doubtful whether this methodology would have helped much if the target language had been Old English (Anglo-Saxon) instead, because of the much greater linguistic distance between the two varieties, which are most appropriately seen as two entirely different, but related languages. What we have in our case are two historical language stages as far apart linguistically as Modern English and Anglo-Saxon (see section 1.3).

POS tagging is also a different kind of analysis from that provided by a morphological analyzer. The former will provide only one analysis of each text token, the most probable one in the given context, whereas the latter will provide all readings licensed by the lexicon and grammar, but will not take context into account. POS tagging will *always* provide an analysis, however, whereas a morphological analyzer may fail to do so if a lemma is lacking in its lexicon or a form is not allowed by its grammar. Thus, POS tagging and morphological analysis are complementary; POS tagging is often used to select among competing morphological analyses, the combination of the two methods thus providing both disambiguation and lemmatization. Our goal in the project described here is to provide (undisambiguated) morphological analysis including lemmatization of text words, in order to link them to the corresponding dictionary entries. A natural future extension of the work would be a disambiguation module, either a POS tagger or, e.g., a constraint grammar (Karlsson et al., 1995), but we believe that the differences between Old and Modern Swedish are great enough to preclude an easy adaptation of a Modern Swedish POS tagger or constraint grammar.

We are partway through the project. The morphological analysis module is completed for all regular paradigms and some others (section 2), and a small number of lexical entries have been provided with inflectional information (section 2.3). This is primarily what we report on in this paper. We are now in the process of providing all lexical entries with inflectional information in the form of paradigm identifiers (section 2). This task is not completely trivial, since the extant texts do not always allow us to determine to which inflectional class a particular lexical entry should belong.

However, the main challenge still remaining is the issue of how to deal in as principled a way as possible with the considerable linguistic variation present in the texts that we are working with, and which presents us with a different situation compared to working with modern texts (sections 1.3 and 3).[1]

1.3. Down the Foggy Ruins of Time: Orthographic Inconsistency, Linguistic Variation and Language Change in Old Swedish Texts

The texts are from the so-called Old Swedish period (1225–1526), conventionally subdivided into Classical Old Swedish (1225–1374) and Late Old Swedish (1375–1526), covering a time span of 300 years. The language of the extant Old Swedish texts exhibits considerable variation, for at least the following reasons:

1. The orthography was not standardized in the way that we expect of modern literary languages;
2. the language itself was not standardized, in the sense, e.g., that a deliberate choice would had been made about which of several competing forms should be used in writing; and
3. the Middle Ages was a time of rapid language change in Swedish, perhaps more so than any subsequent period of similar length.

The first shows itself in a great variation in spelling; the "same" word can be spelled in a number of different ways, even on the same page of a document. Not only is there variation in the orthography itself, but also geographical variation, because no unified standard variety had been established at the time when the texts were produced.

The second factor makes itself felt in the number of variant forms in our inflectional paradigms (section 2).

As for the third factor, during the second half of the Old Swedish period the language underwent a development from the Old Norse (or Modern Icelandic, or Old English) mainly synthetic language type to the present, considerably more analytical state. In addition (or perhaps compounding this process), the sound system of Swedish was thoroughly reorganized.

For instance, in the nouns, the case system changed profoundly during this period, from the old four-case system (nominative, accusative, dative, genitive, in two numbers) to the modern system with a basic form and a genitive clitic which is the same in all declensions (as opposed to the old system where there were a number of different genitive markers), and where most functions that the older case forms expressed by themselves have been taken over by a combination of free grammatical morphemes and a much more rigid constituent order.

In the texts that interest us, these changes are in full swing, which manifests itself as variation in inflectional endings and in the use of case and other inflectional categories and in the distribution of the corresponding forms.

It is not always easy to tease out the contributions of these different factors to the linguistic motley evinced by the texts. Without doubt, the diachronic component is important – the texts are after all from a period three centuries in length – but it is also probable that the lack of standardization simply allows normal synchronic language variation to "shine through" in the texts, as it were, rather than being eliminated as is normally the case with modern, normalized written standard languages.

1.4. The Reference Dictionaries

The three main reference dictionaries of Old Swedish are:

- Söderwall (1884) (23,000 entries);
- Söderwall (1953) (21,000 entries); and
- Schlyter (1887) (10,000 entries)

The overlap between the three dictionaries is great, so that we are actually dealing with less than 25,000 different headwords. On the other hand, compounds – whether written as one word or separately – are not listed as independent headwords, but as secondary entries under the entry of one of the compound members. Thus, a full morphological description reflecting the vocabulary of the three dictionaries will contain many more entries, possibly by an order of magnitude more.

As an example of the kind of information that is available in the dictionaries, we will briefly discuss the entries for the word *fisker* (Eng. 'fish'), as it appears in these dictionaries. The entry *fisker* in Söderwall (1884) is shown in Figure 1. From this entry we learn that *fisker* is a masculine noun (indicated by "m.", in the second line of the entry),

[1]Of course we are quite aware of the fact that spelling variation is an empirical fact of modern language as well. There would be no need of spellcheckers otherwise. In the case of Old Swedish, however, there were no spelling norms, as far as we can tell. At the most, there were local scribal practices in the monasteries (which was where most of the text production occurred), different in different places and never codified, to our knowledge.

fisker (Söderwall) (fysker Lg 3: 301; -ar BSH 5: 5067 (1512). fiisker: fiisk RK 3: 4179. -ar), m. [Isl. fiskr] L. **1)** *fisk.* han tok w fiske tolpänigh Bu 100. taka fiska KL 12. thz första han katadhe vt sin krok tha fik lhan en storan fisk ib. ib 13. Bo 240. Lg 546, 3: 9, 10, 301, 302. i slike watne äru tholka fiska GO 978. ätin the fiska oc hwitan maat Bir 4: 15. färska ällir salte fiska ib 5: 32. tw pund skarpa fisca SD NS 1: 656 (1407). - koll. han (qvarndammen) skal vara open _ree vikur vm varenä _aa fisken gaar vpp ok swa lenge vm hösten. _aa vatneth er mykith ok fiksen gar vpp FH 3: 4 (1352). ib 4: 15 (1451), 16. SD 5: 699 (1347, gammal afskr.). äta fisk oc hwitan maat Bir 4: 15. VKR 17, 62. fäghin är han som fyrme ok findher han fikh (för fiskh) a diska GO 105. tw stykke fisk Bir 5: 31. tw stykke färskan fisk ib 32. eet stykke stekan fisk Bo 234. ii pund fiisk RK 3: 4179. **2)** ?iiij (4) lösa järn bultar, item xi (11) lösa fyskar, item 1 fangabult BSH 5: 506 (1512). - Jfr arbeidis-, bnären-, flat-, horn-, hval-, skal-, skarp-, skat-, sma-, spit-, stok-fisker. — **fiska bater** (-baater: -baat Su 363), m. *fiskarebåt.* Su 363. — **fiska ben**, n. *fiskben.* eet fiska ben sath fast j hans halse Bil 900. KL 370. ST 102. — **fiska dike**, n. *fiskdamm.* ST 299. — **fiska dröt**, f.L. — **fiska fiäl**, n. *fiskfjäll.* aff rutnom fiska fiällom Bir 3: 203. — **fiska fänge** (fiske-), n. *fiskafänge.* aff the fiske fängeno Lg 3: 11. aff hwario fiske fänge ib. — **fiska hovudh** (hwffwd LB 7: 265), n. *fiskhufvud.* LB 7: 265. PM XLVIII. — **fiska kyn** (-kön), n. *fiskslag.* alla handa fiska kön Al 6495. — **fiska lim**, n. *fisklim.* tak fiska lim giorth aff maghommen PM XLVII. — **fiska liver** (-leffwer: leffrenas PM XXXVIII), f. *fisklefver.* PM XXXVIII. — **fiska läghe**, n. *fiskläge.* RK 1: (Yngre red. af LRK) s. 263. Jfr fiskeläghe. — **fiska skal**, f. *musselskal, snäckskal.* trykte han ällir wredh vth aff vlla fätthen ena fiska skal äller eeth kar fwlt mz daagh (concham rore implevit) MB 2: 88. — **fiska slagh**, n. *fiskslag.* mang the fiska slagh, som aldrig fingos ther förra Lg 3: 11. — **fiska sudh** (fiiska sodh LB 7: 159), n. *fiskspad.* aff fersko fiska sudhi LB 3: 182. ib 7: 159. — **fiska thiuver**, m. L.

Figure 1: The entry *fisker* (Eng. 'fish') in Söderwall's dictionary of Old Swedish

and that it has been attested in a number of variant spellings (*fysker*, *fiisker*, *fiisk*). We also find references to occurrences of the word in the classical texts, and finally there is a listing of the compounds in which it occurs, e.g. *fiska slagh* (Eng. 'type of fish'). Söderwall (1953) is, basically, intended as a complement to Söderwall (1884), citing more forms and more attestations, originating in texts that became available after Söderwall's time.[2] Schlyter (1887) – as its title indicates – describes the vocabulary of the medieval Swedish laws, and its entries generally contain a bit less information than those in Söderwall (1884).

2. A Computational Morphology for Old Swedish

2.1. Functional Morphology

The tool we are using to describe the morphological component is Functional Morphology (FM) (Forsberg, 2007; Forsberg and Ranta, 2004). We chose this tool for a number of reasons: it provides a high-level description language (namely the modern functional programming language Haskell (Jones, 2003; Haskell, 2008)); it uses the character encoding UTF-8; it supports tasks such as (compound) analysis and synthesis; and, perhaps most importantly, it supports compilation to many standard formats, such as XML (The World Wide Web Consortium, 2000), LexC and XFST (Beesley and Karttunen, 2003), GF (Ranta, 2004), and full-form lexicons, and provides facilities for the user to add new formats.

[2]Although Söderwall is given as the author of this work, it was actually compiled after his death by members of *Svenska fornskriftsällskapet* (the Swedish Ancient Text Society).

The morphological model used in FM is *word and paradigm*, a term coined by Hockett (1954). A paradigm is a collection of words inflected in the same manner and is typically illustrated with an inflection table.

An FM lexicon consists of words annotated with paradigm identifiers from which the inflection engine of FM computes the full inflection tables.

Consider, for example, the citation form *fisker*, which is assigned the paradigm identifier *nn_m_fisker*. The paradigm identifier carries no meaning, it could just as well be any uniquely identifiable symbol, e.g. a number, but we have chosen a mnemonic encoding. The encoding is read as: "This is a masculine noun inflected in the same way as the word *fisker*" (which is trivially true in this case). If the paradigm name and the citation form is supplied to the inflection engine, it would generate the information in Table 1. To keep the presentation compact, we have contracted some word forms, i.e., the parenthesised letters are optional.

We also show (in the last column of Table 1) how this paradigm is presented in traditional grammatical treatises of Old Swedish, e.g. those by Wessén (1969) and Pettersson (2005). For a discussion of the differences between the our paradigms and those found in traditional grammatical descriptions, see section 2.3

The starting point of the paradigmatic specification, besides the dictionaries themselves, are the standard grammars of Old Swedish mentioned above, i.e., those by Noreen (1904), Wessén (Wessén, 1969; Wessén, 1971; Wessén, 1965), and Pettersson (2005). The number of paradigms in the current description by part of speech are

nn_m_fisker fisker ⇒

Lemma	fisker			**Traditional normalized form**
POS	nn			
Gender	m			
Number	**Def**	**Case**	**Word form**	
sg	indef	nom	*fisker*	*fiskẹr*
sg	indef	gen	*fisks*	*fisks*
sg	indef	dat	*fiski, fiske, fisk*	*fiski, fisk*
sg	indef	ack	*fisk*	*fisk*
pl	indef	nom	*fiska(r), fiskæ(r)*	*fiska(r)*
pl	indef	gen	*fiska, fiskæ*	*fiska*
pl	indef	dat	*fiskum, fiskom*	*fiskum*
pl	indef	ack	*fiska, fiskæ*	*fiska*
sg	def	nom	*fiskrin*	*fiskrin*
sg	def	gen	*fisksins*	*fisksins*
sg	def	dat	*fiskinum, fisk(e)num*	*fiskinum*
sg	def	ack	*fiskin*	*fiskin*
pl	def	nom	*fiskani(r), fiskæni(r)*	*fiskani(r)*
pl	def	gen	*fiskanna, fiskænna*	*fiskanna*
pl	def	dat	*fiskumin, fiskomin*	*fiskumin*
pl	def	ack	*fiskana, fiskæna*	*fiskana*

Table 1: The inflection table of *fisker*

as follows:

Part of speech	# of paradigms
Noun	38
Adjective	6
Numeral	7
Pronoun	15
Adverb	3
Verb	6

2.2. The FM Description

The paradigms of Old Swedish, which in our description amount to 75 paradigms, are defined using the tool Functional Morphology (FM). We will now give some technical details of the implementation by explaining how some of the verb paradigms in our morphology were defined. The main objective is not to give a complete description, but rather to provide a taste of what is involved. The interested reader is referred to one of the FM papers.

An implementation of a new paradigm in FM involves: a type system; an inflection function for the paradigm; an interface function that connects the inflection function to the generic lexicon; and a paradigm name. Note that if the new paradigm is in a part of speech previously defined, then no new type system is required.

A paradigm in FM is represented as a function, where the input is one or more word forms (typically the citation form or principle parts) and a set of morphosyntactic encodings, and the output of the function is a set of inflected word forms computed from the input word forms. It is a set instead of a single word form to enable treatment of variants and missing cases.

More concretely, if we represent the paradigm of regular nouns in English as a function, and only consider a morphosyntactic encoding for number, we would then define a function that expects a regular noun in nominative singular. If this function is given the word `"elephant"`, then the result would be another function. This function would, if an encoding for singular is given to the function, return `{"elephant"}`, and if an encoding for plural is given, return `{"elephants"}`. The resulting function may be translated into an inflection table given that the morphosyntactic encoding is ensured to be enumerable and finite (how this is ensured in FM will not be discussed here).

Turning now to the verb paradigms of Old Swedish, a `Verb` is a function from a morphosyntactic encoding, `VerbForm`, to a set of word forms with the abstract name `Str`.

```
type Verb = VerbForm -> Str
```

The type `VerbForm` defines the inflectional parameters of Old Swedish verbs. We only include those parameter combinations that actually exist, which will ensure, by type checking, that no spurious parameter combinations are created. A morphosyntactic encoding in FM is an algebraic data type, consisting of a list of constructors, where a constructor may have zero or more arguments. The vertical line should be interpreted as disjunction. The arguments here are also algebraic data types (only the definition of `Vox` is given here). A member of this type is, for example, `Inf Active`, where `Active` is a constructor of the type `Vox`.

```
data VerbForm =
 PresSg Modus Vox           |
 PresPl Person Modus Vox    |
 PretInd Number Person Vox |
 PretConjSg Vox             |
 PretConjPl Person Vox      |
 Inf Vox                    |
 ImperSg                    |
 ImperPl Person12
```

```
data Vox =
 Active |
 Passive
```

The `VerbForm` expands into 41 different parameter combinations. These parameter combinations may be given any string realization, i.e., we are not stuck with these rather artificially looking tags, we can choose any tag set. For example, instead of `PretConjSg Passive`, we have *pret konj sg pass*.

The next step is to define some inflection functions. We start with the paradigm of the first conjugation, exemplified by the word *ælska* (Eng. 'to love'). The inflection function `aelska_rule` performs case analysis on the `VerbForm` type. There is one input word form, which will be associated with the variable `aelska`. The function `strs` translates a list of strings to the abstract type `Str`. The function is built up with the support of a set of helper functions, such as `passive` that computes the active and passive forms, `tk` that removes the nth last characters of a string, and `imperative_pl` that computes the plural imperative forms (inflected for person).

```
aelska_rule :: String -> Verb
aelska_rule aelska p =
 case p of
 PresSg Ind Act ->
  strs [aelska++"r",aelska]
 PresSg Ind Pass -> strs [aelska ++"s"]
 Inf v -> passive v [aelska]
 ImperSg -> strs [aelska]
 ImperPl per ->
  imperative_pl per aelsk
 PresPl per m v   ->
  indicative_pl (per,m,v) aelsk
 PretInd Pl per v ->
  pret_ind_pl (per,v) aelsk
 PretConjPl per v ->
  pret_conj_pl (per,v) (aelsk++"a")
 PresSg Conj v ->
  passive v [aelsk++"i",aelsk++"e"]
 PretInd Sg _ v -> passive v [aelska++"þi"]
 PretConjSg v ->
  passive v [aelska++"þi", aelska++"þe"]
 where aelsk = tk 1 aelska
```

The inflection function `aelska_rule` computes 65 word forms from one input word form, e.g. *kalla* (Eng. 'to call').

Given that we now have defined an inflection function for a verb paradigm, we can continue by defining the other paradigms in relation to this paradigm, i.e., we first give the parameter combinations that differ from `aelska_rule` and finalize the definition with a reference to `aelska_rule`. This is demonstrated in the inflection function `foera_rule`, the paradigm of the third conjugation.

```
foera_rule :: String -> Verb
foera_rule foera p =
 case p of
 PresSg Ind Act ->
  strs [foer++"ir", foer++"i"]
 PresSg Ind Pass -> strs [foer++"s"]
 Inf v -> passive v [foera]
 ImperSg -> strs [foer]
 PretInd Pl per v ->
  pret_ind_pl  (per,v) foer
 PretConjPl per v ->
  pret_conj_pl (per,v) foer
 PretInd Sg _ v -> passive v [foer++"þi"]
 PretConjSg v ->
  passive v [foer++"þi", foer++"þe"]
 _  -> aelska_rule foera p
 where foer = tk 1 foera
```

The last inflection function we present, representing the fourth conjugation paradigm, is `liva_rule`, defined in terms of `foera_rule`. Note that we use two different forms when referring to `foera_rule`: we use `lif++"a"`, i.e., the input form where 'v' has been replaced with 'f', for the preterite (i.e., past tense) cases, and the input form for all other cases.

```
liva_rule :: String -> Verb
liva_rule liva p =
 case p of
 PresSg Ind Act  ->
  strs [liv++"er", liv++"ir", liv++"i"]
 PresSg Ind Pass -> strs [lif++"s"]
 ImperSg         -> strs [lif]
 p | is_pret p   -> foera_rule (lif++"a") p
 _               -> foera_rule liva p
 where liv = tk 1 liva
       lif = v_to_f liv
```

When the inflection functions are defined, we continue with the interface functions. An interface function translates one or more input words, via an inflection function, into an entry in the generic dictionary. This is done with the function `entry` that transforms an inflection function into an inflection table. If the current part of speech has any inherent parameters such as gender, those would be added here. The inherent parameters are not inflectional, they describe properties of a word, which is the reason why they appear at the entry level.

```
vb_aelska :: String -> Entry
vb_aelska = entry . aelska_rule

vb_foera :: String -> Entry
vb_foera = entry . foera_rule

vb_liva :: String -> Entry
vb_liva = entry . liva_rule
```

The interface functions need to be named to connect them with an external lexicon. This is done with the function `paradigm`. The names are typically the same as those of the interface functions. Every paradigm is also given a list of example word forms, which provides paradigm documentation and enables automatic generation of an example inflection table, which is done by FM applying the current interface function to its example word forms. The list of paradigm names, denoted here with `commands`, is later plugged into the generic part of FM.

```
commands = [
  paradigm "vb_aelska" ["ælska"] vb_aelska,
  paradigm "vb_foera"  ["føra"]  vb_foera,
  paradigm "vb_liva"   ["liva"]  vb_liva
           ]
```

We can now start developing our lexicon. The lexicon consists of a list of words annotated with their respective paradigm, e.g. the word *røra* (Eng. 'to touch') and *føra* (Eng. 'to move'), which is inflected according to the paradigm `vb_foera`.

```
vb_foera "røra" ;
vb_foera "føra" ;
```

The lines above are put into an external file that is supplied to the compiled runtime system of FM.

2.3. The Development of the Morphological Description and the Lexicon

In this project, we have collaborated with a linguist who is also an expert on orthographic and morphological variation in Old Swedish. In the first phase of the project, she defined the inflectional paradigms on the basis of the dictionaries and the actual variation empirically observed in the texts.

The FM description was developed in parallel with this work. The linguist selected a set of sample words from the dictionaries and annotated those with the appropriate paradigms. The full inflection tables could then be generated immediately and the result evaluated by the linguist.

At the time of writing, about 3,000 main lexical entries (headwords; see section 1.4) have been provided with inflectional information in the form of a paradigm identifier.

In our work in a parallel project to the one described here, where we are producing a large computational morphological lexicon for modern Swedish (Borin et al., forthcoming 2008a; Borin et al., forthcoming 2008b), the number of inflectional classes (paradigms) turns out to be on an order of magnitude more, i.e., around 1,000 rather than around 100.[3] Note that this holds equally for the written standard language and colloquial spoken Swedish.[4] This is something that calls for an explanation, since under the (generally accepted, at least in some form) assumption of uniformitarianism (Janda and Joseph, 2003), we would not expect to find less diversity in Old Swedish than in the modern language.

First we may note that our morphological description is not yet complete. For example, while it covers all four weak verb conjugations, as yet it accounts for only two out of the nine or so classes of strong and irregular verbs. However, even standard grammars of Old Swedish like that of Wessén (1969) list somewhere in the vicinity of 100 paradigms, and no more. We believe that the main factor here is our lack of information. For many lexical entries it is even difficult to assign an inflectional class, because the crucial forms are not attested in the extant texts, and of course, there are no native speakers on whose linguistic intuitions we could draw in order to settle the matter.

Some of the diversity built into our paradigms could thus conceivably be a case of different lexical entries now brought under the same paradigm, actually consistently using different alternatives for expressing a particular combination of morphosyntactic features; we will probably never know.

In the standard reference grammars of Old Swedish, inflectional paradigms are consistently idealized in the direction of a (re)constructed Old Swedish, arrived at on the basis of historical-comparative Indo-European and Germanic studies. In this connection, the actual variation seen in texts has been interpreted as a sign of language change, of "exceptional" usages, etc.[5] (Johnson, 2003). In our paradigms, we have endeavored to capture the actual variation encountered in the texts and in the dictionary examples (but see section 3).

3. Computational Treatment of Variation

As we have mentioned already (section 1.3), the source of variation in the texts are of three kinds: no standardized spelling; no standardized forms; and language change (diachronic drift). For our work on the computational morphological description of Old Swedish, we have found it natural and useful to make an additional distinction, namely that between *stem variation* and *ending variation*, since it has seemed to us from the outset that we need to treat stems and endings differently in this regard.

This gives us altogether six possible combinations of factors, as shown in the following table:

	spelling variation	lack of lg standardization	language change
stem variation	S_1	S_2/L_1	S_3/L_2
ending variation	M_1	M_2	M_3

(Legend: S=spelling rules; L=lexical component; M=morphological component)

[3] The distribution of inflectional patterns in the modern language is Zipfian in shape: Nearly half the paradigms are singletons, almost a fifth of them have only two members, etc.

[4] Although for slightly different reasons in the two cases: In the written standard language, it is generally the low-frequency words that have unique paradigms, e.g. learned words and loanwords. In the spoken language, high-frequency everyday words show variation in their inflectional behavior. There is some overlap, too, e.g. the strong verbs.

[5] and possibly even of carelessness or sloth on the part of the scribes; cf. the following quote, which well captures an attitude toward linguistic variation traditionally prevalent among linguists:

> Variation in Navajo pronunciation had long disturbed Haile (to Sapir, 30 March 1931: SWL): "Sometimes I do wish that the informants would be more careful in pronunciation and follow some system which would conform to theory. ... Apparently no excuse, excepting that informants are too lazy to use it correctly." Sapir responded (6 April 1931: SWL) that—at least in collecting texts—it was "not absolutely necessary to have the same words spelled in exactly the same way every time."
> (Darnell, 1990, 257)

The table reflects the fact that we have decided already to handle all inflectional ending variation – regardless of its origin – in the morphological component, i.e., our paradigms contain all attested ending variants, still a finite and in fact rather small set, which partly motivates their uniform treatment.

Representing a dead language with a finite corpus of texts, the Old Swedish stems could in theory be treated in the same way. The corpus is big enough, however, that we will need to treat it as unlimited in practice, and hence the stems as a set that cannot be enumerated.

In order not to bite off more than we can chew, we have tentatively decided to treat all stem variation as a spelling problem (with one exception; see below). It will then be natural to look to some kind of solution involving edit distance, e.g. *universal Levenshtein automata*, see, for example, Mihov and Schulz (2004).

However, the spelling is not completely anarchistic, far from it: For example, the /i:/ sound will be written <i>, <y>, <j>, <ii>, <ij>, and possibly in some other ways, but not, e.g., <a> or <m>, etc. Thus, a rule-based method may be more appropriate, or possibly a hybrid solution should be sought.

In the table above, the use of subscripts (S_1, M_3, etc.) hints at the possibility of distinguishing formally among different types of information even within a component. The present morphological description does not make a distinction between ending variation due to spelling variants of the "same" ending (from a historical-normative point of view – e.g., indef sg dat *fiski/fiske* – and ending variation whereby "different" endings occupy the same paradigmatic slot, e.g., indef sg dat *fiski/fisk*. However, there is no technical reason that we could not make this and other distinctions on the level of paradigms or even on the level of individual lexical entries. In fact, our work on the Old Swedish morphological description has clearly indicated the need for this kind of facility.[6]

There is one kind of stem variation which does not fit neatly into the picture painted so far, namely that brought about by inflectional morphological processes, in our case those of Ablaut and Umlaut. At the moment, the strong verb class paradigms do not account for variation in the realization of the Ablaut grades of the stem vowels – which of course we find in the texts – and we are still undecided as to how to treat them, by a separate normalization step or in the FM description. In the latter case we would then probably need to duplicate some information already present in the spelling rules component.

4. Summary and Conclusions

We have implemented a morphological description for Old Swedish using Functional Morphology, a tool which supports automatic morphological analysis and generation, as well as the generation of full-form lexicons. The description is intended to be used in an online Old Swedish text reading aid, where it will perform on-the-fly analysis of words in the texts in order to present the user with possible lexicon entries for the word.

The description is fairly complete, but its usefulness for this intended practical purpose is still limited by the large amount of linguistic variation found in the texts.

We have created a small test lexicon (about 3,000 entries), and we are now working on adding inflectional information to all of the headwords in the digital versions of the Old Swedish reference dictionaries (section 1.4).

We have started to look at the linguistic variation characteristic of the Old Swedish texts. Variation in inflectional endings is already uniformly handled in the morphological component, regardless of its origin, while we have still not made the final decision on a strategy to handle the various kinds of stem variation found in Old Swedish.

Acknowledgements

The Old Swedish morphological description on which our computational morphology is based, was made by Rakel Johnson, Department of Swedish Language, University of Gothenburg.

The work presented here was financed in part by Swedish Research Council grant 2005-4211 (2006–2008) awarded to Aarne Ranta, Chalmers University of Technology, for the research project entitled *Library-Based Grammar Engineering*, and in part by the Faculty of Arts, University of Gothenburg, through its support to Språkbanken (the Swedish Language Bank).

We would also like to acknowledge CLT – the Centre for Language Technology, Göteborg <http://www.clt.gu.se> – for providing a creative atmosphere in which multidiscplinary collaborations such as this come naturally.

5. References

Kenneth R. Beesley and Lauri Karttunen. 2003. *Finite State Morphology*. CSLI Publications, Stanford University, United States,.

Lars Borin, Markus Forsberg, and Lennart Lönngren. forthcoming 2008a. The hunting of the BLARK – SALDO, a freely available lexical database for Swedish language technology. In *Festschrift to Professor Anna Sågvall Hein*. Uppsala University, Dept. of Linguistics and Philology.

Lars Borin, Markus Forsberg, and Lennart Lönngren. forthcoming 2008b. SALDO – the Swedish associative thesaurus, version 2. Technical report, Språkbanken, University of Gothenburg.

Regna Darnell. 1990. *Edward Sapir: Linguist, Anthropologist, Humanist*. University of California Press, Berkeley / Los Angeles / London.

Markus Forsberg and Aarne Ranta. 2004. Functional Morphology. *Proceedings of the Ninth ACM SIGPLAN International Conference of Functional Programming, Snowbird, Utah*, pages 213–223.

Markus Forsberg. 2007. *Three Tools for Language Processing: BNF Converter, Functional Morphology, and Extract*. Ph.D. thesis, Göteborg University and Chalmers University of Technology.

[6]It is not difficult to think of situations where this would be useful in modern language descriptions as well; for instance, it would be useful to be able to record the frequency of occurrence of homographs according to which lexical entry they represent.

Haskell. 2008. Haskell homepage. <http://www.haskell.org>.

Charles Hockett. 1954. Two models of grammatical description. *Word*, 10:210–234.

Richard D. Janda and Brian D. Joseph. 2003. On language, change, and language change – or on history, linguistics, and historical linguistics. In Brian D. Joseph and Richard D. Janda, editors, *Handbook of Historical Linguistics*, pages 3–180. Blackwell, Oxford.

Rakel Johnson. 2003. *Skrivaren och språket*. Ph.D. thesis, Department of Swedish Language, Göteborg University.

Simon Peyton Jones. 2003. *Haskell 98 Language and Libraries: The Revised Report*. Cambridge University Press.

Fred Karlsson, Atro Voutilainen, Juha Heikkilä, and Arto Anttila, editors. 1995. *Constraint Grammar: A Language-Independent System for Parsing Unrestricted Text*. Number 4 in Natural Language Processing. Mouton de Gruyter, Berlin and New York.

Stoyan Mihov and Klaus Schulz. 2004. Fast approximate search in large dictionaries. *Computational Linguistics*, 30(4):451–477.

John Nerbonne and Petra Smit. 1996. GLOSSER-RuG: In support of reading. In *COLING–96. The 16th International Conference on Computational Linguistics. Proceedings, Vol. 2*, pages 830–835, Copenhagen. ACL.

Adolf Noreen. 1904. *Altschwedische Grammatik*. Halle. Facsimile available online: <http://lexicon.ff.cuni.cz/texts/oswed_noreen_about.html>.

Gertrud Pettersson. 2005. *Svenska språket under sjuhundra år*. Studentlitteratur, Lund, Sweden.

Aarne Ranta. 2004. Grammatical Framework: A Type-theoretical Grammar Formalism. *The Journal of Functional Programming*, 14(2):145–189.

P. Rayson, D. Archer, A. Baron, J. Culpeper, and N. Smith. 2007. Tagging the Bard: Evaluating the accuracy of a modern POS tagger on Early Modern English corpora. In *Proceedings of Corpus Linguistics 2007*. University of Birmingham, UK.

C.J. Schlyter. 1887. *Ordbok till Samlingen af Sweriges Gamla Lagar. (Saml. af Sweriges Gamla Lagar 13)*. Lund, Sweden.

Knut Fredrik Söderwall. 1884. *Ordbok Öfver svenska medeltids-språket. Vol I–III*. Lund, Sweden.

Knut Fredrik Söderwall. 1953. *Ordbok Öfver svenska medeltids-språket. Supplement. Vol IV–V*. Lund, Sweden.

The World Wide Web Consortium. 2000. Extensible Markup Language (XML). <http://www.w3.org/XML/>.

Elias Wessén. 1965. *Svensk språkhistoria: Grundlinjer till en historisk syntax*. Stockholm, Sweden.

Elias Wessén. 1969. *Svensk språkhistoria: Ljudlära och ordböjningslära*. Stockholm, Sweden.

Elias Wessén. 1971. *Svensk språkhistoria: Ordböjningslära*. Stockholm, Sweden.

Generating Tailored Texts for Museum Exhibits

Dana Dannélls

Natural Language Processing Research Unit
Department of Swedish Language
University of Gothenburg
SE-405 30 Gothenburg, Sweden
dana.dannells@svenska.gu.se

Abstract

This paper reports work that aims to generate texts in multiple languages from ontologies following the Conceptual Reference Model (CRM) ISO standard for conceptual models of museums. The rationale of this work is to increase users' knowledge and interest in the cultural heritage domain by allowing the user to select his preferable syntax presentation and influence the order of the generated information using generation techniques and Semantic Web technologies. We chose for study a small amount of logical relations represented in the ontology and wrote a grammar that is capable to describe them in natural language through user editing. We present the multilingual source authoring environment, which is built upon the grammatical framework (GF) formalism and show how it is utilized to generate multiple texts from the CRM domain ontology. The initial results comprise texts, which vary in syntax and content.

1. Introduction

During the last decade, the awareness of the need for personalization has become fundamental for cultural institutions such as museums and libraries while aim to produce textual descriptions of museum exhibits tailored to the visitor's knowledge, interests, and personal preferences, such as preferred vocabulary, syntax, sentence length etc. One of the first examples of personalization in a museum context was developed in the Intelligent Labelling Explorer (ILEX) project,[1] by using Natural Language Generation (NLG) techniques. More recently, applications within the cultural heritage (CH) domain have seen an explosion of interest in these techniques (Novello and Callaway, 2003; O'Donnell et al., 2001; Androutsopoulos et al., 2007).

The process of NLG starts from an ontology that describes a certain domain. Recently, natural language generators that are targeted towards the Semantic Web ontologies have started to emerge. A strong motivation for generating texts from ontologies is that the information represented in an ontology has a true potential to provide a large amount of text if this text is realized correctly. Gradually, the cultural heritage knowledge domain which is often characterized by complex semantic structures and large amounts of information from several different sources will benefit from the complete generation of the information delivered in the ontology.

Web ontology languages pose many opportunities and challenges for language generators. Although standards for specifying ontologies provide common representations to generate from, existing generation components are not compatible with the requirements posed by these new-coming standards. This issue has been previously addressed by developing domain-dependent authoring interfaces that are built upon an ontology and that allows it to be deployed through knowledge editing (Brun et al., 2000; Hartley et al., 2001; van Deemter et al., 2005). These interfaces are links between the ontology and the user who can manipulate the content of the document indirectly in his/her own language. An example of a template-based authoring tool that makes use of this technique within the CH domain was presented by Androutsopoulos et al. (2007). An alternative approach to template-based NLG that is particularly relevant in cases where texts are generated from logical forms in several languages simultaneously is a grammar-based approach (Bateman, 1997).

In this paper we present a multilingual source authoring tool, which is built upon the grammatical framework (GF) formalism to generate texts from the underlying semantic representation that is based on the Conceptual Reference Model (CRM) domain ontology. The authoring environment is similar to those described in Power and Scott (1998), Dymetman et al. (2000) and van Deemter and Power (2003). [2] The focus is on the process starting from a fixed semantic representation to a surface realization, with emphasis on the syntactical sentence structure, and the content variation.

The structure of the paper is as follows. In section 2 we elaborate the notion of ontology and describe both the reference ontology model and the grammar formalism that our application is built upon. Section 3 presents the grammar implementation and explains how it is utilized to generate tailored descriptions from a formal representation language. Section 4 finishes with conclusions and a discussion of future work.

2. Background

In the context of the work presented here, an *ontology* is understood as a formal model that allows reasoning about concepts, objects and about the complex relation between them. An ontology holds meta-level information about different types of entities in a certain domain and provides a structure for representing contexts, it is not human readable as it is designed to be processed by computer systems.

[1] http://www.hcrc.ed.ac.uk/Site/ILEXINTE.html

[2] The advantages of utilizing this family of domain authoring approaches that are coupled with multilingual text generation are elaborated in Scott (1999).

Examples of Web ontology-languages that have been developed by the W3C Web-Ontology working group are OWL and DAML+OIL.[3] The basis for the design of these Web technology languages based on the RDF Schema is the expressive Description Logic (DL) *SHIQ* (Horrocks et al., 2003). These languages provide extensive reasoning capabilities about concepts, objects and relationships between them.

2.1. Generating from an Ontology

In an ontology, an object may be described by semantic graphs whose nodes (concepts) represent parts of an object, and the arcs (relations) represent partial constrains between object parts. Each relation described in a logical language is binary, i.e. it connects between two nodes. In order to present a piece of information about an object represented in an ontology, multiple sentences must be formulated. It becomes valuable if these sentences that build the final text can be adapted to various contexts or users.

There has been successful attempts to generate from ontologies (Wilcock, 2003; Wilcock and Jokinen, 2003; Bontcheva and Wilks, 2004; Bontcheva, 2005). Wilcock (2003) and Wilcock and Jokinen (2003) have shown how RDF/XML generation approach can be extended so that the information embedded in the ontology can be exploited to generate texts from Web ontology-languages such as DAML+OIL and OWL without the need for a lexicon. Bontcheva (2005) demonstrated how to minimize the effort when generating from Web ontology-languages while being more flexible than ontology verbalisers. Some of the difficulties reported by these authors concern lexicalization and in establishing context variations.

2.2. The CRM Ontology

One initiative to enable an ontology in the context of the cultural heritage is the Conceptual Reference Model domain ontology. The International Committee for Documentation of the International Council of Museums Conceptual Reference Model (CIDOC-CRM)[4] is a core ontology and ISO standard for the semantic integration of cultural information with library archive and other information (Doerr, 2005). The primary role of the CRM is to enable information exchange and integration between heterogeneous sources of cultural heritage information.

The central idea of the CIDOC-CRM is that the notion of historical context can be abstracted as things and people. It concentrates on the definition of relationships rather than classes to capture the underlying semantics of multiple data and meta structures. It tends to provide an optimal analysis of the intellectual structure of cultural documentation in logical terms, which is available in several formats such as RDF and OWL that have hardly been explored yet. The work described in this paper is based on the OWL version of the ontology.[5]

[3]http://www.w3.org/TR/

[4]http://cidoc.ics.forth.gr/

[5]http://cidoc.ics.forth.gr/OWL/cidoc_v4.2.owl

2.3. The Grammatical Framework (GF)

The Grammatical Framework (Ranta, 2004) is a functional grammar formalism based on Martin-Löf's type-theory (Martin-Löf, 1973) implemented in Haskell.[6] GF focuses on language independent semantic representations. It differentiates between domain dependent and domain independent linguistic resources, as it is designed to be applicable both to natural and to formal languages. One abstract grammar can have several corresponding concrete grammars; a concrete grammar specifies how the abstract grammar rules should be linearized in a compositional manner.

Multilingual functional grammatical descriptions permit the grammar to be specified at a variety of levels of abstraction, which is especially relevant for constructing a detailed mapping from semantics to form. This aspect is crucial for natural language generation to work. What makes the grammar suitable for generating from ontologies and in particular from OWL, is that it allows multiple inheritance. GF has three main module types: abstract, concrete, and resource. Abstract and concrete modules are top-level, in the sense that they appear in grammars that are used at runtime for parsing and generation. They can be organized into inheritance hierarchies in the same way as object-oriented programs. The main advantage with converting the ontology to GF is that we can make use of the rich type system in the concrete syntax for capturing morphological variations. Our approach is based on the idea suggested by Khegai et al. (2003) who utilized GF to automatically generate multiple texts from semantic representations. The source authoring environment deploys similar techniques to those introduced in Power and Scott (1998), Dymetman et al. (2000) and van Deemter and Power (2003).

3. Generating from the Ontology

We chose for study a small amount of logical relations represented in the ontology and wrote a grammar that is capable to describe them in natural language through user editing. The following code is a fragment taken from the ontology we employed. The code states that the class *PaintingP9091* must have at least one value *TypeValue* on property *has_type*; the individual *TypeValue* is an instance of the class *cidoc:E55.Type*[7] and has two property values: “tool” and “painting”.

```
<owl:Class  rdf:about="PaintingP9091">
 <owl:Restriction>
   <owl:onProperty rdf:resource="&cidoc;P2F.has_type"/>
     <owl:hasValue rdf:resource="#TypeValue"/>
  </owl:Restriction>
</owl:Class>
<owl:Thing rdf:about="#TypeValue">
   < rdf:type rdf:resource="&cidoc;E55.Type"/>
   <Tool  rdf:datatype="&xsd;string">tool
   </Tool>
   <Painting  rdf:datatype="&xsd;string">painting
```

[6]Haskell is a standardized purely functional programming language with non-strict semantics. Similar to Lisp and Scheme.

[7]The notation &cidoc; is used instead of the whole namespace,i.e http://cidoc.ics.forth.gr/OWL/cidoc_v4.2.owl#

</Painting>
</owl:Thing>

The above fragment exemplifies the representation of the classes and relationships that are utilized by the grammar. In the grammar implementation, classes are represented as categories; properties are functions (rules) between two categories, where each property links between two classes; individuals are lexical categories (strings). Below is a representation of the *mkObject*, which corresponds to a function that links between the classes of an *Object*:

mkObject:ObjectNodeI→ObjectNodeII→ObjectNodeIII→Object;

In this example the *Object* category corresponds to *PaintingP9091*. Each *ObjectNode* is a class, according to the above ontology representation, *ObjectNodeI* corresponds to the cidoc class *cidoc:E55.Type*. It is followed by *ObjectNodeII*, i.e. *cidoc:E52.Time-Span* and *ObjectNodeIII*, i.e. *cidoc:E21.Person*, as shown below.

{Type} *instance_of* ObjectNodeI
{Time-Span} *instance_of* ObjectNodeII
{Person} *instance_of* ObjectNodeIII

Consequently, individuals such as "tool" and "painting" are terminals and are declared in the concrete syntax. In the next sections we describe the abstract and the concrete representations

3.1. The Abstract Representation

The abstract syntax is a context-free grammar where each rule has a unique name. An abstract rule in GF is written as a typed function. The categories and functions are specified in GF by *cat* and *fun* declarations. Below is a fragment of the grammar:

cat

Object ;ObjectNodeI ; Type ;
ObjectNodeII ; Time-Span ;
ObjectNodeIII ; Person ;

fun

HasType_This : Type → ObjectNodeI;
HasType_Here : Type → ObjectNodeI;
HasType_Template : Type → ObjectNodeI;
HasTimeSpan: Time-Span → ObjectNodeII;
CarriedOutBy_Painting: Person → ObjectNodeIII;
CarriedOutBy_Tool: Person → ObjectNodeIII;

The abstract syntax gives a structural description of a part of the domain. It has several advantages, one of which is the ability to utilize the same categories differently depending on the semantic complexity of the context. Here we declared three functions for the *ObjectNodeI* to achieve context variations, though very simple ones. Similarly, we declared two functions for the *ObjectNodeIII*, however, the difference between *CarriedOutBy_Painting* and *CarriedOutBy_Tool* is the choice of the verb in the linearization rule. The verb *painted by* is applied when the subject is the noun *painting*, but the verb *created by* is applied when the subject is the noun *tool*, in cases when the object is an instance that belongs to the category *Person*.

3.2. The Concrete Representation

Each category and function introduced in the abstract syntax has a corresponding linearization type in the concrete syntax. Linearization rules are declared differently for each target language. In addition, each concrete syntax also contains grammatical parameters and grammar rules, which are used to ensure grammatical correctness for each language, in our case English and Swedish. An example of linearization rules taken from the English concrete syntax is the following:

lin

CarriedOutBy_Painting obj = {s = det ! obj.num ++ cop ! obj.num ++ "painted by" ++ obj.s ; num=obj.num};

Painting = {s = "painting" ; num = sg} ;
Painting = {s = "paintings" ; num = pl} ;

Grammatical features are supported by GF and the agreement between the pronoun and the verb is enforced in the generated sentences. The variable *obj* represents a terminal string. The parameter *num* is an abbreviation for the parameter type "number", it contains the inherent number that can be either singular (sg) or plural (pl). The operation *det* is a determiner, and the operation *cop* is copula verb.

3.3. The Authoring Environment

Figure 3 illustrates the source authoring environment. The left-side window shows the abstract syntax tree, which represents the *Object* structure. The large window positioned to the right is the linearization area, the editing focus is presented as the highlighted metavariable *?3*. The bottom area shows the context-dependent refinement for the *ObjecNodeIII*, there are two possible relations to choose from.

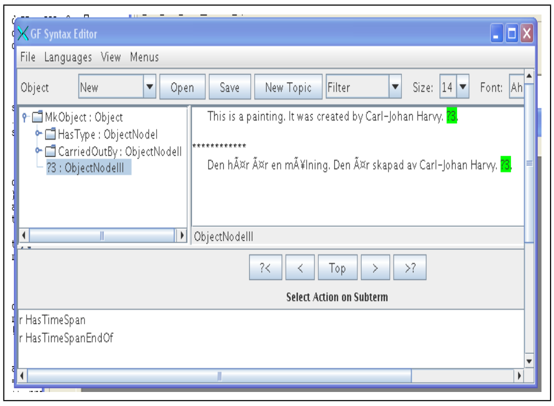


Figure 1: The GF source authoring environment.

The authoring tool that is built upon the GF grammar makes it possible to generate the following texts:

English
(1) Here we have a painting. It was painted by Carl-Johan Harvy. It was made in 1880.

(2) This is a tool. It was made in 1880. It was created by Carl-Johan Harvy.

(3) On the second floor of the history museum we have paintings. They were created by Siri Derkert. They were produced in Italy.

Swedish
(1) Här har vi en målning. Den är målad av Carl-Johan Harvy. Den är gjord på 1880 talet.

(2) Det här är ett redskap. Det är gjort på 1880 talet. Det är tillverkat av Carl-Johan Harvy.

(3) På andra våningen i historiska museet har vi målningar. De är tillverkade av Siri Derkert. De är producerade i Italien.

The difference between the first and second sentence is the order in which the *ObjectNodeII* and the *ObjectNodeIII* appears, this is done with the help of the *variants* function that allows for syntactic variations by reordering the linearized categories. The third sentence illustrates a typical example of a combined template and grammar based generation, e.g. the fixed sentence: "On the second floor of the history museum" that has been prewritten.

4. Conclusions and Future Work

In this paper we present a multilingual grammar-based approach, the aim of which is to generate exhibit descriptions following the CRM domain ontology. We present the GF authoring tool, which allows users to choose the content and the form of the output text. In this work we attempt to establish new methods that support the user on receiving information based on the Semantic Web in the cultural heritage domain.
We chose for study a small amount of logical relations represented in an ontology and have started to examine the capabilities of utilizing a grammar to bridge between ontology representations and different users. This is one of the few attempts to generate texts from CIDOC CRM.
Future work will focus on ontology studies and on particular problems of generating for cultural heritage. We are also planning to utilize the Resource Grammar Library that has been developed to provide the linguistic details for application grammars on different domains. This will be a step towards high quality summary generation. Our goal is to built a grammar that reflects the ontology structure and supports all the OWL features to allow the user to interact with the full ontology.

5. References

I. Androutsopoulos, J. Oberlander, and V. Karkaletsis. 2007. Source authoring for multilingual generation of personalised object descriptions. *Natural Language Engineering*, 13(3):191–233.

J. A. Bateman. 1997. Enabling technology for multilingual natural language generation: The kpml development environment. *Natural Language Engineering*, 3(15–55).

K. Bontcheva and Y. Wilks. 2004. Automatic report generation from ontologies: the miakt approach. In *Nineth International Conference on Applications of Natural Language to Information Systems (NLDB)*. Manchester.

K. Bontcheva. 2005. Generating tailored textual summaries from ontologies. In *Second European Semantic Web Conference (ESWC'05)*, Crete.

C. Brun, M. Dymetman, and V. Lux. 2000. Document structure and multilingual authoring. In *In Proc. of First International Natural Language Generation Conference (INLG)*, Mitzpe Ramon, Israel, June.

M. Doerr. 2005. The cidoc crm, an ontological approach to scheme heterogeneity. In Dagstuhl Seminar, editor, *Semantic Interoperability and Integration*, pages 1862–4405.

M. Dymetman, V. Lux, and A. Ranta. 2000. Xml and multilingual document authoring: Convergent trends. In *In Proceedings of COLING*. Saarbrucken, Germany.

A. Hartley, D. Scott, J. Bateman, and D. Dochev. 2001. Agile – a system for multilingual generation of technical instructions. In *In MT Summit VIII, Machine Translation in the Information Age*, pages 145–150.

I. Horrocks, P.F. Patel-Schneider, and F. van Harmelen. 2003. From shiq and rdf to owl: The making of a web ontology language. *Journal of Web Semantics*, 1(1):7–26.

J. Khegai, B. Nordström, and A. Ranta. 2003. Multilingual syntax editing in gf. pages 453–464, Mexico. Springer.

P. Martin-Löf. 1973. An intuitionistic theory of types: Predicative part. In H. E. Rose and J. C. Shepherdson, editors, *In Proc. of Logic Colloquium '73, Bristol, UK*, volume 80, pages 73–118.

A. Novello and C. Callaway. 2003. Porting to an Italian surface realizer: A case study. In *Proc. of the 9th European Workshop on NLG*, pages 71–78.

M. O'Donnell, J. Oberlander C. Mellish, and A. Knott. 2001. Ilex: an architecture for a dynamic hypertext generation system. *Natural Language Engineering*, 7(3):225–250.

R. Power and D. Scott. 1998. Multilingual authoring using feedback texts. In *17th International Conference on Computational Linguistics*, pages 1053–1059.

A. Ranta. 2004. Grammatical framework, a type-theoretical grammar formalism. *Journal of Functional Programming*, 14(2):145–189.

D. Scott. 1999. The multilingual generation game: Authoring fluent texts in unfamiliar languages. In *Proc. of the 16th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1407–1411.

K. van Deemter and R. Power. 2003. High-level authoring of illustrated documents. *Natural Language Engineering.*, 9(2):101–126.

K. van Deemter, E. Krahmer, and M. Theune. 2005. Real versus template-based natural language generation: a false opposition? *Computational Linguistics*, 31(1):15–23.

G. Wilcock and K. Jokinen. 2003. Generating responses and explanations from rdf/xml and daml+oil. In *Knowledge and Reasoning in Practical Dialogue Systems IJCAI-2003*, pages 58–63.

G. Wilcock. 2003. Talking owls: Towards an ontology verbalizer. In *Human Language Technology for the Semantic Web and Web Services*, pages 109–112.

Non-Standard Russian in Russian National Corpus (RNC)

Elena Grishina

Institute of Russian Language, RAS
121019, Volkhonka, 18/2, Moscow, Russia
rudi2007@yandex.ru

Abstract

The RNC is a 150 million-word collection of Russian texts, thus, it is the most representative and authoritative corpus of the Russian language. It is available on the Internet at www.ruscorpora.ru. The RNC contains texts of all genres and types, which cover Russian of the 18-21 centuries. That's why the problem of non-standard units in the RNC and the possible ways of their normalization is now the most important and urgent. To solve the problem we suggest 1) expanding the base of RNC Vocabulary, 2) improving the RNC Grammar parser in some dimensions, 3) using the preserving annotation in some kinds of texts. The paper describes what type of non-standard units may be normalized with this or that strategy of standardizing and which are the cases, when the users paradoxically are not interested in transforming the non-standard units into the standard ones. The paper contains considerable number of non-standard samples from the sub-corpora of the RNC – the Spoken, Electronic, Accentological Corpora, the Corpus of Old Russian Texts and of Poetry.

1. Russian National Corpus: Content and Structure

The Russian National Corpus (RNC) is a collection of Russian texts of the 18–21 centuries, which are supplied with different types of annotation – morphological, semantic, accentological, sociological, metatextual.

The RNC has been functioning since April 2003 and is accessible at www.ruscorpora.ru. Now it contains circa 150 million words. It's planned to bring its capacity up to 200 million words[1]. Therefore, the today RNC has become one of the most representative and authoritative corpora of Russian.

The RNC embraces all genres of Russian written texts – fiction (prose, drama, poetry) and non-fiction (journalism, scientific texts, memoirs, letters, business documents, theological writings, etc.). Also the RNC contains the scripts of Spoken Russian, including movie transcripts. A user of the RNC can form his sub-corpus according to any parameter of metatextual, sociological, morphological and semantic annotation and to any combination of all possible parameters.

The ***metatextual annotation*** describes a text as a whole (its title, author, date of creation, genre/type, and so on). The ***morphological annotation*** assigns the grammatical characteristics to a wordform. The ***semantic annotation*** characterizes a lexeme as a member of this or that semantic class, e. g. names of persons, of animals, diminutives, adjectives of good-rating, verbs of speech or moving, and so on. The ***sociological annotation*** is used within the Corpus of Spoken Russian (see below): every token is annotated from the point of view of sex and age of a speaker (if they are known), and thus a user may form his own sub-corpus).

The whole structure of the RNC now looks as follows:

- the ***Main Corpus*** [97,5 million tokens] – the corpus of contemporary Russian (the period between the middle of the 20 century and the beginning of the 21 century; the Main Corpus also includes the ***Electronic Sub-Corpus*** [1,2 million tokens] – the collection of the internet texts)
- the ***Corpus of Early Texts*** (the 18 – the middle of the 20 century, 51 million tokens)
- the ***Corpus of Poetry*** (Russian poetry of the 18–21 centuries, 1,9 million tokens)
- the ***Corpus of Spoken Russian*** (the scripts of spoken texts and movie transcripts of the period of 1934–2007, 5,5 million tokens)
- the ***Corpus of Dialect Texts*** (contains the records of dialect specimens from different regions of Russia, 0,144 million tokens)
- the ***Accentological Corpus*** (contains Russian poetry with marked arses[2] and the movie transcripts with marked stress; the Accentological Corpus will be available at 2008).

It's quite natural that any corpus of so large capacity and of so wide spectrum of text databases as the RNC contains a lot of non-standard words and wordforms. That's why the creators of the RNC were forced to elaborate the integral and consecutive strategy to settle the Non-Standard Units (NSU) and to turn them into the standards, as far as possible.

2. Types of NSU and Technology of Normalization

Any token in the RNC is characterized as non-standard if it isn't presented in the RNC Vocabulary. The Vocabulary consists of three zones: 1) Lemma (the title of the entry, the citation form of a word), 2) Wordform (the set of the Inflected Forms of a Lemma), 3) Grammar (grammatical characteristics of an inflected form). Therefore, NSU may be non-standard as a lemma, as a wordform, or from the point of view of its grammatical properties.

These types of NSU are to be exemplified.

2.1. Lemma-NSU

Obviously, this case is the most frequent, and exactly Lemma-NSU occur when we're speaking about non-standard units.

2.1.1. Proper Names

The proper names form the extensive class of linguistic units, which have some specific grammatical characteristic. The class includes toponyms and anthroponyms (first names, including pet names, family names, patronymics).

2.1.2. Abbreviations

The class includes acronyms and initialisms.

[1] The prehistory and the contemporary state of the RNC are available to be learnt about in (NKRJA, 2005, 2008).

[2] 'Arsis' (*plural* 'arses') is the special term in classical prosody for a syllable on which the ictus falls in a metrical foot.

2.1.3. New Loanwords

Russian is a language which is quite open to adoptions, so the stream of loanwords never subsides.

2.1.4. Obsolete and Rare Words

As it was mentioned above, the RNC contains the Corpus of Early Texts and the Corpus of Poetry, so it is quite natural that the quantity of dated, rare and stylistically marked words in the RNC is considerable.

2.1.5. Substandard Vocabulary

On the other hand, provided that the RNC includes a lot of patterns of modern literature, the Sub-Corpus of Electronic Communication, the patterns of private Spoken Russian, and the great amount of journalism, the presence of the substandard words and word combinations in the Corpus is of no wonder.

2.1.6. How We Solve the Problems

The only way to deal properly with the Lemma-NSU, listed in the §§2.1.1–2.1.5, is to expand the RNC Vocabulary regularly. The sources of the expanding may be as follows.

- **Existing dictionaries of Russian.** Substantially, the problems, named in §§2.1.1, 2.1.3–2.1.5, were solved by means of some Russian dictionaries – dictionaries of proper names (Petrovsky, 1996; Gramm (supplement), 2003), of loanwords (SIS, 2008), of substandard words (Elistratov, 2000). The word-lists of the fundamental dictionaries of Russian (Ortho, 2005; Gramm, 2003) also were of use. Naturally, it would be more convenient to use for these purposes the Russian Ontologies, as it is often done when NERC (Named-Entity Recognition and Classification) is used (see (Nadeau, 2007) among others). But the fact is that the generation of the Russian ontologies is now at its starting stage (see, e. g. (Nevzorova at al., 2007), and, above all, the RNC itself may be used as one of the source of the required ontology.
- **The list of wordforms, which lack grammar.** It is widely known, that the language development always surpasses the dictionaries development. Therefore, the Vocabulary of the RNC itself is the reliable source for expanding of the RNC: we ought to make the RNC profile after each completion of the Corpus and then to expand the RNC Vocabulary. For example, contemporary Russian lacks a dictionary of abbreviations, so the only way to solve the problem of abbreviations (see §2.1.2) is to address to the RNC itself.
- **The regular derivation.** It takes place when one of the groups of Lemma-NSU are the derivatives of the other, meaning that the second group is generated from the first one according to the set of regular rules. For example, Russian patronymics are the derivatives of the Russian names, and the list of the derivative rules is relatively short and clear. Therefore, to expand the list of patronymics in the RNC Vocabulary (see §2.1.1), there is no need to refer to dictionaries: we can receive the necessary units, using the dictionary of the Russian first names (for example, (Petrovsky, 1996; Gramm (supplement), 2003)) and the set of derivation programs.

The process of the expanding consists not only of the determination of the list of the supplementary lemmas, but also of their grammaticalization: we ought to adjust a lemma to its grammatical characteristics (the grammatical index, which describes a lemma's type of paradigm and other grammatical parameters).

Naturally, the main body of the grammaticalization of the new entries is carried out with the grammar parser[3]. But the grammatical structure of Russian doesn't permit us to rely totally on the work of the parser: we need to check the results carefully by ourselves.

For example, the parser never knows, how to mark the name of the town *Suzdal'*: it may be masculine or feminine, and this difference leads to the difference in the case endings. Therefore, only a linguist can choose the right gender (masculine, in this case). Naturally, the toponym *Suzdal'* is quite frequent and well-known, so it is included in all Russian toponymic dictionaries. But we face far more difficult cases, e. g. the town name *Ivdel'*, which is usually masculine, but the natives very often decline it as feminine. The toponymic dictionaries miss this place name, therefore, here we need a linguist to solve the problem.

Only a linguist can distinguish the family name *Kurchatov* from the toponym *Kurchatov*: their paradigms differ in several position (for example, InstrSg *Kurchatovym* vs. *Kurchatovom*).

Only in manual mode the grammar characteristics of new adoptions can be defined: for example, *praimeriz* (< Eng. primaries) or *g'orlz* (< Eng. girls): the parser automatically marks them like 'masculine Singular, *o*-declension', when really they are 'Plural indeclinable'.

A priori it is not known, what are the numeral and gender characteristics of an abbreviation. For example, the initialisms of the similar literal structure may have different set of grammatical characteristics: *CRU* = SingNeu, *SBU* = SingFem, *SŠA* = Plur, etc.

2.1.7. Graphic Corruptions

The Corpus of Electronic Communication, the Corpus of Early Texts and the Corpus of Poetry raise the question of the graphic and orthographical corruptions.

1) The text may be **corrupted orthographically**, when an author reflects in his spelling the way of his own pronunciation or when there is a misprint.

The simple way to solve the problem of orthographical corruption is to use automatic speller, when a text is being prepared to be included in the RNC. The programs of the kind may automatically change a wrong spelling into a right one (when we deal with the unambiguous correlation between the wrong spelling and the right spelling)[4].

But the fact is that there is a particular class of the RNC users, who doesn't want us to correct wrong spellings. Firstly, these are the specialists in Russian orthography. They investigate the contemporary tendencies in Russian

---

[3] The RNC uses two grammar parsers: DiaLing (in the Disambiguated Sub-Corpus) and MyStem (in the Main Corpus). Both parsers are PoS-taggers, not syntactical ones: the fact is that the morphology of Russian gives a user the possibility to solve a lot of the problems of Russian syntax.

[4] Therefore, nowadays we may mirror the early days situation: formerly, a corpus was the testing area for a speller (see, among others, (Chae Young-Soog, 1998), today, vice-versa, we may use a speller to improve a corpus.

spelling to elaborate timely recommendations for the change of the set of the Russian orthographical rules. Naturally, they want to know, what is the real orthographical practice. On the other hand, the investigators of the old Russian texts (for example, the texts of the 18 century) are specially interested in orthographical practice of the period. Therefore, their scientific aspirations would be violated, if the creators of the RNC automatically correct the old orthography and make it standard.
Therefore, in this case we have the only way to solve the problem: an inaccurate spelling ought to be preserved in order that any concerned user can find it in a text. On the other hand, that very incorrectness must be supplied with the correct form. This process may be accomplished automatically, with the help of speller, or in manual mode. Anyway, as a final result we receive an annotation of the following structure:

$$Inc_{\text{correctness}}\{Cor_{\text{rect}}Sp_{\text{elling}} + Gram_{\text{mar Characteristics}} + Sem_{\text{antic Characteristic}}\}.$$

Inc is a usual unit of the text, so it may respond to a usual query, as any other word in the Corpus; *CorSp* is the virtual correct unit, which corresponds to *Inc*, and also may respond to a usual query (but CorSp is invisible for a user, so it's presence in a text is virtual); *Gram* and *Sem* are the grammatical and the semantic characteristics of the *CorSp* respectively.
This way to annotate NSU is called **preserving annotation**. The preserving annotation gives us the following possibilities: 1) to find all contexts, which contain that very *Inc*; 2) to find all possible contexts, which contain that very *CorSp*, including the contexts with corresponding *Inc*; 3) to find all possible contexts, containing *CorSp*, which shows itself as *Inc*; 4) to find all possible contexts, which contain that very *Gramm*, including the contexts with *Inc,* the contexts with *CorSp*, or both; 5) to find all possible contexts, which contain that very *Sem*, including the contexts with *Inc,* the contexts with *CorSp*, or both.
Naturally, the preserving annotation is used in manual mode when the annotated texts are of exclusive importance (for example, some texts of 18 century, of electronic communication, or scripts of Spoken Russian). Also the preserving annotation is used, when we deal with the Educational Sub-Corpus of RNC (designed for school teaching) and with the Disambiguated Sub-Corpus (its volume is about 5 million tokens), where the grammatical homonymy was disambiguated in manual mode.
In other cases we may use the **orthoparser**, which would 1) mark *Inc*[5] in analyzed text, 2) automatically generate the set of corresponding *CorSp*, and 3) assign to *CorSp* necessary *Gramm* and *Sem*. For the Russian this problem may be solved on the basis of the existing spellers at an early date.

2) When the author deliberately enciphers his text, we deal with **graphical corruptions**. The reason for graphical corruptions are multifarious, but the main ones are as follows: 1) netspeak, 2) lazy, and 3) enciphering proper.

[5] Naturally, the very frequent cases of the wrong usage of one correct wordform instead of another correct form, for example, the wrong usage of the InstrSing of *o*-nouns instead of GenPlur and vice-versa, which takes place in Russian texts very often (e.g., *voronom* instead of *voronov*), may be corrected only in manual mode.

As the examples of the **netspeak** graphical corruptions we may consider so called *yazyk padonkaff* 'the language of scums', or, in other terms, *olbanskiy yazyk* 'the Albanian language'. It's mainly used in Internet, and its main characteristics is the deliberate distortion of the literal cover of a word and the ostentatious contrast to the common orthography. For example: the spelling *padonkaff* instead of the normative *podonkov*, the spelling *olbanskiy* instead of *albaskiy,* and so on.
The **lazy** reason to use graphical corruptions is to save writer's efforts and time (see, for example, regular way of writing *sobsno* instead of *sobstvenno* 'actually', *scha* (Russian *щa*) instead of *seychas* 'now', and *2* (*to*) and *4* (*for*) in English.
In Russian Internet communication the **enciphering** graphical corruptions is widely used. It's due to the strong policy of moderating and to the implied automatic elimination of taboo words. It is widely known, that Russian is a language with the ramified system of taboo-words, which are used with pleasure by the visitors of the Internet forums. So, to trick an automatic moderator, the visitors use the enciphered corruptions. For example, a) the spelling *3,14zdet'* instead of ***pi**zdet'* [taboo for 'talk nonsense, gabble', the word itself is included in the RNC Vocabulary], because 3,14 is the value of *pi*(π); b) the spelling *2,72bat'* instead of ***e**bat'* [taboo for 'make love with smb', the word itself is included in the RNC Vocabulary], because 2,72 is the value of *e*, the base of natural logarithm; c) the spelling @bat' (see previous point), because @ is the symbol of *e*-mail. And so on.
Due to the small number of graphical corruptions and because of their fluctuation and instability we can't treat them automatically. Moreover, we mustn't add them to the RNC Vocabulary. So, the only way to transform all these NSU is the preserving annotation (see above), started up in manual mode. Such mode is widely used to edit the texts of Internet communication to expand the RNC.

2.1.8. Word-Forming Doublets

Some derivative units in Russian are duplicated with slight differences in phonetics, grammar and stylistics. This doublets always have the same etymological history. The dictionaries of Russian never include all these doublets – in most cases they contain only one of the twins, so the other is treated in the dictionaries as a bastard and therefore is never marked and annotated in the RNC.

2.1.9. The Open Sets of Derivates

This class of linguistic units is quite a problem for any dictionary. The RNC Vocabulary is no exception. All the derivatives, which are created according to any productive word-formative model, can't be included in a dictionary. The lexicographers usually focus on the frequency of this or that derivative, and take their decision, concerning just these very characteristics of a unit. But this way to solve the problem is of no use for our purposes, because our aim is not to eliminate a rare derivative, but to minimize the quantity of tokens, lacking grammatical characteristics, in the RNC.

2.1.10. How We solve the Problems

To solve the problems, addressed in the §§2.1.8-2.1.9, we may act in two ways. First, we may try to expand the

RNC Vocabulary as much as possible, including all new derivates and doublets in it. This way is simple, but inefficient: any new text, included in the RNC, may add a derivative or a doublet, which has been absent in the Corpus until now. The openness is the main feature of productive derivative sets.

Therefore, the only way to solve the problem is to provide a special module as a component of the RNC grammar parser, which would generate the necessary units automatically, regardless of their real presence in the RNC. For example, every Russian noun with the suffix *-enije*, originating to a verb (*gorenije* 'burning' < *goret'* 'to burn', *burenuje* 'drilling' < *burit'* 'to drill', and so on, the set is open), has a doublet with the suffix *-en'je* (*goren'je, buren'j*). In most cases their meaning are absolutely equal, let alone the slight stylistic differences, and they have two regular differences in grammar: in LocSing the nouns with *-enije* have the ending *-i* and the nouns with *-en'je* have *-e*, and in GenPlur the noun with *-en'je* change their stems and receive the vowel *-i-* (not *goren'j,* but *gorenij*). Naturally, to treat these units, there is no need to include in the RNC Vocabulary all possible pairs – it's enough to include one of them and to generate the doublet automatically.

The acute problem of the kind is the open set of composites with the same first or second part of a word, for example, *videoplejer* 'video cassette player', *videoinzhener* 'video engineer', *videopirat* 'video pirate' and so on. It's quite obvious that it is absolutely impossible to list all the derivates of the kind, because the association of two parts in this case is absolutely free and agglutinative, and the first part of such composites may combine with all nouns or/and adjectives. Therefore, in such case we ought to create the module in the RNC grammar parser, which can 1) divide the composites of the kind into two parts (one of them – non-variable, the other – variable one), 2) check, whether the variable part of the composites is present in the RNC Vocabulary (that is, whether it functions as a self-reliant unit), and if so, then 3) consider the composite of this structure as a word-combination, not a word. In this case, there is no need to bother, whether this or that composite is present in the RNC Vocabulary – it's enough to make sure that the RNC Vocabulary contains the first and the second parts separately.

2.2. Wordform-NSU

This type of non-standard Russian takes place, when the RNC Vocabulary includes the right lemma and the right set of necessary grammatical characteristics, but one-to-one correspondence between the lemma and the set of its grammatical features collapses because of the grammatical peculiarity of a certain wordform.

This case takes place when any substandard wordform is used in a text. For example, in standard Russian the only normative GenPlur of the lemma *yablok-o* 'apple' is *yablok-Ø*. This wordform is fixed in all dictionaries of Russian. Therefore, when the grammar parser of the RNC meets in a text GenPlur *yablok-ov*, it leaves this bastard form with no annotation. The same situation takes place when the vowel-variant of the passive/middle verbal postfix *-s'a* is used after a vowel, in place of the correct consonant-variant *-s',* for example, *mol'us'a* 'I'm praying' instead of standard *mol'us'*. So, in these cases the wordform adopts the alien ending/morpheme.

The other case of the kind is the usage of the correct, but rather rare variant of an ending or of a morpheme. For example, the standard InstrSingFem of the nouns of *i*-declension is *-ju* (*yarost'-ju* < *yarost'* 'fury'). But in the old and poetic texts it is very easy to find variants with the ending *-iju* (*yarost'-iju*). And this duality is regular. The RNC parser now leaves the wordforms like *yarostiju* without any annotation, that is it treats them like non-standard. But it's a mistake – these wordforms are quite standard in the specific contexts and in the specific types of texts.

To surmount the grammar obstacle of the kind we have to improve the grammar basis of our grammar parser. First, we ought to list all the similar problem points of Russian Grammar (for example, in the Russian nominal paradigm the problem points obviously are GenSingMasc, LocSingMasc, NomPlurMasc, GenPlurMasc, InstrSingFem, and so on). For all these points we must allow all possible oscillations of the standards, for example, we must recognize the endings GenPlur *-ov* and *-Ø* for the nouns like *yabloko* (Neuter Inanimate with the velar as the last consonant of a stem) as equal in rights. Naturally, in this transformed grammar parser the endings *-ju* and *-iju* for the nouns like *yarost'* would have equal value.

2.3. Grammar-NSU

This is the rarest case, when we face the omission in the grammar of the Corpus: the grammatical phenomenon has already been described in Russian scientific grammars, but the creators of the grammar parser (MyStem) have evaluated this phenomenon as too specific, and so have been mistaken. We mean the so called new Vocative in Russian. The forms like *mam!* (< *mama* 'mummy'), *pap!* (< *papa* 'daddy'), *Kol'!* (< *Kol'a* 'pet name for *Nikolay*'), *Mash!* (< *Masha* 'pet name for Mary'), and so on are extremely popular in the informal Russian. Just now we can find the Vocative only in the Disambiguated Sub-Corpus of the RNC, where the forms of the kind had been disambiguated in manual mode. In the rest of the RNC the vocative forms are grammaticalized incorrectly.

Therefore, to solve the problem we ought to transform the grammar parser in the following way. For all the Russian anthroponyms of the *a*-declension (i.e. for all names, having the ending *-a* in NomSing) and for some appellatives of the same grammatical structure the parser must suppose one more grammar case, the Vocative. If a name doesn't include any suffix, containing an unstable vowel, then the Vocative of the name coincides with the GenPlur ($Masha_{\text{Voc}}$=$Masha_{\text{GenPlur}}$=*Mash*). If a name includes a suffix, containing the unstable vowels *-o-/-e-*, then the Vocative of the name coincides with the NomSing, but the last vowel *-a* is cut off ($Mashka_{\text{Voc}}$= $Mashka_{\text{NomSing}}$ – *-a* = *Mashk*[6]).

2.4. NSU in Spoken and Accentological Corpora

The very specific types of NSU are presented in the two very important sub-corpora of the RNC – the Spoken Corpus[7] and the Accentological Corpus. These corpora include the linguistic objects which usually are not reflected in the written Russian. They are the stress and a lot

[6] The GenPlur of the names of such structure would include the unstable vowel: *Mashek*.

[7] About the Spoken Corpus of the RNC see (Grishina, 2006).

of peculiarities of oral Russian speech[8]. It's widely known that an ordinary written Russian text isn't accentuated and it doesn't contain the linguistic units, which are highly frequent in the spoken Russian. If that's the case, then the RNC grammar parser, which has been expected to deal only with the written Russian, would be mistaken dealing with the spoken or accentuated Russian.

If so, then we ought to upgrade the parser and to teach it to take into account the peculiarities of the spoken Russian. These are the examples.

1) In Russian there are a lot of grammatical forms which are distinguished only with the help of stress. For example, *polúchite* 'you will receive' vs. *poluchíte* 'Receive it!', *kátera* 'boat, GenSing' vs. *katerá* 'boat, NomPlur', *Lídok* 'Lidka, informal for the name Lidia, GenPlur' vs. *Lidók* 'one of the pet names for Lidia, NomSing', *ó* '1) the name of the letter O, 2) one of the variant of the particle *vot*, 3) the interjection *o!*' vs. *o* 'the preposition *about*', and so on.

In all these cases the grammar parser must distinguish the forms and lexemes with/without stress and also the different positions of the stress in a word.

2) The Spoken Russian texts contains the multifarious set of the contracted (simplified) and apocopic forms of the words. For example, the very frequent in spoken Russian words like *seychas* 'now', *tys'acha* 'thousand' have the standard simplified oral variants *schas, scha, tyscha*. Naturally, this kind of pronunciation is reflected in the scripts, because it's very important to distinguish, when a speaker uses the full pronunciation and when he let himself use the simplified forms[9].

On the other hand, each unchangeable word in Russian, which has the vowel ending, may lose this vowel without any alteration of the meaning of the word. These forms without the ending vowel are called apocopic and are very frequent in the spoken Russian (some of them have even become literary variants, for example, *chtoby /chtob* 'in order to'[10].). The scripts of the spoken Russian contain the great amount of the apocopic forms.

Finally it is necessary to mention the reflection of the dialect peculiarity, which takes place in the scripts of the spoken Russian, for example, the reflection of the Caucasian accent (*sl'ushay* instead of *slushay* 'listen', and so on). These phenomena are extremely frequent in the Dialectal Corpus of the RNC.

All these simplified, apocopic and dialectal variants are now misadjusted: some of them are skipped over with the grammar parser and don't receive any grammar characteristics, some of them receive wrong grammatical qualities (for example, the apocopic variant of the particle *pr'am* – instead of *pr'amo* 'real, really, exactly' – is analyzed as the NomSing of the short form of the adjective *pr'amoj* 'straight').

To solve the problems, concerned in this section, we must act in different ways. When a phenomenon is rare and sporadic, or when it takes place only in the highly specific types of texts (for example, in the dialectal texts), the only way to grammaticalize these bastard forms is the preserving annotation, started in manual mode (see §2.1.7). When the phenomena are frequent and regular, we ought to provide their presence in the RNC Vocabulary. For example, the frequent apocopic and simplified form are to be included in the Vocabulary. Similarly, the regular contracted forms of the Russian masculine patronymic ought to be included in the Vocabulary as the variants of the full forms: the oral variant *Ivanych < Ivanovich, Nikolaich < Nikolajevich* 'the son of Ivan or Nikolay' can be obtained in the course of the regular transformations of the initial patronymics (*-ov- → Ø, -i- → -y-* OR *-jev- → Ø*).

3. Conclusion

Thus, we may summarize the previous description and review all the types of the normalization of the NSU in the RNC.

1. **Expanding of the RNC Vocabulary,** sources:
   - **Existing dictionaries** [Proper names (2.1.1); Loanwords (2.1.3); Obsolete and rare words (2.1.4); Substandard vocabulary (2.1.5); Simplified and apocopic form (2.4)]
   - **1.2. RNC itself** [Abbreviations (2.1.2)]
   - **1.3. Regular derivation** [Russian patronymics (standard form) (2.1.6)]

2. Improvement of the RNC grammar parser
   - **2.1. Creating of the special analyzing module** [Word-forming doublets (2.1.8, 2.1.10); Open sets of derivatives (2.1.9, 2.1.10); Russian patronymics (informal forms) (2.4)]
   - **2.2. Reforming of the grammar parser** [Wordform-NSU (2.2); Grammar-NSU (2.3)]

3. Preserving annotation
   - **In manual mode** [Orthographical corruptions (2.1.7); Graphical corruptions (2.1.7); Spoken and accentological NSU (2.4)]
   - **Orthoparser** [Orthographical corruptions (2.1.7)]

Acknowledgements

The work of the RNC group is supported by the program "Philology and Informatics" of Russian Academy of Sciences. The author's investigation is supported by the RFBR (RFFI) under the grants 06-06-80133a and 08-06-00371a.

References

Chae Young-Soog (1998). Improvement of Korean Proofreading System Using Corpus and Collocation Rules / Language, Information and Computation (PACLIC12), 18-20 Feb. P. 328-333

Crowdy, S. (1994). Spoken Corpus Transcription // Literary and Linguistic Computing. 9(1). P. 25-28;

Elistratov, V.S. (2000). Slovar' russkogo argo. M.

Gramm (supplement) (2003) – Zalizn'ak, A.A. Grammaticheskij slovar' russkogo jazyka. Prilozhenie: Imena sobstvennye. M.

Gramm (2003) – Zalizn'ak, A.A. Grammaticheskij slovar' russkogo jazyka. M.

Grishina, E. (2006). Spoken Russian in the Russian National Corpus (RNC) // LREC2006: 5th International Conference on Language Resources and Evaluation. ELRA, 2006. P. 121-124

[8] About the peculiarities of a spoken speech and their reflection in the written texts see (Panunzi, 2004).

[9] The creators of the BNC were the first to list the basic difficulties of the orthographical record of a spoken speech and the main types of the discrepancies between 'written' written speech and 'spoken' written speech (see, for example, (Crowdy, 1994)).

[10] About the usage of the apocopic forms in the spoken Russian see (Grishina, 2007).

Grishina, E. (2007). Text Navigators in Spoken Russian. // Proceedings of the workshop “Representation of Semantic Structure of Spoken Speech” (CAEPIA’2007, Spain, 2007, 12-16.11.07, Salamanca), Salamanca. P. 39-51.

Nadeau, D. (2007). Semi-Supervised Named Entity Recognition: Learning to Recognize 100 Entity Types with Little Supervision. Thesis submitted to the Faculty of Graduate and Postdoctoral Studies in partial fulfillment of the requirements for the PhD degree in Computer Science Ottawa-Carleton Institute for Computer Science School of Information Technology and Engineering University of Ottawa. Ottawa.

Nevzorova, O., Nevzorov V., Zin’kina, Ju., P’atkin, N. Integral’naya Tekhnologiya Razresheniya Omonimii v Sisteme Analiza Tekstovykh Dokumentov LOTA // Kompjuternaya Lingvistika i Intellektual’nyye Tekhnologii. Trudy Mezhdunarodnoj Konferencii “Dialog 2007”. Bekasovo, Rossiya. P. 422-427.

NKRJA (2005). Natsional’nyj korpus russkogo yazyka: 2003–2005. Rezul’taty i perspektivy. M.

NKRJA (2008). Natsional’nyj korpus russkogo yazyka: 2006–2008. Rezul’taty i perspektivy. (forthcoming)

Ortho (2005). Russkij orfograficheskij slovar’. M.

Panunzi, A., and al. Using PiTagger for Lemmatization and PoS Tagging of a Spontaneous Speech Corpus: C-Oral-Rom Italian // LREC2004: 4th International Conference On Language Resources And Evaluation. May 2004. Lisbon, Portugal

Petrovsky, N.A. (1996). Slovar’ russkih lichnyh imen. M.

SIS (2008). Grishina, E.A. Novejshij illustrirovannyj slovar’ inostrannyh slov. M. (forthcoming)

Creating a Parallel Treebank of the Old Indo-European Bible Translations[1]

Dag T. T. Haug, Marius L. Jøhndal

University of Oslo
daghaug@ifikk.uio.no, mlj32@cantab.net

Abstract

In this paper, we report on the creation of a syntactic annotation scheme to be used in a comparative study of the oldest extant versions of the New Testament in Indo-European languages: the Greek original, as well as the translations into Latin, Gothic, Armenian and Church Slavonic. The tag set was created in a pilot study involving syntactic annotation of the Gospel of Mark in Greek and Latin. The resulting tag set is well suited for capturing syntactic variation between these languages, particularly in areas having to do with pragmatics and information structure – as the treebank is created within a larger project in this field – but also more general syntactic differences.

1. Introduction

The project *Pragmatic Resources of Old Indo-European Languages* (PROIEL) aims to study the linguistic means of information structuring which are offered by the grammar of Greek, Latin, Armenian, Gothic and Church Slavonic, i.e. the means that the lexicon and the syntax of these languages make available[2] for expressing such categories as old and new information, contrast, parallelism, topicality and others. Five particular phenomena will be examined in the PROIEL project:

- Word order
- The definite article
- Discourse particles
- Anaphoric expressions, including zero anaphora
- Participles and absolutive constructions

These topics were chosen because they are known to be important in information structure systems cross-linguistically and because they are areas where the languages in the corpus are likely to diverge. For example, Ancient Greek is the only language in the corpus to have a grammaticalized definite article. This language is also well known for its abundance of discourse particles, which cannot be rendered directly in the target languages. Word order is notoriously free in these languages, and while this led to direct adoption of the Greek word order in many cases, there are still patterns that cannot be rendered directly. Similarly, the anaphoric and participial systems vary widely.

The most important objective for our treebank is to be able to represent these phenomena correctly with as fine-grained information as possible. On the other hand, it is likely that in the course of the project, we will find other phenomena that are relevant to the general topic of information structure, so we need to be prepared to adapt our scheme to changing requirements. Finally, it is important that the treebank is created in such a way as to be useful for a wider audience, no matter what topics they are interested in. The annotation scheme must therefore be suitable for representing the general structure of sentences in these languages.

It was decided that rather than focussing on creating coherent data from the very start of the annotation process, the best way to accomplish our objectives was to annotate a pilot text while we were developing the annotation scheme. This way we could maximize the value of feedback from annotators, gain experience with the annotation process itself, and have a readily available testbed during development of the software. The remainder of this paper describes this process and its outcome.

2. Preparing the pilot text and creating the annotation tools

Building a treebank is labour-intensive, so our initial concerns were to avoid duplication of efforts and to get our annotators started as quickly as possible. This was greatly facilitated by the availability of a morphologically annotated electronic version of the Greek New Testament (Sandborg-Petersen, 2008) and by the work done by the Perseus digital library (Crane, 1987; Crane et al., 2001) on their electronic version of Jerome's Vulgate and word-lists for Latin and Greek.

We used these resources to prepare the text for the pilot study. This text consists of the Greek and Latin versions of the Gospel of Mark – which in each language amounts to roughly 13,000 words or 10% of the complete New Testament.

Due to the complexity of Biblical textual criticism, and since the purpose of the overarching project is to do a cross-linguistic comparative study, we chose to ignore manuscript variants. Our texts are instead based on the text of a specific edition, and we only correct digitization errors, should these occur.

The preparation and annotation of the pilot text proceeded in four stages:

- Pre-processing
- Automated morphological tagging
- Manual annotation by annotators
- Manual review by a reviewer

The pre-processing stage involved segmentation, detection of sentence boundaries and sentence alignment. Segmentation is occasionally problematic as certain morphemes behave as separate entities in the syntactic model we use, but

[1]The research project described here is funded by the *Norwegian Research Council*'s YFF program, grant no. 180632. We thankfully acknowledge this support. The glossing in this paper follows the Leipzig Glossing Rules (`http://www.eva.mpg.de/lingua/resources/glossing-rules.php`).

[2]Prosody presumably also played a role which is, however, irrevocably lost for us.

form part of other words. This happens, for example, with instances of *krasis*[3] in Greek. A trickier case is presented in Latin where enclitics like *-que* cannot always be tokenized unambiguously.

For detecting sentence boundaries, we decided to use punctuation as a guide, as the canonical division into verses frequently fails to match what we consider to be meaningful syntactic units. A particular problem was presented in our Vulgate text as it lacks punctuation altogether. To solve this, the punctuation from a different electronically available edition, the Clementine Vulgate, was merged into our text by first aligning the orthography of the two editions, then using an implementation of the standard *diff* algorithm (Hunt and McIlroy, 1976) and finally patching the target text using only chunks that involved insertion of punctuation. This simple approach gave good results in spite of numerous textual differences, and only occasionally resulted in off-by-one errors in sentence division.

To answer the questions posed in the research project, corpus users will eventually need to be able to query the same sentence in multiple languages. This requires sentences to be aligned, and our experiments indicate that Gale and Church (1993)'s algorithm performs this task well with chapters as hard delimiters and sentence boundaries as soft delimiters.

As ready-made word-form lists were available, morphological tagging was done simply as an assignment of one or more pairs of lemma and morphological tags to each token in the text. The morphological tag set that we developed is more fine-grained than those of our source data, so for many tokens the level of detail of the assigned morphological tags was insufficient. We were able to address this by manually adding assignment rules, and harvesting additional rules from data already disambiguated by annotators. The morphological tags are positional tags and derived from the system used by the Latin Dependency Treebank (Bamman and Crane, 2006). For the purposes of the PROIEL project, more fine-grained distinctions had to be made for certain parts of speech, in particular pronouns, whose functions are primarily pragmatic. We have also deviated from the traditional grammarian's view of morphology and adopted a more 'modern' view by treating adverbs that double as prepositions as intransitive prepositions, and by merging the two categories particle and adverb (see table 1).

It has furthermore been necessary to introduce a means for indicating ambiguity of form or unresolvable syncretism, e.g. for nouns that alternate between genders. This has been solved by allowing multiple inheritance within each field of a positional tag, so that, for example, the tag for masculine gender has a super-tag that indicates masculine or neuter gender, one that indicates masculine or feminine gender etc.

The two manual stages of the process – annotation and review – were performed using variants of the same graphical interface. We wanted an interface that students could use on

Major part of speech	**Minor part of speech**
Verb	
Noun	declinable common noun indeclinable common noun declinable proper noun indeclinable proper noun
Pronoun	relative pronoun interrogative pronoun indefinite pronoun demonstrative pronoun personal pronoun possessive pronoun personal reflexive pronoun possessive reflexive pronoun reciprocal pronoun
Numeral	declinable cardinal number indeclinable cardinal number ordinal number
Adjective	
Article	
Adverb	comparable adverb relative adverb interrogative adverb other non-comparable adverb
Conjunction	
Subjunction	
Preposition	
Interjection	
Foreign word	

Table 1: The parts of speech defined in the PROIEL morphological tag set.

typical campus workstations which frequently have a limited choice of installed software and restrict students' rights to run local, stand-alone applications. We therefore chose to develop a light-weight web-based interface that would function with only a modern browser and client-side scripting.

The interface is designed as an incremental 'wizard' that splits the annotation into three steps. First, annotators verify and, if necessary, adjust sentence boundaries. We have found that this ought to be restricted as annotators felt tempted to override the judgements of the text editors and therefore did excessive adjustments of sentence boundaries. This may be due to the style of our texts in which sentences tend to 'run together' and a large number of sentences are introduced by *and*. The choice between coordinating a main clause with the preceding clause or not is thus often an arbitrary one. We therefore let annotators adjust the sentence boundaries only one token at a time so that they could only correct off-by-one errors.

The second step of the 'wizard' involves morphological disambiguation. Annotators are presented with the output from the automated morphological tagging and are asked to choose the correct lemma and tag pair in cases of ambiguity. Finally, in the third step, annotators build depend-

[3] The term *krasis* refers to a contraction in which the final vowel of one word coalesces with the initial vowel of the next, and the two words are written together.

ency structures for each sentence. This is done visually and guided by a simple rule-based 'guesser' that suggests the most likely dependency relation based on the morphology of head and dependent.

In addition to the interface for annotation, and an interface for text browsing, we added functionality for tracking change history and for inserting cross-references to other information sources such as dictionaries. In particular, we have made use of *Strong's Concordance* (Strong, 1890) and the *Analytical Lexicon of the Greek New Testament* (Friberg et al., 2000), as these were the basis for lemmatization in our Greek text.

The system is based on Ruby on Rails with a database backend. As a by-product of this choice, the system offers not only a traditional web-interface to the corpus, but also exposes a RESTful XML interface that can be used by clients to query the database. This should facilitate interchange of data and direct reuse of our work in other contexts.

3. The development of the annotation scheme

As noted above, all the languages in our corpus have a 'free' word order, i.e. the word order does not indicate syntactic dependencies or grammatical functions, but serves pragmatic purposes. Therefore, while word order data are important for PROIEL, they cannot be conflated with information about grammatical function as is done in a phrase structure grammar. For this reason, it was decided to base the annotation scheme on dependency grammar (DG). This also had the advantage that other projects developing treebanks of Latin, e.g. the Latin Dependency Treebank (LDT), are based on DG, using a faithful adaption of the well-documented Prague Dependency Treebank (PDT) (Hajič, 1998).

We began our work using the Greek and Latin versions of the New Testament, since these exist publicly available in electronic form with morphological annotation. We expected the syntax of most old Indo-European languages to be sufficiently similar to be captured within a single annotation scheme and our experienced with the Greek and Latin texts have confirmed this. There are diverging constructions, of course, but they can all be captured using our primitive syntactic relations, and we do not expect Gothic, Armenian or Church Slavonic to be different in this respect.

3.1. General presentation

While we wanted to keep the option to automatically convert our treebank to a more general format, we soon realised that the level of granularity of the PDT annotation scheme or the LDT annotation scheme (Bamman et al., 2007) would not be sufficient for PROIEL. Table 2 shows the general outline of our annotation scheme in comparison with that used by LDT. It is more fine-grained than the LDT scheme, both in the domain of verbal arguments and that of adnominal functions. To study the interaction between syntax/argument structure and pragmatics in determining word order, we need to be able to separate objects (OBJ) from other arguments of the verb (OBL). Furthermore, agent expressions (AG) are particularly interesting for the syntax-pragmatics interface, because they are both optional and receivers of a thematic role from the verb. [4]

In the adnominal domain, it is well known that there are interesting correlations between types of genitives and information structure. For example, possessive genitives tend to be old information in a text and are typically used to access new referents, whereas object genitives are more often new information. Partitive genitives are special as they, and not their syntactic heads, introduce the discourse referent of a noun phrase: 'two of the disciples' refer to a group of disciples, and not to some kind of 'twoness', unlike 'the teaching of the disciples'. It is therefore essential for PROIEL to distinguish these uses of the genitive.

There is one notable exception to the general pattern that our tags are more fine-grained than those of the LDT; the LDT scheme provides 9 subtypes of auxiliary relations: AuxP for prepositions, AuxC for conjunctions, AuxR for the reflexive passive etc. In our opinion all items with the relation AuxX in the LDT can be conflated to a single relation as instances can still be differentiated based on lexical information when the need arises.

3.2. Granularity

By asking annotators to do fine-grained classification of the data, we run the risk of more inconsistencies in the application of the scheme. For this reason, we have introduced some 'super-tags', i.e. tags that we ask the annotators to use whenever they are in doubt. For example, it can be hard to tell whether a given relative clause is restrictive (ATR) or not (APOS). We provide a tag REL for such cases, so that the annotators do not simply guess.

However, in the case of adnominal tags, we purposefully did not provide any such super-tag, in order to test the viability of making distinctions within this domain. The results were mixed. In the beginning, we asked annotators simply to distinguish attributes and appositions. After a couple of weeks, we introduced more granularity by means of the tags PART, to be used for partitive expressions, and OBL, to be used whenever an expression is an *argument* of the noun – typically an object genitive as in *amor fati* 'love of faith'. OBL was chosen because this is the relation we use for non-object arguments in the verbal domain.

When the pilot was finished, we studied how annotators had used these tags. Although the results are not statistically significant, they were valuable in guiding our development of the annotation scheme. In general, the annotators coped well with the PART relation: of 42 uses of this relation, only 3 were wrong – not too bad a result at such an early stage in the annotators' training. Moreover, the errors could easily be detected automatically, since they did not involve any uses of PART with an adnominal genitive that should have had another relation, but rather the generalisation of PART to other contexts with partitive semantics, i.e. a genitive object and an object of the Greek preposition *apo*.

The concept of arguments of nouns was harder to apply. This relation was used 22 times for items dependent on

[4] The decision to include the AG relation, which combines syntactic function and semantic role, was a pragmatic choice motivated by the fact that we do not expect to have the resources to do a full tectogrammatical annotation as in the PDT.

Latin Dependency Treebank	PROIEL Corpus	Explanation
PRED	PRED	Main clause predicate
*	PRED	Subordinate clause predicate
SBJ	SUB	Subject
OBJ	OBJ OBL AG XOBJ	Object Oblique Agent Open complement clause
ADV	ADV	Adverbial
ATR	ATR NARG PART	Attribute Nominal argument Partitive
ATV	XADV	Free predicative
PNOM	XOBJ	Subject complement
OCOMP	XOBJ	Object complement
COORD	*	Coordinator
APOS	APOS	Apposition
AuxX (X defines the subtype of Aux)	Aux	Auxiliary
ExD	* VOC	External dependency Vocative

Table 2: Sentential functions in LDT and PROIEL. An asterisk in one of the columns indicates that the two annotation schemes diverge in some other way than by one simply being more specific than the other.

nouns, 6 times erroneously. Apparently the possibility of using OBL adnominally tempted annotators into analysing verbs with an object and a PP complement as if the prepositional phrase were dependent on the object, e.g. so that *super* is a dependent of *manus* in the participial construction

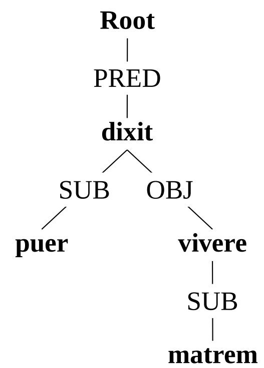
(1) *imponens* *manus* *super*
put.PRS.PTCP.NOM.SG hand.ACC.PL upon
illos
they.ACC.PL.
'laying his hands upon them'

Such errors cannot be detected automatically. Moreover, since the OBL tag is used in more contexts, we run the additional risk of contaminating the entire set of OBL-relations. Not only was OBL used in cases where another relation should have been used, there were also cases where PART and OBL were not used when they should have been. As part of our analysis of the data from the pilot annotation, we examined the 123 cases of genitive nouns dependent on another noun that had been annotated after the introduction of PART and OBL as adnominal tags. 17 of these were given an incorrect analysis, and in 16 cases this was because ATR was used when PART or OBL would have been correct.
The period of pilot annotation has taught us that it is difficult for annotators to distinguish different functions in the adnominal domain. Still we will continue to make these distinctions, but we no longer use the relation OBL, but rather a separate relation NARG (nominal argument) which is devoted to arguments of nouns. In this way, we have an 'exit strategy' in case the the quality of the annotation remains low, since we can merely convert all NARGs to ATRs. Also, since we have now had the opportunity to test the annotators' ability to make fine-grained distinctions in the adnominal domain, we have introduced a super-tag ADNOM so that we no longer force the annotators to choose when they are in doubt.

3.3. Dealing with covert elements

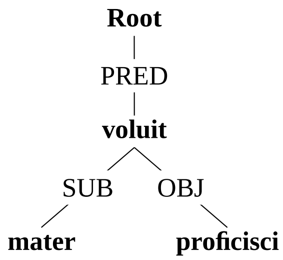
A dependency grammar is well equipped to deal with the free word order of ancient languages. However, it is less well adept at representing another feature typical of old Indo-European languages, namely ellipsis. The DG formalism has difficulties with all constructions without a clear syntactic head, e.g. ellipsis, coordination (and in particular asyndetic coordination) and sentences lacking a verb (most often the copula).
Different solutions have been devised to these problems; in the following we describe our solution, which tries to capture the facts in a theory-neutral manner.
First consider a less well-known problem for dependency grammar, namely ellipsed *dependents*. Ellipsis of dependents is much more frequent than ellipsis of heads and, while it is generally easier to deal with, it can sometimes lead to problems. Consider the treatment of the accusative with infinitive (AcI) (example 2) and the complement infinitive in the LDT (example 3):

(2)



puer *dixit* *matrem*
boy.NOM.SG say.3SG.PFV mother.ACC.SG.
vivere
live.PRS.INF
'The boy said his mother was alive.'

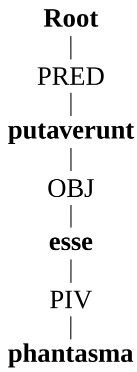
(3)



mater *voluit* *proficisci*
mother.NOM.SG. can.3.SG.PFV travel.PRS.INF
'Mother wanted to travel.'

The fact that we here have two different constructions is signalled only by the presence of a subject daughter in example 2. However, Latin being a pro-drop language,[5] this subject is optional:[6]

(4)



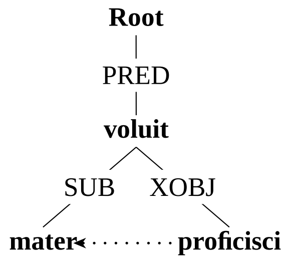
putaverunt *esse* *phantasma*
think.3.PL.PFV be.PRS.INF ghost.ACC.SG
'They thought [he] was a ghost.'

A search for all AcI structures would have to sacrifice precision (by matching all infinitive objects) or recall rate (by matching all infinitives that have a subject daughter). This problem is particularly important to PROIEL, since the subject of the infinitive in such examples as example 4 can only be left out because it is given information in the context.

At first we tried to solve the problem by *not* letting the verb stand in for the whole sentence, but rather let sentences (including AcIs) be represented by an empty node that dominated the verb and its arguments, so that the defining feature of these empty nodes was the possibility (but not necessity) of dominating a subject. However, this quickly leads to problems: the empty elements are hard to deal with computationally and result in an unmotivated distinction between verbs in full sentences, which would be sisters of their arguments, and verbs in participial constructions, which would dominate their arguments. Although this system provided an intuitive way of dealing with so-called 'gapping' (the absence of the verb in the second conjunct, see example 12), we quickly abandoned it.

Inspired by Lexical-functional grammar, we instead chose to represent the structural difference between infinitives in AcIs and complement infinitives as two contrasting relations, OBJ and XOBJ. The latter function is by definition one which cannot have an overt subject, but shares its subject with another element in the clause. We designate this structure-sharing by what we call 'slash notation'.[7] The full representation of example 3 is therefore:

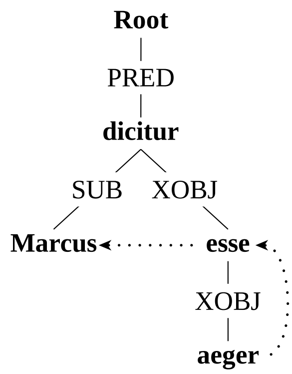
(5)



mater *voluit* *proficisci*
mother.NOM.SG. can.3.SG.PFV travel.PRS.INF
'Mother wanted to travel.'

The arrow in this example should be interpreted as a secondary dependency relation. In this case it shows that *mater* is the subject of both *proficisci* and *voluit*. This accounts for case agreement with predicate nominals in the dependent infinitive construction, as in the following example, which also shows how we deal with instances where the subject of the XOBJ is not overtly realized:

(6)



Marcus *dicitur* *aeger*
Marcus.NOM.SG say.3.SG.PRS.PASS ill.NOM.SG.
esse
esse.PRS.INF.
'Marcus is said to be ill.'

In traditional grammar the subject of *aeger* is supplied by the verb *esse*. We incorporate this by letting the slash arrow point to the head verb whenever it 'ought' to pointed to a 'pro-dropped' argument. This has the further advantage of

[5]The term *pro-drop language* refers to languages in which some pronouns may be omitted when they can be inferred pragmatically.

[6]In this tree, we use PIV for the traditional notion of 'predicative complement', which actually does not exist in our scheme; see below.

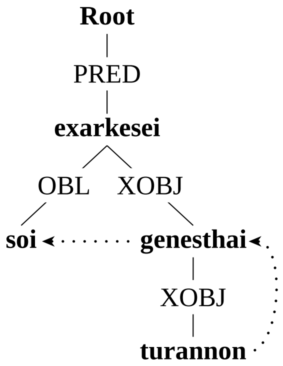
[7]This designation is in turn inspired by the vaguely similar SLASH-lists of Head-driven phrase structure grammar.

making it easier to validate annotations; we can enforce the principle that every XOBJ or XADV relation should have one slash arrow and that this arrow should point towards the head verb or an element dominated by the verb.

esse in turn gets its subject from *Marcus*. Notice that we treat the traditional category of predicative complement as XOBJ, seeing that the facts are the same: the element is subcategorized for by the verb and does not have a direct relation to its subject.

Our representation is neutral between *control* and *raising* analyses. Compare the example above to the following example:

(7)



exarkesei	*soi*	*genesthai*
suffice.3.SG.FUT	you.DAT	become.PFV.INF
turannon		
tyrant.ACC.SG		

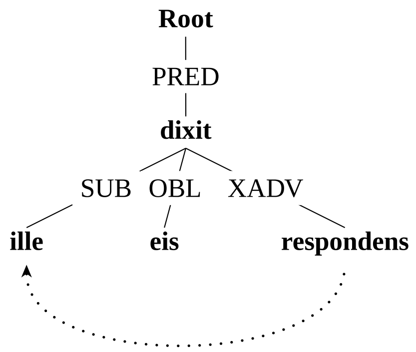
'It will suffice for you to become a tyrant.'

Many theories would treat example 6 as raising and example 7 as control, to explain the differences in case agreement. Our representations are meant only as input to such discussions, so we make no commitment regarding the status of our slash dependencies, nor do we ask our annotators to make decisions based on linguistic theory. The annotators should simply identify the subject of infinitives and predicative complements: if the subject of the XOBJ is not present in the subtree dominated by the governing verb, the slash arrow should point to the verb, which 'supplies the subject' according to traditional grammar.

3.4. Other uses of the slash notation

The slash notation was first introduced to separate AcI's from complement infinitives even in cases where the subject of the AcI has been 'pro-dropped'. It is a modest but powerful addition to our data-model that allows us to obtain a richer annotation for many structures. The slash notation most manifestly translates to infinite predications that are not arguments of the main verb, i.e. predicative (conjunct) participles:

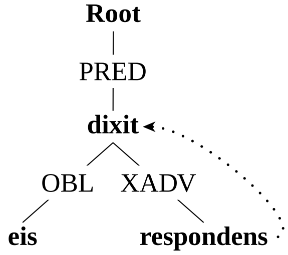
(8)



ille	*respondens*
he.NOM.SG	answer.PRS.PTC.NOM.SG
dixit	*eis*
say.3.SG.PFV	they.DAT.PL

'Answering them, he said'

(9)

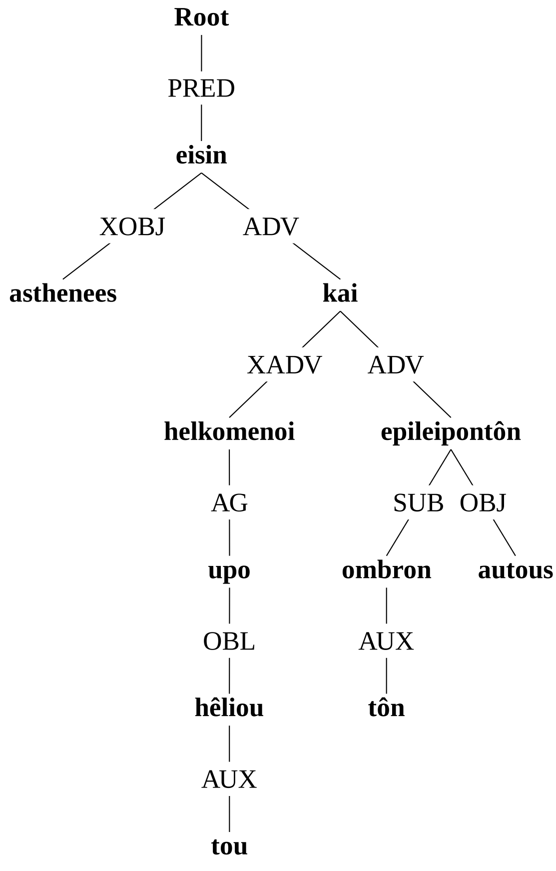


respondens	*dixit*
answer.PRS.PTC.NOM.SG	say.3.SG.PFV
eis	
they.DAT.PL	

Note that this preserves the structure even when the subject of the predicative participle is 'pro-dropped'. This is otherwise hard to achieve: If the participle were to depend on its subject, the structure would be different when the subject was omitted. We could have used complex tags to denote the subject of the participle, but then we would have had to deal with predicative participles that depend on main clause adjuncts, of which there may be several. The slash notation solves this by pointing to the subject of the participle and not to the relation which the subject of the participle has.

Dependency grammars generally speaking only allow coordination of elements related to the same head via the same relation. It is, however, not hard to find examples with a conjunct participle coordinated with an adverbial element, e.g. an absolute genitive:

(10)



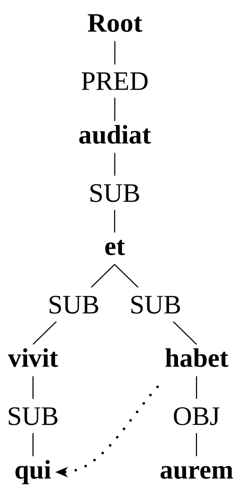
tôn *ombrôn*
the.GEN.PL rain.GEN.PL.
epileipontôn *autous* *kai*
leave behind.PRS.PTCP.GEN.PL they.ACC.PL and
upo *tou* *hêliou*
by the.GEN.SG sun.GEN.SG
helkomenoi *asthenees*
draw up.PRS.PTCP.NOM.PL weak.NOM.PL
eisi
be.3.PL.PRS

'With the rain leaving them being and drawn up by the sun, they [sc. the rivers] are weak.'

Our notation solves this problem. The daughter nodes of XADV relations *always* have a slash arrow, so the 'X' merely serves to indicate the presence of the slash.[8] This means that we can coordinate XADV and ADV without distorting the analysis. The advantage of our notation would become even clearer if the participle had an overt subject, as the two conjuncts in this case would have to have different heads in a 'classical' analysis.

Once introduced, the slash notation can be exploited for richer annotation of other structures that involve ellipsis or structure-sharing. Since the verb substitutes for the whole sentence, we treat coordination of two verbs as sentence coordination and use the slashes to indicate double dependencies (i.e. subject sharing):

(11)

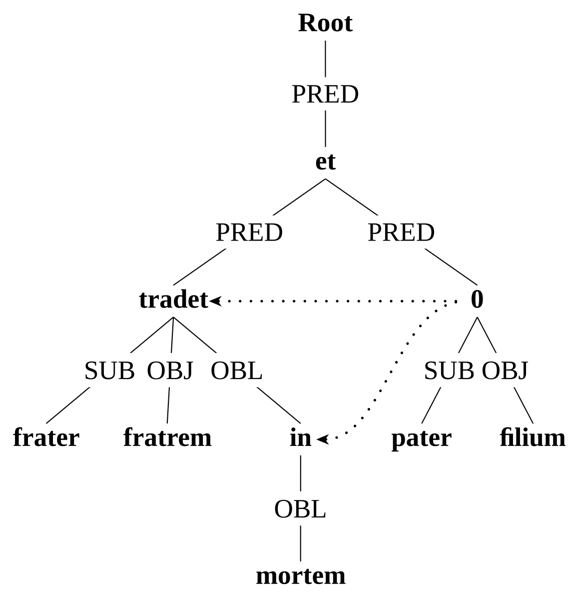


qui *vivit* *et* *aurem*
who.NOM.SG live.3.SG.PRS and ear.ACC.SG
habet *audiat*
have.3.SG.PRS hear.3.SG.PRS.SBJV

'Whoever lives and has ears shall hear.'

The advantage of this notation is evident in gapping constructions where the predicate is omitted in the second conjunct:

(12)



tradet *frater* *fratrem*
deliver.3.SG.PRS brother.NOM.SG brother.ACC.SG
in *mortem* *et* *pater*
to death.ACC.SG and father.NOM.SG
filium
son.ACC.SG

'The brother shall betray the brother to death, and the father the son.'

The combination of a restricted use of empty nodes and the slash notation makes it possible to preserve the structure of the tree. We also capture the fact that the argument *in mortem* is shared between the two conjuncts. The two slash arrows have rather different interpretation: the one from the empty node to the verb indicates sharing of lexical material, whereas the one from the empty node to the preposition *in* indicates a double dependency.

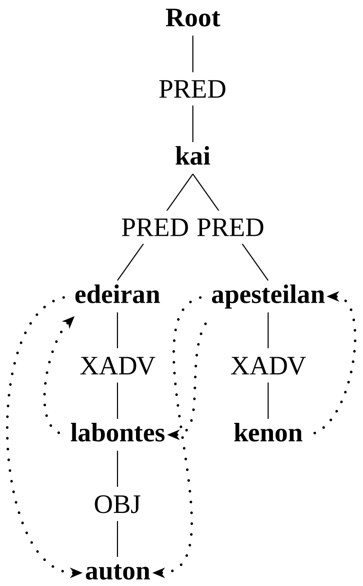
Since the slash relation is not labelled, it is important that the relation can be interpreted based on other information in the sentence. And, in fact, this remains possible. We can distinguish three uses of the slash notation.

- Slash arrows from an empty node to a sister node signal predicate identity
- Slash arrows from an XOBJ or XADV node to a mother or sister node indicate the subject of the infinite verb
- Slash arrows from other verbal nodes signal a shared argument

The first case is not a dependency relation at all, so there is no need to infer a label. In the second case, the slash arrow always indicates a SUB relation: there is widespread typological support for 'controlled' functions always being subjects, and this holds for the old Indo-European languages as well. Only the third kind of slash arrow may have different labels. We therefore constrain such arrows to cases where the shared arguments have the same function in both conjuncts. This is by far the most frequent case. The following example illustrates how unambiguous interpretation is possible even in complex cases:

[8]The relation between XOBJ and OBJ is of another nature since verbs subcategorize differently for OBJ and XOBJ.

(13)



kai	*labontes*	*auton*	*edeiran*
and	take.PRS.PTCP	he.ACC.SG	beat.3.PL.PFV

kai	*apesteilan*	*kenon*
and	send away.3.PL.PFV	emtpy.ACC.SG.

'Having captured him, they beat him and sent him away empty-handed.'

In this graph, we capture the information that *auton* is an object of all verbs in the sentence; that the subjects of the free predicatives *labontes* and *kenon* are elided arguments of the verbs *edeiran* and *apesteilan*; and that *labontes* is an adverbial adjunct (here, in fact, equivalent to a subordinate temporal clause) relevant to both main verbs.

Thus the simple addition of an extra binary relation in our data model enables us to capture a wide variety of facts about structure sharing without introducing a plethora of empty nodes. Notice also that our two levels of annotation are not interdependent: while the slash arrows cannot be interpreted without the dependency tree, the opposite does not hold. If in some processes (such as automated parsing) we are forced to exclude slash relations, the dependency tree can still be drawn independently and the slash relation added by other means.

The annotators made two kinds of errors in dealing with the slash arrows: sometimes they forgot to use them where they should have been used, and sometimes they attached arrows indicating double dependencies to an empty node. In example 12, they introduced an empty OBL-node under the empty verb in second conjunct, and a slash arrow from the OBL-node to the preposition *in*. In this way, they enforced a more consistent interpretation of the slash notation as an indication of identity of lexical material. We considered this option, but rejected it due to the proliferation of empty nodes it leads to.

Fortunately, both these errors are easily detectable, and as future extension the annotation interface will enforce validation constraints that prohibit dependency graphs that have an XOBJ or XADV nodes lacking a slash arrow, or in which a slash arrow exits an empty node which has been assigned a non-PRED relation.

4. Conclusion

While work on the complete PROIEL corpus is still in its infancy, we feel that the pilot stage of the project has enabled us to establish a relatively firm base for the annotation scheme to be used and the accompanying tools that annotators will rely on. In the course of our work, we have been confronted with many of the 'classical' difficulties that syntactic theory still struggles with, e.g. the difficulty of strict morphological categorisation and the analysis of ellipsis in Dependency Grammar, but also encountered novel problems that arise in ancient Indo-European languages. Our solutions to these problems should enable us to address the needs of the PROIEL project, but still be sufficiently theory neutral to ensure that the corpus will be useful for others, and the technologies used should enable an open exchange of data and eliminate many obstacles for potential reuse of our data.

5. References

David Bamman and Gregory Crane. 2006. The design and use of a Latin dependency treebank. In Jan Hajič and Joakim Nivre, editors, *Proceedings of the Fifth International Treebanks and Linguistic Theories*, pages 67–78, Prague. Data available from `http://nlp.perseus.tufts.edu/syntax/treebank`.

David Bamman, Marco Passarotti, Gregory Crane, and Savina Raynaud. 2007. Guidelines for the syntactic annotation of Latin treebanks. Technical report, Tufts Digital Library, Medford. Version 1.3.

Gregory Crane, Robert F. Chavez, Anne Mahoney, Thomas L. Milbank, Jeffrey A. Rydberg-Cox, David A. Smith, and Clifford E. Wulfman. 2001. Drudgery and deep thought: Designing digital libraries for the humanities. *Communications of the ACM*, 44(5):34–40.

Gregory Crane. 1987. From the old to the new: Integrating hypertext into traditional scholarship. In *Hypertext '87: Proceedings of the 1st ACM conference on Hypertext*, pages 51–56. ACM Press.

Timothy Friberg, Barbara Friberg, and Neva F. Miller. 2000. *Analytical Lexicon of the Greek New Testament*. Baker, Grand Rapids.

William A. Gale and Kenneth W. Church. 1993. A program for aligning sentences in bilingual corpora. *Computational Linguistics*, 19(1):75–102.

Jan Hajič. 1998. Building a syntactically annotated corpus: The Prague dependency treebank. In Eva Hajičová, editor, *Issues of Valency and Meaning. Studies in Honor of Jarmila Panevová*, pages 12–19. Charles University Press, Prague.

James W. Hunt and M. Douglas McIlroy. 1976. An algorithm for differential file comparison. Computing Science Technical Report 41, Bell Laboratories.

Ulrik Sandborg-Petersen. 2008. Tischendorf's 8th edition Greek New Testament with morphological tags. Version 2.0. `http://morphgnt.org/projects/tischendorf`.

James Strong. 1890. *The exhaustive Concordance of the Bible : showing every word of the text of the common English version of the Canonical Books*. Methodist Book Concern, New York.

Integrating Language Technology in a web-enabled Cultural Heritage system

Penny Labropoulou[1], Harris Papageorgiou[1], Byron Georgantopoulos[1], Dimitra Tsagogeorga[1], Iason Demiros[2], Vassilios Antonopoulos[2]

[1] Institute for Language and Speech Processing
Epidavrou & Artemidos, Athens, Greece
[2] Qualia Technologies of Understanding
Voriou Ipirou 4, Athens, Greece
{ penny, xaris, byron, dtsag } @ilsp.gr ; { idemiros, vantonop } @qualia.gr

Abstract

This paper describes a web-enabled sophisticated Cultural Heritage (CH) system giving access to digital resources of various media, which exploits Language Technologies (LT) in order to enhance the performance of the search and retrieval mechanisms. More specifically, the paper presents the system requirements and architecture, drawing aspects from: (a) the cultural data repository and its particularities; (b) the unified metadata scheme that has been devised, integrating elements from various metadata standards, providing thus a rich description of the resources; (c) the thesauri (one of the major pillars of the system) that provide uniform access to the resources. The LT that form part of the system construction and use are presented in detail, focusing on the Term Extraction and Named Entity Recognition tools used in the construction of the thesauri and the metadata annotation process, and the Term Matching module exploited in the mining process for the identification of query terms which appear in a morphosyntactically similar form in the thesauri.

1. Introduction

In this paper we present an integrated system currently under development which aims to provide web-enabled access to digitized Cultural Heritage (CH) resources of various media. The paper describes the main components of the system and the way these are exploited to facilitate users' access. Section 2 presents the cultural resources handled in the project and the metadata scheme that has been implemented to describe them. Section 3 presents the system architecture and Section 4 the thesauri that are integrated in the system as a separate module. Finally, Section 5 discusses the Language Technology (LT) methods and tools that have been (a) exploited in the course of the project to facilitate the system construction and (b) integrated in the search and retrieval mechanisms to improve the system performance. In the last section, we present conclusions and future work.

2. Cultural resources collection & metadata standards

2.1 Description of the resources

The CH system presented in this paper aims at providing web-enabled sophisticated access to the digital cultural resources of the Centre for Asia Minor Studies (CAMS). The institute's resources range from paper manuscripts to 3-D objects related to the everyday life cycle of Greeks in Asia Minor from the end of the Byzantine era until the 1922 divide. From these, only a subset has been selected for digitization:

- the **Archive of Oral Accounts** (AOA), which includes "testimonies" of refugees recorded on paper by field researchers in the 1930's - 1940's;
- the **Manuscript Archive** (MA), made up of manuscripts containing refugees' reminiscences;
- the **Photographic Archive** (PA), with photos picturing everyday life in Asia Minor, the 1922 events and the resettlement in Greece;
- the **Map and Sketches Archive** (MSA), with maps and cartographic sketches drawn by the Cartographic Service;
- and, finally, a **Video Collection** (VC), including audiovisual material depicting Asia Minor settlements now and then.

As evidenced by the above description, the material not simply pertains to different media but, in fact, constitutes five distinct archives, which have been collected and documented at different times by different people to a varying degree of granularity and according to different classification schemes, although they all address the same subject. The main goals of the project were to:

- digitize the selected material to a format suitable for long-term preservation;
- document the data with any information deemed necessary for its preservation and exploitation, catering for the interrelation of the resources so as to highlight their common features and allow unified access to the whole set;
- make the resources publicly available to all interested parties, ranging from the research community to laypersons such as Asia Minor descendants, school students and people interested in finding out more about the particular historical period.

2.2 The Metadata Scheme

The core feature of the platform we have developed for the access and exchange of the CAMS cultural resources lies on the metadata scheme used for their documentation. The CAMS metadata scheme is composite, adopting as a base the Encoding Archival Description (EAD, http://www.loc.gov/ead/) enriched with elements drawn from various metadata standards for digital resources (e.g. Dublin Core (http://dublincore.org/) for the whole set, TEI (http://www.tei-c.org/index.xml) for the MA, DIG35 (http://www.i3a.org/i_dig35.html) for the PA, etc.), adapted to the project's needs and supplemented, where required, with the institute's particular requirements. In this way, the CAMS scheme succeeds in encoding different types of information regarding the form and content of the original resources and their digital substitutes, in a uniform representation, regardless of the medium format of each particular item.

The adopted methodology follows considerations reported in (Duval et al, 2002) stating: "In a modular metadata world, data elements from different schemas as well as vocabularies and other building blocks can be combined in a syntactically and semantically interoperable way". The resulting scheme has been implemented as an Application Profile and incorporated in the Metadata Annotator Workbench ("MetaWork") which is integrated within the overall Service Oriented Architecture of our platform. The MetaWork interface provides annotators with a powerful tool for efficient metadata collaborative editing of digitized cultural resources and search thereof.

3. System architecture

The web-enabled CH system that provides access to the CAMS material draws elements from the following pools of resources:

- the digital substitutes of the CAMS originals, that have been authorized for web access;
- the annotation metadata accompanying them;
- two thesauri implementing the classification schemes of the cultural material.

A Service-oriented architecture connecting various components and subsystems that are loosely coupled and easily managed has been implemented (Figure 1). The architecture enables the seamless integration of different components that are situated behind the portal and represented as services. At the front end, users interact with a single composite service that aggregates LT and back-end data sources. User friendliness, on the one hand, and utmost functionality and informativeness, on the other, have been the guiding principles behind its development.
The user interfaces allow querying on a specific subset of the metadata which has been carefully selected by the CAMS personnel to reflect the needs of intended users. Querying on this subset is performed by free text input and value selection from lists, depending on the query element. The two most important query elements are related to the classification of the material, which falls along two axes, namely geographical and thematic, implemented in the form of thesauri.

4. Thesauri description

The use of thesauri in knowledge mining systems has well proven its merits (Carpineto et al., 2001). In the CH domain, thesauri can be exploited to mine knowledge hidden in textual data (combined with LT tools) or, in the simplest scenario, to improve access to the resources.
In CH systems, thesauri are commonly used as part of the annotation metadata scheme providing classification information. The two CAMS thesauri are also used for the same purpose but, in contrast to the usual practice, they are kept outside the annotation scheme. This has been a deliberate choice: an important requisite of the project was the ability to open up to the national and international research community, by providing the appropriate infrastructure for establishing links with other parties collecting similar material; by keeping the thesauri content independent of the annotation scheme, they can be easily merged/replaced/supplemented by other thesauri and term lists or linked with meta-thesauri, allowing for seamless cross-collection access, without jeopardizing the annotation scheme. Linking with the appropriate documents is performed at the annotation scheme level.
The thesauri are integrated in the query system providing access to the resources with which the terms are linked. As such, they are of utmost importance to the system performance. Until now, users interested in CAMS resources had access to the material through the intermediary of specialized personnel, who have full knowledge of the contents and form of the material and who could process the users' queries so as to present them with the most appropriate material in response to their queries. This process is replaced in the web-based services by the thesauri: the query system "interprets" the users' queries through the thesauri in order to mine the best plausible results in the digital repository.
Access to the resources via the thesauri is performed in the form of free text input in the query system, or by browsing through the thesaurus viewer, which presents terms hierarchically or alphabetically sorted. The user can choose between viewing the detailed term card or the associated digital resources.

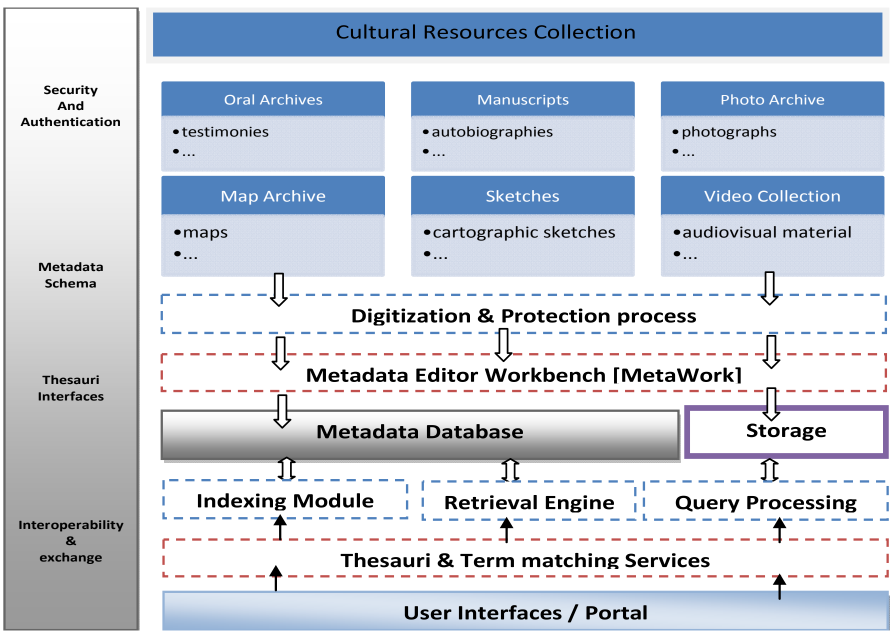


Figure 1: System architecture

In both modes, the thesauri relations[1] are exploited to improve the system performance: in the browsing mode, the user can navigate from higher to lower nodes, from non-preferred to preferred terms and/or to related terms; in the search mode, the system is responsible for query expansion utilizing the relations in order to improve recall and precision, a methodology well established in the bibliography of information retrieval systems.
The ***thematic thesaurus*** has been constructed in such a way that it could be expanded in the future so as to include other material of the institute itself or an external source. In this endeavour, the Ethnographic Thesaurus (http://et.afsnet.org/about.html) has served as our "prototype", i.e. a subset of the ET facets have been selected for our purposes, and a similar structure for the hierarchical organization of the terms has been adopted. The thesaurus contains approximately 1,100 terms in each of the three project languages (Greek, English, and Turkish).
The ***geographical thesaurus*** includes ca. 4,500 terms, of which about 2,500 represent preferred terms. This is due to the existence of multiple alternative names used for the same place, reflecting the coexistence and interaction of the Turkish and Greek populations (and others, of course) in the area; this variety is further accentuated by the presence of various dialects and idioms rendering more linguistic and/or spelling variants. The organization of this thesaurus has followed the original system devised by CAMS, because it reflects the situation in the time period under investigation: some of the settlements included in the material do not exist any longer, while others have changed names; therefore, it is extremely difficult to attribute the names with absolute certainty to current locations.
The development of both thesauri has been assisted by a term extraction tool described in Section 5.1.

5. LT deployment in the system

Existing LT tools and techniques have been deployed in the framework of the project to improve access to the cultural material in two ways:

(a) during the construction process, to minimize the time and cost effort of building certain system modules, namely the thesauri and the annotation metadata and
(b) in the form of components integrated in the system, to enhance its performance: two modules, a stemmer and a term matching module are activated in the query subsystem.

5.1. LT in the system development

The construction of a thesaurus is an important time- and effort-cost procedure. To minimize this cost, ***Term Extraction*** (TE) methods are recommended for term harvesting from digitized resources. In CH systems, such resources are the textual data contained in the written documents and the metadata of audiovisual items. However, in this project, this could not be the case because the written documents are not digitized at text level while term-intensive annotation metadata (e.g. titles of manuscripts, captions of photographs etc.) were not available at the time of the thesauri construction.

Thus, a backup plan that we decided to explore was to apply the TE procedure to an electronic corpus of *relevant* texts[2]. The corpus consisted of texts provided by CAMS, consisting of descriptions of the resources (e.g. promotional material, newspaper articles) and research work based on them (mainly academic articles from the CAMS Bulletin and scientific journals), as well as texts from internet sources which, however, addressed only a small part of the CAMS themes (mainly the Asia Minor expedition and tragedy); a very small subset is taken directly from the annotation metadata, namely the questionnaire used in the construction of the AOA. Finally, the idea of performing multilingual TE was quickly abandoned because the vast majority of the texts were in Greek, very few in English and none in Turkish.
The thesaurus construction procedure followed in the course of the project consisted of the following steps:
(a) pre-processing of the corpus, i.e. tokenization, below-part-of-speech (POS) tagging and lemmatization;
(b) ***Named Entity (NE) recognition***, catering for the identification of named locations (the "terms" included in the geographical thesaurus);
(c) TE aiming at identifying candidate terms for both thesauri;
(d) manual validation of the list of candidate terms by subject experts, aiming at selection of valid terms;
(e) processing of the term list in order to add terms and establish the appropriate relations between them; for the purposes of the project, this step included the task of adding the translation equivalents.

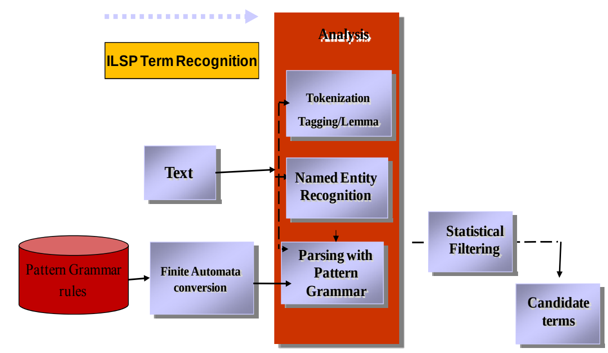


Figure 2: Term extraction workflow

Figure 2 shows the automatic processing stages used in the above procedure. The LT tools employed are:

- the ILSP tokenizer, that performs tokenization and identification of sentence boundaries and other surface linguistic phenomena (punctuation, dates, numbers etc.);
- the ILSP tagger (Papageorgiou et al., 2000), which consists of a transformation-based POS tagger (compatible with the relevant EAGLES standards) and a lexicon-based lemmatizer: the tagger assigns initial tags to known words by looking up in a lexicon created from the manually annotated corpus during training; then, a suffix-lexicon is used for initially tagging unknown words; a set of contextual rules are

[1] The form of the thesauri is conformant to the ISO2788 and ISO5964 standards for monolingual and multilingual thesauri.

[2] Relevant texts should not be overlooked in TE since they contain alternative modes of expressing similar notions which qualify as "non-preferred terms" for access to the material.

then applied to improve the initial phase output; finally, the lemmas are retrieved from a Greek morphological lexicon containing 70K lemmas;
- the MENER NE recognition system (Giouli et al., 2006): it is a highly modified version of the best-scoring system that participated in the CoNLL-2003 shared task developed by Hai Leong Chieu and Hwee Tou Ng. MENER adopts a single-level maximum entropy approach that combines sentence-based local evidence about words, with global evidence, which is a collection of features drawn from other occurrences of those words within the same document;
- a TE module developed at our institute (Georgantopoulos & Piperidis, 2000): it is a hybrid system comprising:
  - corpus parsing of below-POS tagged texts, based on a term pattern grammar endowed with regular expressions and feature-structure unification, which produces single- and multi-word candidate terms;
  - statistical filtering, using the TFIDF metric, in order to remove grammar-extracted terms lacking statistical evidence; TFIDF is a standard weight computation method which favours terms having high frequency in the current document and low frequency in a reference corpus (collection of documents); the reference corpus is a general language news corpus of about ~10K words in size for Greek.

All the resources are semi-automatically linked to the appropriate term in the metadata annotation process: some of the metadata (among which, titles of manuscripts and videos as well as captions of the PA) have been made available at later stages of the project in digital form in simple document files[3]; these have been lemmatized and checked for inclusion of terms from the thesauri; the links established in this procedure will be checked by the annotator during the final annotation process.

In addition, the NE module has been used for the identification of named persons in the captions of photographs; these will be coded in the relevant photographs' metadata, providing links to the metadata associated with other resources: e.g. a person that appears on a photograph may be an interviewee in the AOA or the writer of a manuscript in the MA.

5.2. LT in the query system

The query system benefits from two modules that enhance its performance: a stemmer and a term matching module.

Stemming is the technique that has been selected for the free text queries (in Greek and English); the reasons for having selected stemming over morphological lexica is that it can be easier implemented for other languages (especially for English), while the metadata vocabulary, although rather limited, contains mainly domain specific words which are not included in general language lexica.

The most important LT module integrated in the system is the ***Term Matching module***. This is used to enhance the thesauri query subsystem.

Terms in thesauri are usually kept in one single form, called *canonical*. However, they can be encountered in various other forms which may differ from the canonical form in morphological and other aspects. In addition, spelling mistakes, word omissions, changes in functional words, in word order etc. will not allow a term to be located in its exact form in a predefined term list. Matching problems arise when a system lacking linguistic processing capabilities is used for looking up terms in various forms against a thesaurus. This derives from the fact that crude string comparison techniques will fail in matching terms that are morphologically or in other ways varied. A more sophisticated technique is needed in order to identify the closest canonical term or terms (ideally *the* term) that the user originally was looking for. This process is called *term matching.*

As aforementioned, the problem of alternative names is especially accentuated in the geographical thesaurus; for instance, nowadays Izmir is also encountered under the names of Smyrna, Smirna, Smyrni, Smirni etc., related to naming at different time periods and variance in the transcription systems into English. In addition, we should stress the fact that the names of the settlements (a large number of which are of Turkish origin) have been recorded by the institute's field researchers, who often had no knowledge of Turkish, and who tried to put into the Greek alphabet what they understood from the refugees' pronunciation. Thus, the original Turkish name may have been paraphrased into Greek (either because the Greek inhabitants of Asia Minor used a slightly modified version of the original name or simply because the recording was not the most appropriate one). In this situation, term matching is of great value to the query enhancement, since it allows the system to locate closely similar forms (paraphrases) of the same word. In the thematic thesaurus, its use is less evident but still appreciated, since it allows for finding spelling variants without entering them in the thesaurus.

The term matching algorithm takes as input a query term and returns a ranked set of the closest canonical terms. It operates in two pipelined stages:

1. Word matching: The words of the query term are matched against the wordforms occurring in the thesaurus. An error tolerant finite-state recognizer is used to recognize all the strings in the regular set plus any strings that are produced from the regular set by a small number of unit editing operations. The edit distance between two strings measures the minimum number of unit editing operations of insertion, deletion, substitution and transposition of adjacent symbols required to convert the one string into the other (Oflazer, 1996). The algorithm, being string-oriented, is generic enough to be applicable to all the three languages of the thesaurus. For example, ed(Φουντουκλού, Φουντουκλιά)=2 and ed(Smyrna, Smirni)=2.

[3] Had these metadata been available earlier, their use in the TE procedure would have greatly contributed to the results, since they are term-intensive data for the particular subject.

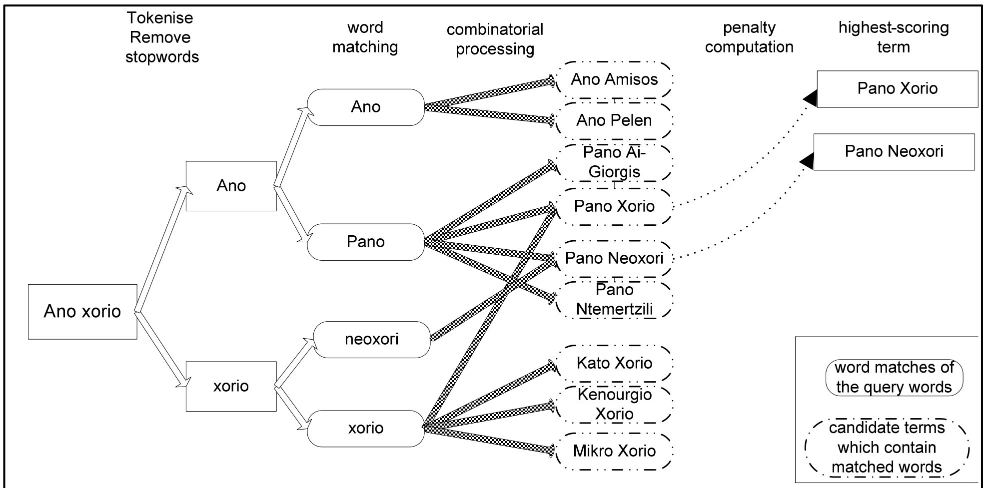


Figure 3: Term matching process

In order to avoid the over-generation of useless suggestions having too many spelling mistakes, we have imposed a maximum allowable number of errors *t*, where *t*'s value is analogous to the word length. After experimentation, we have found the following values for *t* that demonstrate best results:

length	t
1 - 4	1
5 - 7	2
8 – 9	3
10+	4

The matching procedure examines every possible path from the start node to the final nodes, in depth-first mode, so that the parsed strings deviate no more than *t* editings from the string to be matched. The algorithm is very efficient since the edit distance of each new (longer) string is recursively calculated from the edit distances of its substrings.

2. Term matching: We recall for each word retrieved in the previous step the set of terms where it is encountered in the thesaurus. Combinatorial processing is performed on the set of candidate terms in order to decide which thesaurus terms match best the query term, based on a penalty-assigning mechanism illustrated in the following formula:

$$\text{term-penalty}_i = \sum_{j=1}^{n} (\text{edit-penalty}_{ij})$$

where *n* is the number of words in the query term, *term penalty* is the penalty of the i-th candidate term, *edit-penalty* is the pre-computed edit distance of the j-th word in the i-th term (or a high constant penalty if no match exists). Finally, the terms are ranked according to their penalty, in ascending order.

The scoring mechanism effectively computes the join of all the canonical terms that contain at least one of the query words (verbatim or variants). It then classifies these terms through the penalising schema: exact word matches do not get a penalty score, error tolerant matches are penalised with their respective edit distance and, finally, terms are heavily penalised for every query word they do not contain. As a result, good scoring terms are the ones that contain word matches for all or most of the query words. Terms containing all of the query words are guaranteed to outrank all the other candidates. We further filter terms having an equal number of common words with the query term by sorting them according to the word matching distances.

Figure 3 illustrates the term matching process when the input query is "Ano xorio". The best two terms contain word matches, though not exact, for both query words, "Ano" and "xorio".

6. Conclusions and future work

In this paper, we have described a CH system giving access to heterogeneous digitized resources. We have focused on the benefits of exploitation and integration of LT both at the metadata annotation process and the search and retrieval interface as well as the challenges such an endeavour imposes. In the near future, we intend to evaluate the system in quantitative and qualitative terms.

7. References

Carpineto, C., De Mori, R., Romano, G., Bigi, B. (2001). An information theoretic approach to automatic query expansion. *ACM Transactions on Information Systems*, Vol 19, n. 1, pp. 1--27.

Duval, E., Hodgins, W., Sutton, S. & Weibel, S.L. (2002). Metadata principles and practicalities. *D-Lib Magazine*, 8(4). Available: doi:10.1045/april2002-weibel.

Georgantopoulos, B. & Piperidis, S. (2000). A Hybrid Technique for Automatic Term Extraction. *Proceedings of the ACIDCA 2000 Conference*.

Giouli, V., Konstadinidis, A., Desipri, E., Papageorgiou, H., Piperidis, S. (2006). Multi-domain Multi-lingual Named Entity Recognition: Re-visiting & grounding the resources issue. *International Conference on Language Resources & Evaluation, LREC 2006*.

Oflazer, K. (1996). Error-tolerant Finite State Recognition with Applications to Morphological Analysis and Spelling Correction. *Computational Linguistics* Volume 22 No.1.

Papageorgiou, H., Prokopidis, P., Giouli, V., Piperidis, S. (2000). A Unified POS Tagging Architecture and its application to Greek. *International Conference on Language Resources & Evaluation, LREC 2000*, pp.1455--1463.

Morphological Tagging of Old Norse Texts and Its Use in Studying Syntactic Variation and Change

Eiríkur Rögnvaldsson[1], Sigrún Helgadóttir[2]
Department of Icelandic, University of Iceland[1], Árni Magnússon Institute for Icelandic Studies[2]
{Árnagarði við Suðurgötu, IS-101[1], Neshaga 16, IS-107[2]}, Reykjavík, Iceland
E-mail: {eirikur,sigruhel}@hi.is

Abstract

We describe experiments with morphosyntactic tagging of Old Norse narrative texts using different tagging models for the TnT tagger (Brants, 2000) and a tagset of almost 700 tags. It is shown that by using a model that has been trained on both Modern Icelandic texts and Old Norse texts, we can get 92.7% tagging accuracy which is considerably better than the 90.4% that have been reported for Modern Icelandic. In the second half of the paper, we show that the richness of our tagset enables us to use the morphosyntactic tags in searching for certain syntactic constructions and features in a large corpus of Old Norse narrative texts. We demonstrate this by searching for – and finding – previously undiscovered examples of two syntactic constructions in the corpus. We conclude that in an inflectional language like Old Norse, a morphologically tagged corpus like this can be an important tool in studying syntactic variation and change.

1. Introduction

In a previous project (Helgadóttir 2005; 2007), we have trained the TnT tagger written by Brants (cf. Brants, 2000) on a corpus of Modern Icelandic. The corpus used in that project was created in the making of the Icelandic Frequency Dictionary (*Íslensk orðtíðnibók*, henceforth IFD; Pind et al., 1991). The IFD corpus is considered to be a carefully balanced corpus consisting of 590,297 tokens with 59,358 types – both figures including punctuation.

The corpus contains 100 fragments of texts, approximately 5,000 tokens each. All the texts were published for the first time in 1980–1989. Five categories of texts were considered, i.e. Icelandic fiction, translated fiction, biographies and memoirs, non-fiction (evenly divided between science and humanities) and books for children and youngsters (original Icelandic and translations).

The texts were pre-tagged using a specially designed computer program and the tagging was then carefully checked and corrected manually. Thus, this corpus is ideal as training material for data-driven statistical taggers, such as the TnT tagger.

In the present project, we applied the TnT tagger trained on the Modern Icelandic corpus to Old Norse (Old Icelandic) texts.[1] This paper describes the results of this experiment, and also describes our experiments with using the morphologically tagged Old Norse corpus to search for syntactic constructions. We conclude that in an inflectional language like Old Norse, a morphologically tagged corpus like this can be an important tool in studying syntactic variation and change.

[1] It is customary to use the term 'Old Norse' for the language spoken in Norway, Iceland and the Faroe Islands up to the middle of the 14th century. The overwhelming majority of existing texts written in this language is either of Icelandic origin or only preserved in Icelandic manuscripts. For the purposes of this paper, 'Old Norse' is thus synonymous with 'Old Icelandic'.

2. Tagging Modern Icelandic

In this section, we describe the tagset used in our research, and give a brief overview of our experience with the training of the TnT tagger on Modern Icelandic texts.

2.1 The tagset

The tagset developed for the IFD corpus is very large, compared to tagsets designed for English at least, such as the Penn Treebank tagset (Marcus et al., 1993). The size of the tagset of course reflects the inflectional character of Icelandic, since it is for the most part based on the traditional Icelandic analysis of the parts of speech and grammatical categories, with some exceptions where that classification has been rationalized.

In the tag strings, each character corresponds to a single morphosyntactic category. The first character always marks the part of speech. Thus, the sentence *Hún hefur mætt gamla manninum* 'She has met the old man' will be tagged like this:

(1) Hún *fpven*
hefur *sfg3eþ*
mætt *ssg*
gamla *lkeþvf*
manninum *nkeþg*

The meaning of the tags is as follows:

(2) *fpven:* pronoun (f) – personal (p) – feminine (v) – singular (e) – nominative (n)
sfg3eþ: verb (s) – indicative (f) – active (g) – 3rd person (3) – singular (e) – past (þ)
ssg: verb (s) – supine (s) – active (g)
lkeþvf: adjective (l) – masculine (k) – singular (e) – dative (þ) – definite (v) – positive (f)
nkeþg: noun (n) – masculine (k) – singular (e) – dative (þ) – suffixed article (g)

Of the word forms in the IFD corpus, 15.9% are ambiguous as to the tagset within the IFD. This figure is quite high, at least compared to English, which reflects the fact that the inflectional morphology of Icelandic is considerably more complex than English. Icelandic nouns can have up to 16 grammatical forms or tags, verbs up to 106 different tags, and adjectives up to 120 tags. Altogether, 639 different tags occur in the IFD corpus, but the total sum of possible tags is around 700.

Some of the ambiguity is due to the fact that inflectional endings in Icelandic have many roles, the same ending often appearing in many places (e.g. *-a* in *penn**a*** for all oblique cases in the singular (acc., dat., gen.), and accusative and genitive in the plural of the masculine noun *penni* 'pen', producing 5 different tags for one form of the same word). The most ambiguous of word forms in the IFD, *minni*, has 24 tags in the corpus, and has not exhausted its possibilities (Bjarnadóttir, 2002).[2]

2.2 Training the tagger

The computer files for the IFD corpus each contain one text excerpt. Each file was divided into ten approximately equal parts. From these, ten different disjoint pairs of files were created. In each pair there is a training set containing about 90% of the tokens from the corpus and a test set containing about 10% of the tokens from the corpus. Each set should therefore contain a representative sample from all genres in the corpus. The test sets are independent of each other whereas the training sets overlap and share about 80% of the examples. All words in the texts except proper nouns start with a lower case letter.

Results for ten-fold cross-validation testing for the TnT tagger are shown in table 1 (cf. Helgadóttir, 2005; 2007). It is worth noticing that these results show lower performance rates when the tagger is applied to the Icelandic corpus than is achieved for example for Swedish as reported in Megyesi (2002). In that study, TnT was applied to and tested on the SUC corpus with 139 tags compared to the Icelandic tagset of almost 700 tags. Performance rates are also considerably lower than have been reported for the systems trained on the Penn treebank.

Type	*Accuracy %*
All words	90.4
Known words	91.7
Unknown words	71.6

Table 1: Mean tagging accuracy for all words, known words and unknown words for TnT.

Table 1 shows results for known words, unknown words and all words. Mean percentage of unknown words in the ten test sets was 6.84. This is similar to what was seen in the experiment on Swedish text (Megyesi, 2002) and indicates that the major difficulty in annotating Icelandic words stems from the difficulty in finding the correct tag for unknown words. Words belonging to the open word classes (nouns, adjectives and verbs) account for about 96% of unknown words in the test sets whereas words in these word classes account for just over 51% of all words in the test sets.

3. Tagging Old Norse texts

Having trained the TnT tagger on Modern Icelandic texts, we wanted to find out whether the tagger could be of help in tagging Old Norse narrative texts, with the purpose of facilitating the use of these texts in research on syntactic variation and change. To create a manually annotated training corpus for Old Norse from scratch would have been a very time-consuming task. Thus, the possibility of using the bootstrapping method that we describe in this section was a key factor in realizing this project.

Bootstrapping is of course a common approach in training taggers and parsers. To our knowledge, however, this approach has not been used in historical linguistics to develop tagging models for a different stage of language than the tagger was originally trained on. Our method somewhat resembles the experiments of Hwa et al. (2005), who used parallel texts to build training corpora by projecting syntactic relations from English to languages for which no parsed corpora were available. The training corpora created using this method were then in turn used to develop stochastic parsers for the languages in question. The whole process took only a small fragment of the time it would have taken to create a manually corrected corpus to train the parsers.

The common factor in our project and the work reported by Hwa et al. (2005) is the use of another language, or (in our case) another stage of the same language, as a starting point in the bootstrapping process. Our experiments with bootstrapping the tagging of Old Norse texts are described in this section.

3.1 Old Norse vs. Modern Icelandic

At a first glance, it may seem unlikely that a tagger trained on 20th century language could be applied to 600-700 years old texts. However, Icelandic is often claimed to have undergone relatively small changes from the oldest written sources up to the present. The sound system, especially the vowel system, has changed dramatically, but these changes have not led to radical reduction or simplification of the system and hence they have not affected the inflectional system, which has not changed in any relevant respects. Thus, the tag set developed for Modern Icelandic can be applied to Old Norse without any modifications.

The vocabulary has also been rather stable. Of course, a great number of new words (loanwords, derived words and compounds) have entered the language, but the majority of the Old Norse vocabulary is still in use in Modern Icelandic, even though many words are confined to more formal styles and may have an archaic flavor.

[2] *minni* can be a noun meaning 'memory', present tense of the verb *minna* 'remind', comparative of the (irregular) adjective *lítill* 'small'. In all of these words we find extensive syncretism, resulting in many different tag strings for this word form in each part of speech.

On the other hand, many features of the syntax have changed (cf. Faarlund, 2004; Rögnvaldsson, 2005). These changes involve for instance word order, especially within the verb phrase, the use of phonologically "empty" NPs in subject (and object) position, the introduction of the expletive *það* 'it, there', the development of new modal constructions such as *vera að* 'be in the process of' and *vera búinn að* 'have done/ finished', etc.

In spite of these changes, we found it worthwhile to try to adapt the tagging model that we had trained for Modern Icelandic to our Old Norse electronic corpus. Our motive was not to get a 100% correct tagging of the Old Norse texts, but rather to facilitate the use of the texts in syntactic research, cf. Section 4 below.

3.2 The Old Norse corpus

Our Old Norse corpus consists of a number of narrative prose texts (sagas), which are assumed to have been written in the 13th and 14th centuries – a few of them probably later. Among these are many of the most famous Old Norse sagas. The division of the corpus is shown in Table 2:

Text	*Tokens*
Family Sagas (around 40 sagas) (*Íslendingasögur*)	1,074,731
Sturlunga Saga ("Contemporary Sagas")	283,002
Heimskringla (Sagas of the Kings of Norway)	250,920
The Book of Settlement (*Landnámabók*)	42,745
Total	1,651,398

Table 2: Division of the Old Norse corpus.

The texts we use are (with the exception of The Book of Settlement) taken from editions, which were published between 1985 and 1991 (Halldórsson et al., 1985-86; Kristjánsdóttir et al., 1988; Kristjánsdóttir et al., 1991). In these editions, the text has been normalized to Modern Icelandic spelling. This involves, for instance, reducing the number of vowel symbols ('æ' is used for both 'ae ligature' (æ) and 'oe ligature' (œ), 'ö' is used for both 'o with a slash' (ø) and 'o with a hook'), inserting *u* between a consonant and a word-final *r* (*maðr* 'man' > *maður*), shortening word-final *ss* and *rr* (*íss* 'ice' > *ís, herr* 'army' > *her*), changing word-final *t* and *k* in unstressed syllables to *ð* and *g*, respectively (*þat* 'it' > *það, ok* 'and' > *og*), etc. Furthermore, a few inflectional endings are changed to Modern Icelandic form.

It must be emphasized, however, that these changes do not in any way simplify the inflectional system or lead to the loss of morphological distinctions in the texts. Thus, the texts are just as good as sources of syntactic evidence as texts that are published in the normalized Old Norse spelling.

On the other hand, we must point out that the original versions of these texts do not exist; the texts are mostly preserved in vellum manuscripts from the 13th through the 15th centuries, but some of them only exist in paper manuscripts from the 16th and 17th centuries. This makes it extremely difficult to assess the validity of these texts as linguistic evidence, since it is often impossible to know whether a certain feature of the preserved text stems from the original or from the scribe of the preserved copy, or perhaps from the scribe of an intermediate link between the original and the preserved manuscript. It is well known that scribes often did not retain the spelling of the original when they made copies; instead, they used the spelling that they were used to. In many cases, two or more manuscripts of the same text are preserved, and usually they differ to a greater or lesser extent. Furthermore, it is known that not all of the editions that our electronic texts are based on are sufficiently accurate (cf., for instance, Degnbol, 1985).

Even though this may to some extent undermine the validity of the texts as sources of syntactic evidence, it does not directly concern the main subject of this paper, which is to show that we can use a tagging model developed for Modern Icelandic to assist us in making the Old Norse corpus a usable tool in studies of syntactic variation and change. There is no reason to believe that possible inaccuracies and errors in the texts – cases where they fail to mirror correctly the syntax of the manuscripts – have any effects on the tagging accuracy. That is, the use of more accurate editions would not lead to less accurate tagging.

3.3 Training the tagger on the Old Norse corpus

We started by running TnT on the whole Old Norse corpus using the tagging model developed for Modern Icelandic (cf. Helgadóttir, 2005; 2007). We then measured the accuracy by taking four samples of 1,000 words each from different texts in the corpus – one from the *Family Sagas*, one from *Heimskringla*, and two from *Sturlunga Saga* – and checking them manually. Counting the correct tags in these samples gave 88.0% correct tags, compared to 90.4% for Modern Icelandic.

Even though these results were worse than those we got for Modern Icelandic, we considered them surprisingly good. The syntax of Old Norse differs from Modern Icelandic syntax in many ways, as mentioned above, and one would especially expect the differences in word order to greatly affect the performance of a trigram based tagger like TnT. However, sentences in the Old Norse corpus are often rather short, which may make them easier to analyze than the longer sentences of Modern Icelandic.

We then selected seven whole texts (sagas) and two fragments from the *Sturlunga* collection for manual correction – around 95,000 words in all. This amounts to one third of the *Sturlunga* collection. The manual correction was a time-consuming task, but the time and effort spent on checking and correcting the output of TnT was only a small fragment of the time and effort it would have taken to tag the raw text.

We trained TnT on the corrected text (95,000 words), tagged the whole corpus again with the resulting model, and measured the accuracy on the same four samples of 1,000 words each as in the first experiment. Now the results were much better – 91.7% correct tags, which is better than the 90.4% accuracy that we got for Modern Icelandic. It may seem surprising how much the accuracy improved when we used this model, especially when we consider that the training corpus was much smaller than the training corpus for Modern Icelandic (95,000 words compared to more than 500,000). On a closer look, however, this is understandable.

First, many of the errors occurring in the first experiment could be predicted and were easy to correct. For instance, the word *er* was always classified as a verb in the third (or first) person singular present indicative ('is, am'), as it usually is in Modern Icelandic. In Old Norse, however, this word is very often a temporal conjunction ('when') or a relative particle ('that, which'). When the tagger was trained on a corrected Old Norse text, it could quickly and easily learn the correct tagging of these words, due to their frequency.

Second, it is well known that tagging accuracy is usually very much lower for unknown words than for known words, and the number of unknown words was much lower in the second experiment. In the first experiment, using the model for Modern Icelandic, the unknown word rate was 14.6%, reflecting the fact that a number of Old Norse words are rare or do not occur in Modern Icelandic. In the second experiment, using the model for Old Norse, the unknown word rate dropped to 9.6%, even though the training corpus was much smaller as pointed out above. This reflects the relatively small vocabulary of the Old Norse texts, which in turn reflects the narrow universe that the texts describe (cf. also Rögnvaldsson, 1990).

Finally, we trained TnT on a union of the corrected Old Norse texts and the Modern Icelandic texts. Thus, the training set for the final experiment consists of around 500,000 words from Modern Icelandic texts plus 95,000 words from Old Norse texts. When we tagged the Old Norse corpus using this model, we got 92.7% accuracy for the same four samples as in the first two experiments. The results of the three experiments are shown in Table 3:

Tagging model	*Accuracy %*
Modern Icelandic model	88.0
Old Norse model	91.7
MI + ON model	92.7

Table 3: Tagging accuracy for Old Norse texts using three different tagging models.

It is possible to improve the results by tagging the texts using all three models and combining the results of different models in various ways. All three models agree on the tags for 84.6% of the words. In 80.9% of the cases, they agree on the correct tag, but for 3.7% of the words, all three models agree on a wrong tag.

For 15.4% of the words, the models disagree. In most cases, two of them assign the same tag and the third model assigns a different tag. In a few cases, each model assigns a separate tag. Thus, if we assume that the tag is correct when all three models agree, we only need to look at 15.4% of the whole corpus. This means that the highest possible accuracy to be obtained using this method is 96.3%, since all models agree on a wrong tag in the remaining cases as pointed out above.

We could also choose to disregard the model that is trained only on Modern Icelandic texts, since it gives much lower accuracy than the other two models. The remaining models agree on the tagging of 93.5% of the words – incorrectly for 4.3% of the words. If we only look at the 6.5% where the models disagree, we are down to around 107,000 words that we have to correct manually. This is a manageable task, which we intend to finish in the near future. We think that performance may exceed 95% after manual revision of the training set, assuming that about half of the disagreements can be correctly resolved. This is an acceptable result in our view, and should be sufficient for most uses of the corpus.

In this connection, it must be pointed out that a majority of the tagging errors only involve one morphosyntactic feature. Thus, nouns are often tagged as accusative instead of dative, or vice versa, whereas gender and number are correctly tagged; verbs are often tagged as 3rd person instead of 1st person, whereas mood, voice, number, and tense are correctly tagged; etc. This means that by using fuzzy search, we should in many cases be able to find what we are looking for, even if the words are not quite correctly tagged.

4. Tagged texts in syntactic research

Over the past two decades, interest in historical syntax has grown substantially among linguists. Accompanied by the growing amount of electronically available texts, this has led to the desire for – and possibility of – creating syntactically parsed corpora of historical texts, which could be used to facilitate search for examples of certain syntactic features and constructions. A few such corpora have been developed, the most notable being the Penn Parsed Corpora of Historical English, developed by Anthony Kroch and his associates (Kroch and Taylor, 2000; Kroch et al., 2004). These corpora have already proven their usefulness in a number of studies of older stages of English (cf., for instance, Kroch et al., 2000; Kroch and Taylor, 2001).

We wanted to know whether our tagged Old Norse corpus could be used in syntactic research in a similar manner as syntactically parsed corpora. We had been using the raw unannotated texts for this purpose (cf., for instance, Rögnvaldsson, 1995; 1996) but the search for certain syntactic constructions and features had proven to be cumbersome and give insufficient results. Although our tagging is morphological in nature, the tags carry a substantial amount of syntactic information and the tagging is detailed enough for the syntactic function of words to be more or less deduced from their morphology and the adjacent words. Thus, for instance, a noun in the nominative case can reasonably safely be assumed to be a subject, unless it is preceded by the copula *vera* 'to be' which is in

turn preceded by another noun in the nominative, in which case the second noun is a predicative complement. A noun in the accusative or dative case can in most instances be assumed to be a (direct or indirect) object, unless it is immediately preceded by a preposition (cf. also Rögnvaldsson, 2006). As is well known, Modern Icelandic also has accusative and dative subjects, and even some nominative objects (Thráinsson, 2007), but these can easily be identified from their accompanying verbs.
To test the usefulness of the tagging of Old Norse texts in syntactic research, we have made a small study of two controversial and disputed features of Old Norse syntax; Object Shift and Passive. These studies are described in this section.

4.1 Object Shift

As originally described by Holmberg (1986), Object Shift is the process of moving a (direct or indirect) object to the left across a negation. In Modern Icelandic, this process applies both to pronouns and full NPs (or DPs), as shown in (3), whereas in the "Mainland" Scandinavian languages (Danish, Norwegian, and Swedish), it only applies to pronouns, as (4) shows (examples from Thráinsson, 2007). The "shifted" object is underlined whereas the negation is in boldface and the "place of origin" of the shifted object is shown by an underscore:

(3) Nemandinn las <u>bókina</u> **ekki** ___
the student read book not
'The student didn't read the book'
Nemandinn las <u>hana</u> **ekki** ___
the student read she not
'The student didn't read it'

(4) *Studenten læste <u>bogen</u> **ikke** ___
the student read book not
'The student didn't read the book'
Studenten læste <u>den</u> **ikke** ___
the student read she not
'The student didn't read it'

It has been suggested that this difference between Icelandic and the Mainland Scandinavian languages is somehow related to the fact that Icelandic has a much richer case morphology than the Mainland Scandinavian languages (cf. Holmberg and Platzack, 1995). If this were so, one would expect to find both types of Object Shift in Old Norse, since the case system of Icelandic is in all relevant respects the same as in Old Norse. The Mainland Scandinavian languages would then be assumed to have lost Object Shift of full DPs due to the loss of case inflections.
However, it has been claimed that Object Shift of full DPs does not occur in Old Norse. Mason (1999) claims to have found two examples of shifted full DP objects in his study of nine Old Norse sagas. Sundquist (2002), on the other hand, concludes "that these two examples do not provide evidence for a full DP Object Shift like in modern Icelandic". Haugan (2001) did not find any examples of full DP Object Shift in his study of Old Norse, and neither did Sundquist (2002) in a study of Middle Norwegian. Thus, Sundquist concludes that "full DP Object Shift is not an option in earlier stages of Mainland Scandinavian".
It is therefore of considerable theoretical interest to search for examples of full DP Object Shift in Old Norse texts. However, this is a tedious and time-consuming task. Even though this is a perfectly grammatical construction in Modern Icelandic, it appears to be very rare in texts. Thus, one can read dozens or even hundreds of pages without finding a single example. When the constructions that we are looking for are that rare, it is easy to overlook the few examples that actually occur in the texts that we read. Given the rarity of full DP Object Shift in Modern Icelandic, one may wonder whether those who have studied Object Shift in Old Norse have looked at a large enough corpus.
We have searched for examples of full DP Object Shift in our morphologically tagged Old Norse corpus. In this search, we use a simple program that searches for a verb in the indicative or the subjunctive, followed by a noun, an adjective, or a demonstrative pronoun in an oblique case, followed by a negation (one of the words *eigi*, *ei*, *ekki* 'not', *aldrei*, *aldregi* 'never'). We allow for up to two words between the noun/adjective/demonstrative pronoun and the negation. Thus, in addition to simple sentences with a noun immediately following the verb and preceding the negation, we will find sentences where both a demonstrative pronoun and an adjective precedes the noun, and sentences where a prepositional phrase consisting of a preposition and a noun follows the head noun. Of course, we will neither get 100% precision nor 100% recall by using this pattern. It will miss some potential examples of Object Shift; for instance, sentences with an adverb modifying a prenominal adjective when a demonstrative pronoun is also present, or sentences with an adjective modifying an object of a preposition, which follows the head noun. Furthermore, this search pattern will return a number of sentences that are not instances of Object Shift.
When we run this search pattern on the Old Norse corpus, it returns 245 examples. The majority of these examples do not show Object Shift. These are for instance sentences like (5):

(5) hann skal <u>þetta fé</u> **aldregi** fá ___ síðan
'he shall this money never get since'
'he shall never have this money again'

In this sentence, the fronted NP *þetta fé* is not an object of the verb *skal*, but rather an object of the verb *fá*. Thus, this is not an instance of Object Shift but rather shows OV order in the VP, which is quite a different matter (see, for instance, Rögnvaldsson, 1996; Hróarsdóttir, 2000).
However, it doesn't take long to clean the search results and throw away the sentences that do not show Object Shift. When we have finished this cleaning, it appears that we really are left with some genuine examples of full DP Object Shift:

(6) a. Nú leita þeir um skóginn og finna Gísla **eigi** ___
now search they about the forest and find Gisli not
'Now they search through the forest and don't find Gisli'

b. er hann dræpi Þórð **eigi** ___ og förunauta hans
when he killed Thord not and companions his
'if he didn't kill Thord and his companions'

c. og fundu Þórð **eigi** ___ sem von var að
and found Thord not as expectance was at
'and not surprisingly, they didn't find Thord'

Using this method, we found at least 9 indisputable examples of full DP Object Shift. This may not be the exact number of such sentences in our corpus. First, in addition to these examples, there are some borderline cases, which may or may not be interpreted as instances of Object Shift. Second, our searching method does not guarantee 100% recall, as explained above. However, this doesn't really matter for our purposes. We have shown conclusively that full DP Object Shift existed in Old Norse, contrary to what has previously been claimed in the literature; and we have demonstrated the efficiency of our searching method.

4.2 Passive

Another controversial feature of Old Norse syntax is the nature of the passive. It has sometimes been claimed (Dyvik, 1980; Faarlund, 1990) that all passive sentences in Old Norse are lexical but not derived by NP-movement (or chain-formation). This claim has been disputed, for instance by Benediktsson (1980), and it has been claimed that the existence of agentive prepositional phrases (*by*-phrases) would be an argument against this analysis, since such phrases presuppose a derivational analysis of passive sentences (Rögnvaldsson, 1995).

Be that as it may, it is quite clear that agentive prepositional phrases in passives are rather rare in Modern Icelandic, and hence, one would not expect to find many of them in Old Norse. Faarlund (2004), for instance, quotes two such examples but concludes: "This is very rarely found, however."

It is not easy to search for such examples in an unannotated electronic text. One would have to search for the preposition *af* 'by', but this preposition is one of the most frequent words in Old Norse so this search would return thousands of sentences. However, once we have a morphologically tagged text, it is relatively easy to search for agentive prepositional phrases. We can search for a past participle, followed by *af*, followed by a nominal (noun, pronoun, adjective) in the dative. Since the distinction between past participle forms and adjectives in the neuter singular is not always clear, and the tagger makes a number of errors in this classification, we also search for the adjectives in addition to the past participles. This search returns some 130 sentences. Most of them are not instances of agentive phrases, since the preposition *af* can also have other functions. Nevertheless, we have found at least 15 sentences with agentive prepositional phrases, only a few of which have previously been quoted in the literature on this subject. Three of these sentences are shown below – the agentive phrases in boldface:

(7) a. að Þorvarður Spak-Böðvarsson hafi skírður verið **af Friðreki biskupi**
that Thorvard Spak-Bodvarsson has baptized been by Fridrek bishop
'that Thorvard Spak-Bodvarsson has been baptized by bishop Fridrek'

b. Og er þetta mál var rannsakað **af lögmönnum**
and when this case was investigated by lawyers
'and when lawyers investigated this case'

c. Óttar gerði sem honum var boðið **af Sighvati**
Ottar did as him was ordered by Sighvat
'Ottar did what Sighvat ordered him'

Thus, our searching method has enabled us to strengthen the evidence for the existence of derivational passive in Old Norse.

5. Conclusion

In this paper, we have demonstrated that it is possible to use a tagging model trained on Modern Icelandic texts to facilitate tagging of Old Norse narrative texts. By using this method, we are able to tag a large corpus of Old Norse with acceptable accuracy in a relatively short time – only a fragment of the time it would have taken to build a tagging model for Old Norse from scratch.

Furthermore, we have shown that a corpus tagged using a rich tagset based on morphosyntactic features can fruitfully be used in the search for a number of syntactic constructions, and hence is a valuable tool in studying syntactic variation and change. Of course, a morphologically tagged corpus like the one we have built doesn't amount to a fully parsed corpus. Several syntactic features cannot be searched for using our method. However, given the tremendous effort it would take to build a parsed corpus of this size, we think our method is an alternative that must be taken seriously.

Later this year, we intend to make the tagged Old Norse texts available on the web using the Xaira program (www.oucs.ox.ac.uk/rts/xaira/) from the British National Corpus. This will enable users to search the corpus for complex patterns using both words and tags in the search text. Thus, the corpus will hopefully be of great use to anyone studying Old Norse language, literature, and culture.

6. Acknowledgements

This project was partly supported by a grant from the University of Iceland Research Fund to the project "The syntactic use of Old Icelandic POS tagged texts".

Thanks to the Árni Magnússon Institute of Icelandic Studies for providing access to the IFD corpus used in this research.

Thanks are also due to three anonymous reviewers who made many valuable comments which helped us improve the paper considerably, although we have not been able to follow all their suggestions.

7. References

Benediktsson, H. (1980). The Old Norse Passive: Some Observations. In Hovdhaugen, E. (Ed.), *The Nordic Languages and Modern Linguistics* 4. Universitetsforlaget, Oslo, Norway, pp. 108-119.

Bjarnadóttir, K. (2002). The Icelandic µ-TBL Experiment: Preparing the Corpus. Paper presented at NLP1 final session, January 9. GSLT, Växjö, Sweden.

Brants, T. (2000). TnT – A Statistical Part-of-Speech Tagger. In *Proceedings of the 6th Applied NLP Conference, ANLP-2000*, Seattle, WA, pp. 224-231.

Degnbol, H. (1985). Hvad en ordbog behøver – og andre ønsker [What a Dictionary Needs – and Others Wish for]. In *The Sixth International Saga Conference. Workshop Papers* I. Det arnamagnæanske institut, University of Copenhagen, Copenhagen, Denmark, pp. 235-254.

Dyvik, H. (1980). Har gammelnorsk passiv? [Does Old Norse have the Passive?] In Hovdhaugen, E. (Ed.), *The Nordic Languages and Modern Linguistics* 4. Universitetsforlaget, Oslo, Norway, pp. 82-107.

Faarlund, J.T. (1990). Syntactic Change. *Toward a Theory of Historical Syntax*. Mouton, Berlin, Germany.

Faarlund, J.T. (2004). *The Syntax of Old Norse*. Oxford University Press, Oxford, UK.

Halldórsson, B., Torfason, J., Tómasson, S., Thorsson, Ö. (Eds.). (1985-86). *Íslendinga sögur* [The Icelandic Family Sagas]. Svart á Hvítu, Reykjavik, Iceland.

Haugan, J. (2001). Old Norse Word Order and Information Structure. Doctoral dissertation, NTNU, Trondheim, Norway.

Helgadóttir, S. (2005). Testing Data-Driven Learning Algorithms for PoS Tagging of Icelandic. In Holmboe, H. (Ed.), *Nordisk Sprogteknologi. Årbog 2004*. Museum Tusculanums Forlag, University of Copenhagen, Denmark, pp. 257-265.

Helgadóttir, S. (2007). Mörkun íslensks texta [Tagging Icelandic Text]. *Orð og tunga*, 9, pp. 75-107.

Holmberg, A. (1986). Word Order and Syntactic Features in the Scandinavian Languages and English. Doctoral dissertation, University of Stockholm, Stockholm, Sweden.

Holmberg, A., Platzack, C. (1995). *The Role of Inflection in the Syntax of Scandinavian Languages*. Oxford University Press, Oxford, UK.

Hróarsdóttir, Þ. (2000). *Word Order Change in Icelandic: from OV to VO*. John Benjamins, Amsterdam, The Netherlands.

Hwa, R., Resnik, P., Weinberg, A., Cabezas, C., Kolak, O. (2005). Bootstrapping Parsers via Syntactic Projection across Parallel Texts. *Natural Language Engineering*, 11(3), pp. 311-325.

Kristjánsdóttir, B., Halldórsson, B., Sigurðsson, G., Grímsdóttir, G.Á., Ingólfsdóttir, G., Torfason, J., Tómasson, S., Thorsson, Ö. (Eds.). (1988). *Sturlunga saga* [The Sturlunga Collection]. Svart á Hvítu, Reykjavik, Iceland.

Kristjánsdóttir, B., Halldórsson, B., Torfason, J., Thorsson , Ö. (Eds.). (1991). *Heimskringla* [The Sagas of the Kings of Norway]. Mál og Menning, Reykjavik, Iceland.

Kroch, A., Santorini, B., Delfs, L. (2004). Penn-Helsinki Parsed Corpus of Early Modern English. http://www.ling.upenn.edu/hist-corpora/PPCEME- RELEASE-1/

Kroch, A., Taylor, A. (2000). Penn-Helsinki Parsed Corpus of Middle English, second edition. http://www.ling.upenn.edu/hist-corpora/PPCME2- RELEASE-2/

Kroch, A., Taylor, A. (2001). Verb-Object Order in Early Middle English. In Pintzuk, S., Tsoulas, G., Warner, A. (Eds.), *Diachronic Syntax: Models and Mechanisms*. Oxford University Press, Oxford, UK, pp. 132-163.

Kroch, A., Taylor, A., Ringe, D. (2000). The Middle English Verb-Second Constraint: a Case Study in Language Contact and Language Cange. In Herring, S., Schoesler, L., van Reenen, P. (Eds.), *Textual Parameters in Older Language*. John Benjamins, Philadelphia, pp. 353-391.

Marcus, M.P., Santorini, B., Marcinkiewicz, M.A. (1993). Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2), pp. 313-330.

Mason, L. (1999). Object Shift in Old Norse. MA thesis, University of York, York, UK.

Megyesi, B. (2002). Data-Driven Syntactic Analysis – Methods and Applications for Swedish. Doctoral dissertation, Department of Speech, Music and Hearing, KTH, Stockholm, Sweden.

Pind, J. (Ed.), Magnússon, F., Briem, S. (1991). *Íslensk orðtíðnibók* [Icelandic Frequency Dictionary, IFD] Orðabók Háskólans, Reykjavík, Iceland.

Rögnvaldsson, E. (1990). Orðstöðulykill Íslendinga sagna [The Concordance to the Icelandic Family Sagas]. *Skáldskaparmál*, 1, pp. 54-61.

Rögnvaldsson, E. (1995). Old Icelandic: A Non-Configurational Language? *NOWELE*, 26, pp. 3-29.

Rögnvaldsson, E. (1996). Word Order Variation in the VP in Old Icelandic. *Working Papers in Scandinavian Syntax*, 58, pp. 55-86.

Rögnvaldsson, E. (2005). Setningafræðilegar breytingar í íslensku. [Syntactic Changes in Icelandic.] In Thráinsson, H. (Ed.) *Setningar. Handbók um setningafræði* [Sentences: A Handbook on Syntax]. (Íslensk tunga III.) Almenna bókafélagið, Reykjavík, Iceland, pp. 602-635.

Rögnvaldsson, E. (2006). The Corpus of Spoken Icelandic and Its Morphosyntactic Annotation. In Henrichsen, P.J., Skadhauge, P.R. (Eds.): *Treebanking for Discourse and Speech, Proceedings of the NODALIDA 2005 Special Session on Treebanks for Spoken Language and Discourse*. Copenhagen Studies in Language 32. Samfundslitteratur, Copenhagen, Denmark, pp. 133-145.

Sundquist, J.D. (2002). Object Shift and Holmberg's Generalization. In Lightfoot, D. (Ed.), *Syntactic Effects of Morphological Change*. Oxford University Press, Oxford, UK, pp. 326-347.

Thráinsson, H. (2007). *The Syntax of Icelandic*. Cambridge University Press, Cambridge, UK.

Building a large dictionary of abbreviations for named entity recognition in Portuguese historical corpora

[1]Oto Vale, [2]Arnaldo Candido Jr.*, [2]Marcelo Muniz, [1]Clarissa Bengtson, [1]Lívia Cucatto, [1]Gladis Almeida, [4]Abner Batista, [4]Maria C. Parreira, [3]Maria Tereza Biderman, [2]Sandra Aluísio

Federal University of São Carlos (UFSCar), University of São Paulo (USP), State University of São Paulo (UNESP)

[1] UFSCar - Department of Linguistics - Via Washington Luís, Km. 235 – C.P. 676 - CEP 13.565-905 - São Carlos, SP Brazil

[2] USP - Centre of Computational Linguistics (NILC)/ Department of Computer Sciences
C.P. 668 - CEP: 13560-970 - São Carlos, SP, Brazil

[3] UNESP, FCLAR - Department of Linguistics - Rod. Araraquara - Jaú Km1, Bairro dos Machados, C.P. 174, CEP: 14.800-901, Araraquara, SP, Brazil

[4] UNESP, IBILCE - Department of Linguistics - Rua Cristóvão Colombo, 2265, Bairro: Jardim Nazareth, CEP: 15054-000 São José do Rio Preto, SP, Brazil

E-mail: otovale@gmail.com, arnaldoc@icmc.usp.br, marcelo.muniz@gmail.com, clabengtson@hotmail.com, liviacucatto@yahoo.com.br, gladis.mba@gmail.com, abnerfortunato@gmail.com, mcparreira2002@yahoo.com.br, mtbider@attglobal.net, sandra@icmc.usp.br

Abstract

Abbreviated forms offer a special challenge in a historical corpus, since they show graphic variations, besides being frequent and ambiguous. The purpose of this paper is to present the process of building a large dictionary of historical Portuguese abbreviations, whose entries include the abbreviation and its expansion, as well as morphosyntactic and semantic information (a predefined set of named entities – NEs). This process has been carried out in a hybrid fashion that uses linguistic resources (such as a printed dictionary and lists of abbreviations) and abbreviations extracted from the *Historical Dictionary of Brazilian Portuguese (HDPB)* corpus via finite-state automata and regular expressions. Besides being useful to disambiguate the abbreviations found in the *HDBP* corpus, this dictionary can be used in other projects and tasks, mainly NE recognition.

1. Introduction

The *Historical Dictionary of Brazilian Portuguese* (*HDBP*), the first of its kind, is based on a corpus of Brazilian Portuguese texts from the sixteenth through the eighteenth centuries (including some texts from the beginning of the nineteenth century). The HDBP is a three-year project, which started in 2006, developed under the sponsorship of CNPq, Brazil. Organizing this historical dictionary has required an extensive, time-consuming analysis of documents, published texts and manuscripts produced by eyewitnesses in the early stages of Brazilian history. One important difficulty in compiling this corpus derived from the absence of press agencies in colonial Brazil, which had a precarious communication system. Only in 1808, after escaping from Napoleon's army, did the Portuguese monarchy transfer the government of the Portuguese empire to Brazil and improved communications. Moreover, peculiarities affecting language must be considered, such as biodiversity and multifaceted cultural traditions from different regions of the country. To implement the *HDBP* project, we created an integrated network of researchers from various regions of Brazil and Portugal, including linguists and computer scientists from 11 universities. Our team comprises 18 researchers holding a PhD, with complementary skills and expertise, and 23 graduate and undergraduate students.

This project fills a gap in Brazilian culture, for it is developing a dictionary that describes the vocabulary of Brazilian Portuguese in the beginning of the country's history. At that time, Brazilian language was still dependent on European Portuguese, even though some vocabulary was already being coined on this side of the Atlantic. On the one hand, the speakers of those days faced a world materially and culturally different from what was known in Europe; therefore, they needed to designate referents of this new universe, which were hitherto unnamed, using words from the Portuguese linguistic system. The hundreds of native languages then spoken in Brazil had their own vocabulary for designating elements of the Brazilian fauna and flora, but these words did not exist in European Portuguese. On the other hand, customs and institutions gradually began to form in this new society with the infusion of new cultures, resulting in new words, different from those used in the Portuguese metropolis.

To build the corpus, we collected documents in public archives and libraries all over Brazil and in Portugal. This corpus totals 2,458 texts; 287,570 sentences; 16,505,808 tokens (of which 368,850 are different from each other); 7,492,473 simple forms[1] (of which 368,529 are different from each other); and 82.2 MB. In a similar endeavor related to European Portuguese, researchers of the Universidade Nova de Lisboa have built the "Corpus Informatizado do Português Medieval"[2], comprising Latin-Romance texts from the ninth to the twelfth centuries, and Portuguese texts from the twelfth to the sixteenth centuries, totaling some 2 million words. Our corpus was built to be processed with corpus processing system UNITEX[3] (Unicode – UTF-16)

* Scholarship CNPq, Brazil.

[1] This is the total number of words in the corpus that are composed of letters belonging to Historical Portuguese alphabet.

[2] Digital Corpus of Medieval Portuguese: http://cipm.fcsh.unl.pt.

[3] http://www-igm.univ-mlv.fr/~unitex/

and with Philologic[4] (Unicode – UTF-8), since the latter is web-based and includes several corpus-processing tools, as for example AGREP [5], used to check for similar or alternative spellings in Philologic. To process this large corpus, we have faced the typical problems researchers are likely to encounter when dealing with old documents, starting with text digitalization. Rydberg-Cox (2003) and Sanderson (2006) state that, in historical Latin, Greek and English texts, to mention just a few languages, words broken at the end of a line are not always hyphenated; word-breaks are not always used; common words and word-endings are abbreviated with non-standard typographical symbols; uncommon typographical symbols pervade non-abbreviated words; and spelling variation is common even within the same text. We encountered these same problems in the *HDBP* project. First of all, the non-existence of an orthographical system in the afore-mentioned centuries generated a Babel of graphic systems being used by many different scribes or copyists. Giusti et al. (2007) focus on this difficulty introducing both an approach based on transformation rules to cluster distinct spelling variations around a common form, which does not always correspond to the orthographic (or modern) form, and choices made to build a dictionary of spelling variants of Brazilian Portuguese based on such clusters.Another problem was scribes' habit of abbreviating words to facilitate handwriting – there are many thousands of such abbreviations. Therefore, for the correct understanding of texts, it was necessary to expand abbreviations, a task that presents two main difficulties. The first is related to the use of modern knowledge sources to perform expansion, since gazetteers, encyclopedias and heuristics currently in use do not address directly the needs of historical material describing people, places, and other entities that often do not appear in modern sources (Crane & Jones, 2006). The second, and perhaps most important, is that even if we had adequate knowledge sources for expanding abbreviations they are highly ambiguous with respect to meaning, which is critical for understanding correctly not only the abbreviations themselves but also the whole text (Kerner et al., 2004).

In general, if abbreviations are not expanded correctly they can limit the effectiveness of: i) information extraction and retrieval systems in digital libraries; ii) electronic index creation from a corpus; iii) Natural Language Processing (NLP) tools, such as taggers, parsers and named entity recognition (NER) systems to enrich corpora linguistically. Within the scope of the *HDBP* project, the failure of proper abbreviation expansion hinders the correct editing of dictionary entries. However, expanding each and every abbreviation manually in a several million-word corpus is a time-consuming, expensive and difficult – if not impossible – task, due to the inherent ambiguity of noun abbreviations, for example. In Section 4, we discuss our approaches to this problem. Automatic acronyms and abbreviations disambiguation have been given close attention in medical and biomedical domains, since text normalization is an important task for successful information retrieval and extraction from texts in these areas (Pakhomov, 2002; Hong et al., 2002; Schwartz & Hearst, 2003; Dannélls, 2006). However, most of this automatic abbreviation disambiguation research has focused on modern scientific material, whereas historical corpora and digital libraries remain largely ignored (Rydberg-Cox, 2003). Moreover, NER systems have only begun to be implemented for digital libraries (Crane & Jones, 2006). Taking the above into consideration, the purpose of this paper is to present our ongoing work to build a large dictionary of abbreviations that contains the pair abbreviation and its expansion, together with morphosyntactic and semantic information. We are developing this process in a hybrid fashion, using linguistic resources (such as a digitalized printed dictionary of abbreviations and the authoritative lists of abbreviations that accompany the material that is being digitalized) and abbreviations extracted from the *HDBP* corpus via finite-state automata and regular expressions. Since expanding abbreviations is a costly process, these automata were created to recognize larger patterns of abbreviations that are NEs or the same pattern that has different types of NEs. For example, the dictionary semantic tags allowed us to identify new NEs, i.e., sequences of words that can be identified as personal names in some contexts, but that are categorized as place, river or organization names in other contexts. We have been working on an iterative process, which started with linguistic and common sense knowledge, to attribute initial NE categories to an abbreviation and later try to capture new NEs to update the respective dictionary entry. Both the new NE categories and the larger abbreviations gathered in this iterative process will be inserted in our dictionary of abbreviations, which will be useful for other projects and tasks, mainly named entity recognition and abbreviation disambiguation in the *HDBP* corpus. In the next section, we explain the details of the *HDBP* corpus and the graphic form of abbreviations found in it, as well as some historical corpus projects that addressed the same issue. In Section 3, we describe the process of building a dictionary of abbreviations. In Section 4, we consider possible applications of the dictionary in the *HDBP* project itself and in other scenarios: abbreviation lookup and expansion; search for spelling variations of abbreviations; and linguistic research. Section 5 contains our conclusions and final remarks.

2. The *HDBP* corpus and its abbreviations

The texts in the *HDBP* corpus were written by Brazilian authors and Portuguese authors who have lived in Brazil for a long time. Among the texts selected for our corpus, there are, for instance, letters of Jesuit missionaries, documents of the bandeirantes (members of the exploratory expeditions that pushed Brazilian borders far into inland areas), reports of sertanistas (explorers of Northeastern Brazil), and documents of the Inquisition. Table 1 shows more details about the composition of our corpus.

Since the emphasis is on word meaning, we have selected mainly published texts with minor editing. Examples of such editing are the separation of words that come together in the

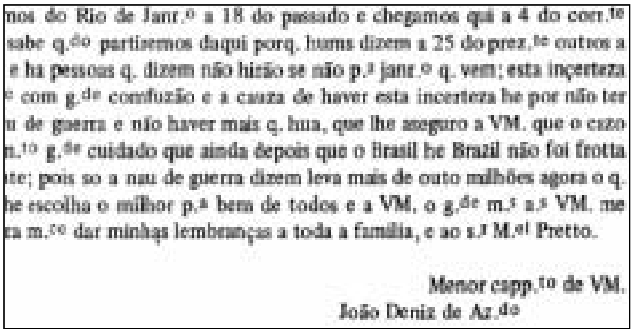
[4] http://philologic.uchicago.edu/index.php
[5] http://www.tgries.de/agrep/

original text, the introduction of punctuation marks, paragraph mark-up to facilitate reading, and the insertion of letters and words in places where editors were sure (or almost) that such items were missing.

Data	**XVIth**	**XVIIth**	**XVIIIth**	**XIXth**
Texts (%)	6.24	26.39	59.78	7.59
Sentences (%)	6.30	18.32	64.34	11.04
Simple Forms (%)	7.60	20.18	62.57	9.65
Megabytes (%)	7.23	19.95	63.09	9.73

Table 1: Distribution of texts by century

This decision was made to avoid potential problems during corpus compilation; however, we still had to deal with the following issues: 1) guaranteeing consistent assignment of Unicode characters in the texts, since digitalization and OCR correction have been done by different groups geographically distant from one another; 2) treating the graphic variation that alters frequency counts in the corpus, thus causing difficulties for the selection of variants in dictionary entries; and 3) expanding the abbreviations that pervade the texts.


nos do Rio de Janr.^{o} a 18 do passado e chegamos qui a 4 do corr.^{te}
sabe q.^{do} partiremos daqui porq. huns dizem a 25 do prez.^{te} outros a
e ha pessoas q. dizem não hirão se não p.^{a} janr.^{o} q. vem; esta incerteza
com g.^{de} comfuzão e a cauza de haver esta incerteza he por não ter
u de guerra e não haver mais q. hua, que lhe asseguro a VM. que o cazo
n.^{to} g.^{de} cuidado que ainda depois que o Brasil he Brazil não foi frotta
te; pois so a nau de guerra dizem leva mais de outo milhões agora o q.
he escolha o milhor p.^{a} bem de todos e a VM. o g.^{de} m.^{s} a.^{s} VM. me
a m.^{co} dar minhas lembranças a toda a familia, e ao s.^{r} M.^{el} Pretto.

Menor capp.^{to} de VM.
João Deniz de Az.^{do}

Figure 1: Excerpt from Cartas Remetidas para Lisboa em 12 de julho de 1715 In: Negócios coloniais, L. Lisanti (ed.)

There are several graphic forms for the abbreviations found in the *HDBP* corpus, some of which are shown in Figure 1:
a) abbreviations with a dot followed by a superscript piece of text, as in “Janr.o”/Janeiro (January) and “corre.te”/corrente (current), which appear in the first line;
b) abbreviations followed by a dot, as in “porq.”/porque (because) and in the three instances of “q.”/que (next/who/next, respectively) in the second and third lines. To be consistent, we used the character “^” to denote superscript, thus generating the forms “Janr.^o” and “corre.^te” showed in (a) above, which can be processed computationally. The same symbol is used when the abbreviation does not possess a dot, but has a superscript chunk, as in “O sor Jesus xpo”/O Senhor Jesus Cristo (The Lord Jesus Christ), leading to the forms “s^or” and “xp^o”. Other abbreviations display numerals, e.g., “8.bro”/Outubro (October), or other characters, e.g., “@” for the word ano (year). Some abbreviations only omit letters, as in “Glo”/Gonçalo (proper name Gonçalo), “Jão”/João (proper name João), “ldo”/licenciado (licensed), “Ros”/Rodrigues (proper name Rodrigues), and “snr” or “snro”/senhor (sir).
An additional difficulty posed by abbreviations is that they hinder the correct segmentation of sentences in UNITEX (Friburger, 2002). In the historical corpus, this problem is magnified by a large variation in the use of punctuation and capitalization, which also affects the recognition of named entities, since the corpus contains capitalized common words, as if they were proper names, and proper names in lower case. Table 2 illustrates problems related to abbreviations: ambiguity and variants. The first column shows 13 different expansions for the abbreviation “A”. The second column illustrates 13 different forms of abbreviating the name of the famous Brazilian city “Rio de Janeiro” (some of them in lower case), which makes them hard to memorize.

alteza (highness)	Rio de Jan.^{ro}
alvará (warrant)	Rio de Jan^{ro}
Amaro (proper name)	Rio de Janr.^{o}
Ana (proper name)	Rio de Jan.^{o}
anima (cheers up)	Rio de Jn^{ro}
ano (year)	Rio de janr^{o}
anos (years)	Rio de jan^{ro}
Antônio (proper name)	R^{o} de jan^{o}
arroba (measure of weight, singular)	R^{o} de Jan^{ro}
arrobas (measure of weight, plural)	R^{o} de janer^{o}
Assembléia (assembly)	R^{o} de Janr^{o}
assinado (signed)	R^{o} de Jnr^{o}
Atual (current)	Rio de Janr^{o}

Table 2: Ambiguity and spelling variation in abbreviations

Most previous work on Brazilian Portuguese historical corpus expands abbreviations manually, such as the project “Para uma História do Português do Brasil”[6] and “Projeto Programa para a História da Língua Portuguesa” (PROHPOR[7]). Also, the Tycho Brahe Project[8] (Paixão de Sousa & Trippel, 2006), whose purpose is to model the relation between prosody and syntax in the process that led from Classical to Modern European Portuguese, contains tagged and parsed texts written by Portuguese authors born between the sixteenth and nineteenth centuries. These texts had their abbreviations expanded manually to facilitate tagging and parsing. Although this corpus is large for the task of syntactic analysis – it is currently composed of 46 texts and still growing – it remains manageable by manual markup made with widely available standards in XML. The large-scale Germany-wide project Deutsch.Diachron.Digital (DDD) (Dipper et al., 2004) was set to build a diachronic corpus of German with texts from the ninth century (Old High German) to the present (Modern German) for linguistic, philological and historical research. This is a long-term project – it is planned to run over seven years – and its large core corpus will reach 40 million words. The abbreviations found in it will be expanded and annotated, based on well-accepted international standards in XML.
All projects mentioned above expand their abbreviations manually; however their development contexts differ from that of *HDBP*, which has only three years to develop both a large corpus and a dictionary. This is the reason why we had

[6] “For a History of Brazilian Portuguese”: http://www.letras.ufrj.br/phpb-rj/
[7] “Project Program for a History of Portuguese Language”: http://www.prohpor.ufba.br/projetos.html
[8] http://www.ime.usp.br/~tycho/

to approach the problem related to abbreviation expansion in a different way, detailed in Section 3.

3. Building a dictionary of abbreviations

In recent years, NLP researchers have focused on standardizing methods to construct linguistic resources, which led to the development of tools now accepted internationally. One of these construction standards, DELA (Dictionnaires éléctroniques du LADL), was developed at LADL (Laboratoire d'informatique documentaire et linguistique, University of Paris 7, France), jointly with the corpus-processing tool INTEX (Silberztein, 2000). DELA became the standard tool for developing electronic lexicons in the research network Relex[9]. These lexicons are used with INTEX, and now also with its open-source counterpart UNITEX (Paumier, 2006). This format allows for declaring simple and compound lexical entries, which can be associated with grammatical information and inflection rules. These dictionaries are linguistic resources specifically designed to perform automatic text processing operations. Types of DELA are DELAF, which comprises inflected simple words, DELAC and DELACF, for non-inflected and inflected compound words, respectively. The dictionaries of simple words (DELAS and DELAF) are lists of simple words associated with grammatical and inflectional information. The grammatical information is mainly morphological and corresponds to gender, number, degree, case, mood, tense, and person. However, with this format, it is possible to add syntactic and semantic information gradually (Ranchhod, 2001). DELAF lexical entries have the following general structure:
(Inflected word),(canonical form).(part of speech)[+(subcategory)]:morphological features

3.1 Customizing UNITEX

Processing lexicographical tasks in a corpus is easier when computational lexicons are available, and that was the reason why we adopted UNITEX in the *HDBP* project. UNITEX supports several languages, including Portuguese. Language-specific resources are grouped in packets referred to as idioms. When the UNITEX-PB (Muniz et al., 2005) was created, a lexicon for contemporary Brazilian Portuguese was incorporated into it. However, due to the peculiarities of historical texts, several changes had to be implemented and a new idiom was created, named "Português Histórico" (Historical Portuguese). These changes included characters that are no longer used in Portuguese, such as the long s (ʃ) and the tilde (~) over consonants. Some diacritical marks differ from the ordinary diacritics currently used in Portuguese, because the former can be placed over consonants. For instance, an accent mark over "m̃" was common in Historic Portuguese. The introduction of such characters was made possible using Unicode when the text was being compiled.

3.2 The hybrid process to build a dictionary of abbreviations

3.2.1 Printed resources

In order to build our dictionary of abbreviations, we employed lexicons together with corpus processing tools, especially to expand and enrich a digitalized printed dictionary (Flexor, 1991) with information about the NE categories appearing in the *HDBP* corpus. Flexor (1991) is a large alphabetically organized dictionary of abbreviations from the sixteenth through the nineteenth centuries. Although it has a large number of abbreviations (see Tables 3 and 4), most of them are not found in our corpus (only 16% appear in the *HDBP* corpus). We performed an experiment to recover abbreviations from the *HDBP* corpus using three simple heuristics, to estimate the amount of abbreviations in the corpus that is not present in the Flexor dictionary. We found out 7,045 abbreviations with the heuristics; only 35% of them (2,473) are in the Flexor dictionary. However, the Flexor dictionary is still worth using as it has abbreviations expansion. This dictionary is being revised to eliminate entries that could be considered spelling variants, as in the following example (pairs are composed of abbreviation and expansion): (Bẽs, bens); (Bẽz, bens); (Bãda, banda), since the tilde was part of the writing system of historical Portuguese.

Simple and Multi-word Abbreviations by Century					
Types	XVIth	XVIIth	XVIIIth	XIXth	Total
Flexor	2,050	4,091	14,376	9,939	21,869
Flexor (%)	9.37	18.70	65.74	45.45	139.26
Intersection of Flexor and Corpus	754	1,323	2,447	1,710	3,529
Intersection of Flexor and Corpus (%)	21.37	37.49	69.34	48.46	176.65
Coverage (%)	16.13				

Table 3: Abbreviations from Flexor (1991) by century, showing the % of forms found in the HDBP corpus[10].

This hybrid approach to build a dictionary has already been successfully used to develop a dictionary of anthroponyms (Baptista, Batista and Mamede, 2006) and was adopted in the HDBP project as well. Besides, we have employed the authoritative lists of abbreviations found in the books we digitalized.

Simple and Multi-word Abbreviations by n-grams							
Types	1	2	3	4	5	6 or +	Total
Flexor	17,872	1,624	833	527	302	711	21,869
Flexor (%)	81.73	7.42	3.81	2.41	1.38	3.25	100.00
Intersection of Flexor and Corpus	3,237	234	33	18	5	2	3,529
Intersection of Flexor and Corpus (%)	91.75	6.60	0.94	0.51	0.14	0.06	100.00

Table 4: Abbreviations from Flexor (1991), by size

Thus far, we have digitalized and processed abbreviations

[9] http://ladl.univ-mlv.fr/Relex/introduction.html

[10] Note that abbreviations can happen in more than one century.

from Flexor (1991) and some of the authoritative lists of abbreviations to be used in the UNITEX system. Initially, the information we had to include in the entries, gathered from printed resources, was just the abbreviation, its expanded form, and the century in which the text had been written. However, considering information retrieval, we soon found out that the canonical form was also extremely important and should be in the dictionary, as it is required in the DELA format. Therefore, we added this information, and now a search for the canonical form capitão (captain), for instance, produces the following forms (nonexhaustive list): Capitão, capitam, Capitaõ, cappitão, Cappitam, capitães, Capitãis, capitans and the abbreviated forms (nonexhaustive list):

Cap^{aens}	Cap^{ams}	Cap^{ans}	Cap^{ens}	Cap^{tm}
Cap^{es}	Cap^{ms}	Cap^{ns}	Cap^{s}	Capão
Cap^{tens}	Cap^{tes}	Capa^{ens}	Capitt^{es}	Capp.
Capm^{s}	Capn^{es}	Capn^{s}	Capns	Capp^{ão}
Capp^{ães}	Capp^{es}	Capp^{tes}	Capt^{es}	Capp^{nn}

Our dictionary of abbreviations differs from its counterparts developed in UNITEX, mainly in the use of a larger number of attributes. The most important attributes that have been added are: ABREV, used to denote abbreviation; SEC16, SEC17, SEC18, and SEC19 to indicate the century to which the lexical entry refers (information from Flexor (1991)) – the century attribute appears only in some entries, since it was not always possible to identify the period in which the abbreviation was used; <ENT>, to denote a named entity (NE) and the tag <INIT>, which is a collocation to extract certain types of NE. Each NE receives additional attributes, according to the category it belongs to. These categories were established by a taxonomy proposed in the evaluation contest of systems for recognizing named entities in Portuguese (HAREM[11]), organized by Linguateca. Among the ten HAREM categories, we have employed nine of them except OBRA (titles, man-made things). Figure 2 shows some lexical entries in DELA format. In the first line of the Figure 2, Brg^es is the form found in the corpus, Borges is the canonical form (lemma), N (noun) is the part-of-speech tag for the entry, ENT+PESSOA+ABREV+SEC19 are additional attributes, and ms (masculine singular) is the morphosyntactic tagging. We also included the expanded form (Borges), which may differ from the canonical form in some cases.

```
Brg^es,Borges.N+ENT+PESSOA+ABREV+SEC19:ms/Borges
Brag.,Braga.N+ENT+PESSOA+LOCAL+ABREV+SEC18:ms/Braga
Br^ça,Braça.N+ENT+VALOR+ABREV+SEC19:fs/Braça
7^bro,setembro.N+ENT+TEMPO+ABREV:ms/setembro
B^eis,bacharel.N+INIT+TITULO+ABREV:mp/bacharel
B.,beco.N+INIT+LOCAL+ABREV+SEC18:ms/beco
Bat^am,batalhão.N+INIT+ORGANIZAÇÃO+ABREV+SEC16:ms/batalhão
Bas^tos,bastardo.N+INIT+PARENTE+ABREV+SEC19:ms/bastardos
```

Figure 2: Entry samples from the dictionary

We have already processed letters A, B, C and some of the authoritative lists of abbreviations. From the 3051 simple abbreviations under letter A, 814 are named entities (<ENT>) and 548 have the tag <INIT>. 1789 were simple abbreviations. There are also 430 multi-word abbreviations in letter A. From the 488 simple abbreviations under letter B, 260 are named entities (<ENT>) and 138 have the tag <INIT>. Only 107 were common abbreviations. Some entries classified as <ENT> are <INIT> as well, such as "Barb^ro" (barber), a family name and a pattern used to introduce this profession. There are also 45 multi-word abbreviations in letter B, such as "Bn^s Ay^s" (Buenos Aires) and "Brigad^ro Insp^or" (Brigadeiro Inspetor/Inspector Brigadier). As for letter C, from the 2187 simple abbreviations, 364 are named entities and 853 have the tag <INIT>. There are also 510 multi-word abbreviations in letter C. All the multi-word abbreviations will be annotated later.

3.2.2 Generic patterns to extract different categories of NEs for an abbreviation

The use of heuristics is efficient for extending lexicons of NEs, such as in the search for words (or n-grams) that begin with a capital letter that is not in the beginning of a sentence or in the search for words followed by titles and forms of address. Thus, heuristic rules allow for the identification of named entities. However, the identification of some abbreviated NEs, such as "V. M." (Vossa Mercê/archaic Portuguese for "you"), is difficult, because the dot that follows V makes the NE look like the beginning of a sentence (". M."), and therefore impossible to be retrieved using the heuristic rule mentioned above[12].

An experiment[13] was performed to investigate NEs in the historical corpus, in order to extend and enrich the dictionary of abbreviations. This experiment was carried out with the dictionary of abbreviations described in Table 3. The three lists of abbreviations for letters A, B, and C and some short lists of abbreviations were first tagged with HAREM categories. The NEs received the tag <ENT>, whereas all entries received the tag <ABREV>. The tag <INIT> was created to designate abbreviated collocations found at the left of certain types of NEs, thus yielding three subcategories that were not present in HAREM, viz. <TITULO> (for jobs/professions and titles/positions), <PARENTE> (for family relations), and <TRATAMENTO> (for forms of address, since they are very pervasive in Flexor's dictionary). Besides, all ten HAREM categories were used to subcategorize <INIT>.

First, using the dictionary of abbreviated forms, we performed a search in the corpus for tag <ABREV> (rule 1), which resulted in 1,795,519 occurrences. Several of them were not abbreviated forms, but stopwords with similar formats. In addition, several occurrences were actually orthographic variants that looked like abbreviations, such as bom/bõ. This prompted us to re-examine the list of abbreviations to remove non-abbreviated variants of stopwords and abbreviated forms of stopwords (we call this the pre-processing phase). For instance, the prepositions "por" and "para" were abbreviated as "p.". However, this was also the abbreviation for padre (priest). Re-examining the list led us to create the rule 2 for searching forms in a UNITEX graph or using the following regular expression to locate retrieve abbreviated forms or the form "p." preceded by determiners:

[11] http://poloxldb.linguateca.pt/harem.php?l=classificacao_v3_sem

[12] Note that sentence breaking was not performed in the corpus preprocessing phase.

[13] In this experiment, we have used UNITEX version 2.0 and set UNITEX to find the longest matches in its searches.

<ABREV>+((o+ao+do+ho).p\.)
With this regular expression, all abbreviated forms in the dictionary can be retrieved, plus the form "p." preceded by determiners, thus decreasing the number of abbreviated forms. On applying the pre-processing cited above and the rule 2, the number of abbreviations dropped to 804,939. Before applying a search using tags, we tested the hypothesis that a significant number of abbreviated forms were either an NE or were in the vicinity of an NE. This test was carried out with the rule 3 depicted by:
(<ABREV>+((o+ao+do+ho).p\.)).(<MOT>+<MOT><MOT>+<MOT><MOT><MOT>+<MOT><MOT><MOT><MOT>)
applied to a search for abbreviated forms containing one to four elements. The number of retrieved abbreviations was 469,640. Therefore, further strategies are necessary to identify NEs, since we observed in our corpus that abbreviations tend to be close to each other. We carried out another search using tags, in which we replaced the tag <ABREV> by <INIT> (rule 4):
(<INIT>+((o+ao+do+ho).p\.)).(<MOT>+<MOT><MOT>+<MOT><MOT><MOT>+<MOT><MOT><MOT><MOT>)
With the rule 4, the number of occurrences dropped to 22,196. More than 50% were the abbreviated form "S.", which stands for "Saint", and abbreviated forms of address such as "S. M." (Sua Majestade/His or Her Majesty), "S. A." (Sua Alteza/His or Her Highness). The names of saints, however, were commonly other types of NE, not associated with PERSON. In fact, they were abbreviations for names of places (fazenda/farm, arrayal/hamlet, mosteiro/monastery, aldeia/village, bairro/district, villa/village etc.), rivers (Corgo/Brook, rio/river), organizations (mosteiro/monastery, fortaleza/fortress). We can check this information looking for such words at the left of the abbreviated form in the excerpt shown in Figure 3. This analysis of abbreviations productivity is useful for identifying and contextualizing new NEs. The use of these new attributes allows for sophisticated searches in the *HDBP* corpus and in other historical corpora. This is important, because we intend to make this resource available for research under request, since we cannot make it public due to copyright issues. It will be possible, for instance, to search for all NEs from the eighteenth century or for all NEs related to persons in their abbreviated forms.

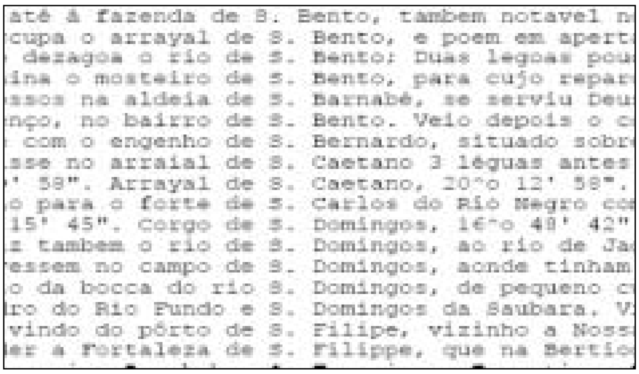

até à fazenda de S. Bento, tambem notavel n
cupa o arrayal de S. Bento, e poem em apert
dezagoa o rio de S. Bento; Duas legoas pou
ina o mosteiro de S. Bento, para cujo repar
ssos na aldeia de S. Barnabé, se serviu Deu
nço, no bairro de S. Bento. Veio depois o c
com o engenho de S. Bernardo, situado sobr
sse no arraial de S. Caetano 3 léguas antes
' 58". Arrayal de S. Caetano, 20° 12' 58".
o para o forte de S. Carlos do Rio Negro co
15' 45". Corgo de S. Domingos, 16° 48' 42"
z tambem o rio de S. Domingos, ao rio de Ja
essem no campo de S. Domingos, aonde tinham
o da bocca do rio S. Domingos, de pequeno c
ro do Rio Fundo e S. Domingos da Saubara. V
vindo do pôrto de S. Filipe, vizinho a Noss
er a Fortaleza de S. Filippe, que na Bertio

Figure 3: Concordances retrieved from sentences that have the abbreviated form "S."

3.2.3. Specific patterns to extract new NEs of a given category

The use of generic search patterns in corpus processing tools shown in Section 3.2.2 is easily carried out by linguists and helps to get insight of the lexical patterning of historical corpora. Moreover, it is more useful to retrieve ambiguous classes of NEs, as illustrated in Figure 4 for the name of saints, since all of them are under letter S. However, we believe it is not efficient to enlarge the dictionary with new and different entries of a given category and its subespecifications. To perform this focused task, we are applying the same process defined for REPENTINO[14], a repository of NEs from modern Portuguese. This process is run in six steps, but the last one was not applied, for we adopted the NE taxonomy defined in HAREM: 1) choose a category for which you intend to search examples of entities; 2) decide which is the most appropriate strategy to search for the examples: a) by tag <INIT>, such as in Rio S. Francisco; b) by context, such as in "localizado na XXX" (located at XXX), which strongly suggests that "XXX" is a place; or c) by discriminating suffixes (modern organizations have in their names characteristic particles such as "Ltda."/Ltd. or "S.A"/Co.); 3) construct the respective pattern to be searched in a given corpus processor or to act as an independent program, and conduct the search; 4) validate manually the obtained candidates, considering the intended category; 5) include positive candidates in the repository; 6) if necessary, create a new category/subcategory, thus expanding the taxonomic classification system.
However, this process had to be adapted to historical corpora, because they have a large number of abbreviations and spelling variations related to both abbreviated words and expanded words. The requirement to accept a new NE from the corpus was that at least one of the components should be in the abbreviated form. We could not adopt the requirement of capitalization, since in historical corpora proper names are not always capitalized. To illustrate the adaptations of this procedure for retrieving new NEs from a corpus, we discuss a case study about hydronyms – names of rivers, streams, creeks, and brooks found in the *HDBP*. Flexor's dictionary (1991) contains 18 entries with the pattern Rio XXX/River XXX, but eight of them refer to the city of Rio de Janeiro (the rest are R^o da Ribr^a, R^o de Reg^o, R^o de S. Fran^co, R^o dos Alm^das, R^o G^de, R^o G^re, R^o Gdr^e, R^o Gr^de, R^o G^re e R^o P^do); there is nothing about Creek XXX (or its variants, brooks, streams), so we began with ten entries. The chosen search strategies were: pattern formed by tag <INIT> and contexts "naveg*"/navigate (on), that includes the several conjugations of the verb to navigate. However, words tagged as <INIT> could appear in their abbreviated or expanded form and, besides, we would have to deal with spelling variations and synonyms (see Table 5). To deal with spelling variations, we adopted two resources: the *HDBP* dictionary of spelling variants, created according to the SIACONF methodology proposed in Giusti et al. (2007)[15], which employs 43 transformation rules to cluster variants under one orthographic form, and the Philologic resource of searching for similar patterns, which uses AGREP. The *HDBP* dictionary of spelling variants has 18,082 clusters, totaling 41,710 variants. In spite of producing false-positives,

[14] http://poloclup.linguateca.pt/repentino/
[15] Available at http://moodle.icmc.usp.br/dhpb/siaconf.tar.gz

AGREP helps to complete variants resulting from SIACONF. To deal with synonyms of river, we used the Brazilian Portuguese Electronic Thesaurus TEP (Greghi et al., 2002).

Searching patterns (63)	Sources (5)	Right Occurrences (112)
rio	river	79
arroio, córrego, corrente, regato, regueira, regueiro, ribeirão, ribeiro, riacho, rio, veia, veio	synonyms	13
arroyo, corrego, corego, corgo regueyro, ribeiraõ, ribeyrão, ribeyraõ, rybeirão, rybeyrão, rebeirão, rebeyrão, ribeirao, ribeiro, ribeyro, rybeiro, ribejro, rjbeyro, rybeyro, riaxo, ryo, rjo, rrio, veya, veyo	spelling variants	7
c^te, cor, cor^e, cor^te, corr^e, corr^te, cort^e, crr^e, curr^te, r^bro, r^o, r^ro, reb^o, rib.^ro, rib^o, rib^ro, riber^o, ribr, ribr^o, ryb^o, ryb^ro, rybr^o, r^bro, r , r\.^o	abbreviations from Flexor (1991)	11
naveg*	context	2

Table 5: Searching patterns for hydronyms

The manual validation is the slowest step (we ckecked 27,808 ocorrencies in 160 minutes – 1,100 checkings per hour), but easier in concordancers, since the pattern formed by abbreviations stands out, which facilitates checking. As a result of this case study we have now 122 abbreviations under category LOCAL, specifically rivers and words related to watercourses, displaying their morphology in this semantic group in the *HDBP* corpus. Some examples are: Ribeyrão de N. Sr.^a do Carmo/Ribeirão de Nossa Senhora do Carmo; Corgo de S. Gonçalo/Córrego de São Gonçalo; rib.^o do Tombadouro, ribeirão do Tombadouro; coRego Ant.^o da Silua, Córrego Antonio da Silva; Rio M.^el Alves, Rio Marechal Alves; R^o doce, Rio doce.

4. Applications of the dictionary of abbreviations

An example of use of the *HDBP* dictionary of abbreviations is the application of UNITEX together with the software Dicionário[16] (Muniz et al., 2005) to assist lexicographers in manually identifying possible canonical forms for an abbreviation or for expanding an abbreviation. Using the concordancer shown in Figure 4, a linguist may find examples of abbreviations in the corpus, but may not be aware of the possible expanded forms for a given abbreviation. On using the software Dicionário together with the concordancer, the abbreviations can be quickly identified and associated with their possible canonical forms and categories. In the context of information retrieval for historical documents, it may be useful to gather, for example, texts reporting certain facts that happened in a certain place. If the index is the expanded abbreviation, we can easily find all abbreviations for a word, such as Bahia (a Brazilian city in the northeastern coast of Brazil). Next, we can locate passages in the corpus related to those abbreviations.

Our dictionary of abbreviations was designed to recognize large patterns of complete abbreviations. It also includes a specific tag for dealing with jobs/professions and titles and forms of address, such as capitão (captain), frei (friar), promotor (prosecutor), Ilustríssimo (Most Illustrious/Honorable), Dom (Don), Majestade (Majesty), Senhor (Sir), and family relations, such as cunhada (sister-in-law), primo (cousin). In linguistic research, it is very important to know whom the text is talking about or whom it is talking to. If we can determine the authorities that are being addressed in a specific text, we can identify the words used in that specific level of formality, given that a letter written to an ordinary person does not contain the same words and level of formality as one written to a monarch, and this is possible because we used NEs and other specific tags.

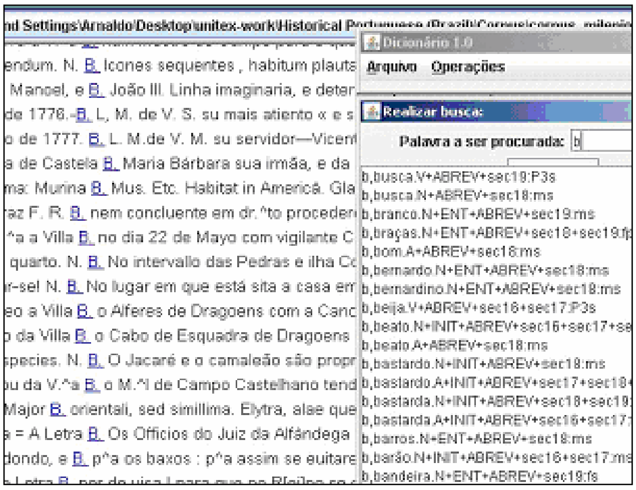


Figure 4: Search for pattern “b” in UNITEX; the program Dicionário helps the manual disambiguation of the abbreviation “b” (in the top right corner). All subcategorization of NEs will be included later.

5. Conclusions and Future Work

To sum up, the *Historical Dictionary of Brazilian Portuguese* is not only a pioneer project, but also a fundamental tool for recapturing and registering the country’s early history through its vocabulary. The compilation of a corpus of historical texts is therefore a crucial step to achieve such aim, since it allows researchers to retrieve the lexicon of a given period. The lexical, morphological, syntactic, and typographic characteristics identified in these texts have been the object of study of various members of our team, which includes philologists, linguists and computer scientists. Among the peculiarities of historical texts, the abbreviated forms pose a special challenge. In addition to their high frequency and ambiguity, a researcher is also faced with the fact that, as far as historical

[16] The software Dicionário is a Java application that handles any dictionary compacted in the DELA format, and allows searching for inflected words.

documents are concerned, there are no standard graphic forms, and abbreviations reflect this inconsistency, displaying a large number of variations. Taking this fact into account and to make a lexicographer's task feasible, special attention was given to abbreviations. An electronic dictionary of abbreviated forms is being built based on printed resources, using the DELA format, which allows us to categorize each new entry morphosyntactically, semantically and pragmatically. New NE categories of abbreviations were found using semantically categorized abbreviations, UNITEX graphs and regular expressions to examine the vicinity of abbreviated forms. Since the process to expand abbreviations demands considerable expertise, these automata and regular expressions were created to recognize only larger patterns of abbreviations that are NEs, spelling variations and synonyms of NEs or the same pattern that has different types of NEs, given that context will provide meaning. With regard to enlarging the dictionary of abbreviations, we focused on a specific NE category (places), subcategorizing it further (hydronyms). This experiment provided us with evaluation data with regard to time spent and productiveness rate of the semi-automatic approach we decided to adopt to guarantee high accuracy for the classification process. We concluded that this approach is worth pursuing once we need to guarantee a high precision classification. In the future, we intend to make this corpus and the dictionary of abbreviations available for those studies on history to which correct NE classification is crucial and mainly as a resource for NE recognition systems.

6. Acknowledgments

The authors are grateful to CNPq (Brazil) for supporting this research.

7. References

Baptista, J., Batista, F.; Mamede, N. (2006). Building a Dictionary of Anthroponyms. In *Computational Processing of the Portuguese Language*, 7th International Workshop, PROPOR 2006, R. Vieira et al. (eds.) Proceedings. Lecture Notes in Computer Science 3960, Berlin: Springer, pp. 21--30.

Crane, G., Jones, A. (2006). The challenge of Virginia banks: an evaluation of named entity analysis in a 19th-century newspaper collection. In *Proceedings of the 6th ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 31--40.

Dannélls, D. (2006). Automatic Acronym Recognition. In *EACL 2006, 11th Conference of the European Chapter of the Association for Computational Linguistics*. pp. 167--170.

Dipper, S.; Faulstich, L.; Leser, U.; Ludeling, A. (2004). Challenges in modelling a richly annotated diachronic corpus of German. In *Proceedings of the Workshop on XML-based Richly Annotated Corpora*. pp. 21--29.

Flexor, M. H. (1991). *Abreviaturas - Manuscritos dos Séculos XVI ao XIX*. 2nd ed. São Paulo: UNESP. 468 p.

Friburger, N. (2002). Reconaissance automatique de noms propres: Application à la classification automatique de textes journalistiques. Thèse (doctorat). Université de Tours. Tours.

Giusti, R., Candido Jr, A., Muniz, M., Cucatto, L., Aluísio, S. (2007). Automatic detection of spelling variation in historical corpus: An application to build a Brazilian Portuguese spelling variants dictionary. In *Proceedings of the Corpus Linguistics 2007 Conference*, Matthew Davies, Paul Rayson, Susan Hunston, Pernilla Danielsson (eds.).

Greghi, J. G., Martins, R. T., Nunes, M. G. V. (2002). Diadorim: a Lexical database for Brazilian Portuguese In *Proceedings of the International Conference on Language Resources and Evaluation* LREC 2002, Manuel G. Rodríguez and Carmem P. S. Araujo (Eds.), v. IV, pp. 1346--1350.

HaCohen, K. Y., Kass, A., Peretz, A. (2004). Baseline Methods for Automatic Disambiguation of Abbreviations in Jewish Law Documents. In EsTAL: International Conference on Advances in Natural Language Processing N. 4, Alicante. *Lecture Notes in Computer Science*. 3230. Berlin: Springer, pp. 58--69.

Hong, Y., Hripcsak, G., Friedman, C. (2002). Mapping abbreviations to full forms in biomedical articles. *J Am Med Inform Assoc*. May–June, 9(3), pp. 262--272.

Muniz, M., Nunes, M. G. V., Laporte, E. (2005). UNITEX-PB, a set of flexible language resources for Brazilian. *In III Workshop em Tecnologia da Informação e da Linguagem Humana*, pp. 2059--2068.

Paixão de Sousa, M. C., Trippel, T. (2006). Metadata and XML standards at work: a corpus repository of historical Portuguese texts. In *Proceedings of V International Conference on Language Resources and Evaluation* (LREC 2006), Genoa, Italy.

Pakhomov, S. (2002). Semi-supervised Maximum Entropy-based Approach to Acronym and Abbreviation Normalization in Medical Texts. In *Medical Texts Proceedings of ACL 2002*. pp. 160--167.

Paumier, S. (2006). Manuel d'utilisation du logiciel UNITEX. IGM, Université Marne-la-Vallée. Available at http://www-igm.univ-mlv.fr/~unitex/ManuelUnitex.pdf

Ranchhod, E. M. (2001). O uso de dicionários e de autômatos finitos na representação lexical das línguas naturais. In *Tratamento das Línguas por Computador. Uma Introdução à Lingüística Computacional e suas Aplicações*, E. Ranchhod (ed.), Lisbon: Caminho, pp. 13--47.

Rydberg-Cox, J. A. (2003). Automatic disambiguation of Latin abbreviations in early modern texts for humanities digital libraries. In *Proceedings of JCDL*, 03, pp. 372--373.

Sanderson, R. (2006). "Historical Text Mining", Historical "Text Mining" and "Historical Text" Mining: Challenges and Opportunities. Talk presented at the Historical Text Mining Workshop, July 2006, Lancaster University, UK.

Schwartz, A. M., Hearst, M. (2003). A simple algorithm for identifying abbreviation definitions in biomedical texts. In *Proceedings of the Pacific Symposium on Biocomputing (PSB) 2003*. pp. 451--462.

Silberztein, M. (2000). Intex: a FST toolbox. *Theoretical Computer Science*, 231, pp. 33--46.

A Conversant Robotic Guide to Art Collections

D. Vogiatzis*, D. Galanis†, V. Karkaletsis*, I. Androutsopoulos†‡, C.D. Spyropoulos*

*Institute of Informatics and Telecommunications, NCSR "Demokritos"
Athens, Greece

† Department of Informatics, Athens University of Economics and Business
Athens, Greece

‡ Digital Curation Unit, Research Centre "Athena"
Athens, Greece

Abstract

We present the dialogue system of a robot that has been developed to serve as a museum guide. The robot interacts with human visitors in natural language, receiving instructions and providing information about the exhibits. Moreover, being mobile, it physically approaches the exhibits it provides information about. Although the robotic platform contains many modules, including navigation, speech recognition and synthesis, our focus in this paper is the dialogue system, which supports the sessions between humans and the robot, as well as the natural language generation engine, which generates the text to be spoken. Both modules are closely interwined and depend on an ontology represented in OWL. The robot supports dialogues in both English and Greek.

1. Introduction

This paper presents the natural language interaction capabilities of a mobile robotic tour guide developed in the Greek project Xenios.[1] The robot adopts a simple finite-state architecture for dialogue management, which is complemented by multilingual natural language generation from OWL ontologies; the latter is used to produce personalised descriptions of exhibits, currently in English and in Greek. The robot guides the visitors to the premises of the Hellenic Cosmos, the cultural centre of the Foundation of the Hellenic World (FHW).[2] The vision and navigation components of the robot were developed by the Foundation of Research and Technology Hellas.[3]

Robots are being deployed experimentally as tour guides in science parks and museums (Bennewitz et al., 2005; Chiu, 2004; Thrun et al., 1999); however, the issue of natural language has been somewhat neglected. Most of the robotic platforms acting as guides rely on prerecorded texts, which are spoken by a speech synthesiser. In contrast, in our work the focus is on dynamic text generation from domain ontologies, in the context of spoken dialogues. Spoken Dialogue Systems (SDS) typically involve speech recognition, dialogue management, language interpretation, language generation, and speech synthesis. In addition, they require domain specific knowledge and linguistic resources (e.g., domain lexicons). CSLU[4], Dipper[5] and Trindikit[6] are three representative examples of software platforms that support the development of SDSs.

SDSs are divided into three major categories, depending on who controls the dialogue (McTear, 2004). In system initiative SDSs, the system asks a sequence of questions to elicit the required parameters of the task from the user. In user initiative SDSs, the dialogue is controlled by the user, who asks the system questions in order to obtain information. In mixed initiative SDSs, the dialogue control alternates between the two participants; the user can ask questions at any time, but the system can also take control to elicit required information or to clarify ambiguous information. In all SDSs, a dialogue management module is required to determine what questions the system should ask, in what order and when, as well as to anchor user utterances to the discourse context (e.g., resolve referring expressions). There are three main categories of dialogue management techniques: finite state based, frame based, and plan based. The most common category are the state based ones, which use graphs. Dialogue is modeled as a network of states connected by edges. At each state, the system can perform one of the following:

1. ask the user for specific information, possibly listing expected answers (in effect offering menu options),
2. generate a response to the user, or
3. access an external application.

In Xenios, we use system initiative dialogues, with finite state dialogue management, and particular options offered to the user at each state. The advantages of this technique are faster development and more robust systems, at the expense of limited flexibility in the dialogue structure.

Recent research has started to examine the efficient use of domain ontologies in dialogue systems (Milward and Beveridge, 2003). For instance, the domain specific lexicon and the grammar of the automatic speech recognition (ASR) component can be partially derived from the ontology. Furthermore, the natural language generation component (NLG) can generate descriptions of the ontology's instances or classes, as in our case. The ontology can also be

[1] http://www.ics.forth.gr/xenios/description.html
[2] http://www.fhw.gr/index_en.html
[3] http://www.forth.gr/
[4] http://cslu.cse.ogi.edu/toolkit/
[5] http://www.ltg.ed.ac.uk/dipper/
[6] http://www.ling.gu.se/projekt/trindi/trindikit/

used during language interpretation, for example to locate properties the user's utterances may refer to.

An entire strand of work in Natural Language Generation (NLG) has focused on producing textual descriptions of classes and instances (entities) of domain specific ontologies (Reiter and Dale, 2000). A representative example is ILEX (O'Donnell et al., 2001), which was mainly demonstrated with museum ontologies. In more recent work, a multilingual extension of ILEX was developed in the context of the M-PIRO project (Isard et al., 2003; Androutsopoulos et al., 2007). The system was tested in several domains including museum exhibits and computing equipment. Similar systems have also been built to automatically produce medical reports (Bontcheva et al., 2004) and drug descriptions (Bontcheva and Wilks, 2004).

The rest of this paper is organised as follows: the architecture of the dialogue system is presented in section 2; in section 3, we present the resources that are used in the dialogue system; then, in section 4, we present the natural language generation module; in section 5, we provide samples of a visitor's interaction with the robot; the paper ends with an overview and conclusions in section 6.

2. Architecture of the Dialogue System

The dialogue system consists of *resources* and *modules* (see Fig. 1). The modules are the *Dialogue System Manager (DSM)*, the *Natural Language Generation Engine (NLG)*, the *Automatic Speech Recognition (ASR)* engine, the *Text To Speech (TTS)* synthesiser and the *Gesture Recogniser*. The latter can detect a set of three human gestures (yes, no, quit), and it is not described here. As TTS we have used the Demosthenes speech synthesiser (Xydas and Kouroupetroglou, 2001). As ASR we have used the BabEar system from Acapela[7], which also provides natural language understanding facilities; the latter are based on a domain specific lexicon and a grammar.

The dialogue system manager (DSM) is the "actor" of the whole dialogue system, in the sense that it is the module that invokes and coordinates all the other modules. The DSM initiates the dialogue, and expects the visitor to select an option from the ones offered. In order to decide the next dialogue state and the text it will utter (through the TTS unit), it takes into account the *dialogue model*, the *interaction history* of the user, as well as the *location* of the robot. All the above contribute into creating more natural dialogues. Finally, there is the communication server, which enables the inter-module communication.

The resources are: the *dialogue model*; the *resources of the NLG*, which are discussed separately below; user models, including databases recording the interaction history of each user; and some databases that hold canned text and other information used during the dialogue. The modules are not domain specific; consequently, they can be easily transferred to another domain, with the exception of the ASR, which needs adjustments. On the other hand, the resources are domain specific and must be designed from scratch for each domain.

[7] http://www.acapela-group.com/index.asp

3. Resources of the Dialogue System

Dialogue Model

The dialogue model is based on a finite state automaton. At each state, the robot utters a particular question or statement. The transitions from a state represent the possible options offered to the human at that state. The dialogue model supports three sessions: *visitor welcome*, *tour* and *dialogue termination*. The dialogue model was developed especially for museums that host art collections, and are equipped with show rooms, where informative movies can be played.

In Fig. 2 we present an overview of the dialogue model. The first session (topmost node), captures the part of the dialogue session from the moment the visitor approaches the robot, till the moment he validates his ticket (validation occurs through a bar code reader, carried by the robot). The bottommost node represents the termination of the dialogue session. The second session (mapped to nodes 2, 3 and 4) concerns the presentation of the buildings and programmes of FHW to a visitor. The robot can talk about particular programmes, offer tour guidance, or do both.

The middle level nodes do not correspond to individual states; instead each one corresponds to approximately 10 states. Moreover, in each state there is a template which combines canned text with calls to the NLG in order to produce the final text to be uttered by the TTS.

Domain Ontology

The ontology represents the buildings and rooms of FHW, as well as the educational programs that are offered. The information about the buildings and the rooms is essential to the robot's navigation, to provide guidance instructions to visitors, and to relate educational programmes with their physical location. For instance, the following ontology extract includes the class `Area`, as well as the entities that belong to that class. For instance, `first-dome` and `ground-dome` represent the first and second floor of the dome (a semi-spherical movie projection hall).

```
Area
  ground-dome
  first-dome
  screening-room
  virtualReality-area
  basement1-building1
  basement2-building1
  first-building1
  ground-building1
```

The next ontology extract represents two types of programs that are offered, in particular there are documentaries about Miletus and the Black sea, as well as educational programs about the Ancient Agora and the ancients' contribution to mathematics.

```
Programme
  Documentary
    miletus
    black sea
  Educational
    ancientAgora
```

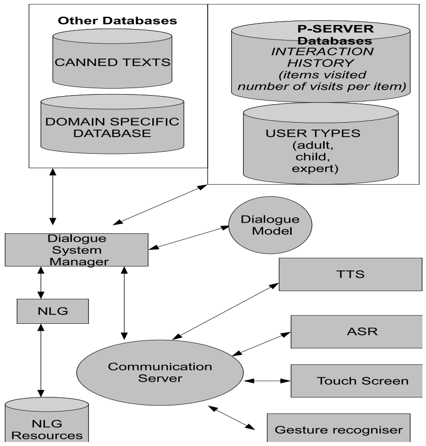


Figure 1: System Architecture

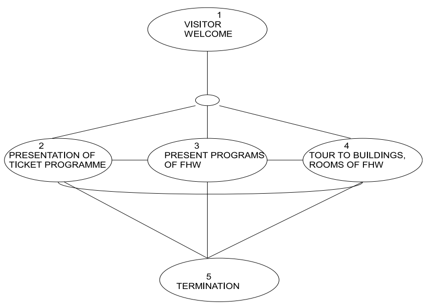


Figure 2: Dialogue Model

`mathematics`

The property `takes-place-at`, which is assigned to every programme, takes as its values entities of the `Area` class; for instance, <`miletus`, `takes-place-at`, `screening-room`>. Summarising, the ontology describes all the buildings and rooms where educational activities take place, as well as a significant part of the programmes that are currently offered.

The ontology can be updated by adding visiting areas, new exhibits, or by updating information on already existing areas and exhibits. The use of the ontology enables the system to describe the newly added or updated objects without further configuration. The ontology is authored in OWL,[8] the Semantic Web standard language for specifying ontologies, using the ELEON editor (Bilidas et al., 2007). Furthermore, ELEON allows annotating the ontology with the linguistic resources required by the NLG engine. User types (e.g., adult, child, expert) can also be defined with ELEON; we discuss user types below. The linguistic resources of the OWL ontology are exported by ELEON in the RDF format the NLG module requires.

The ontology was authored by a group of people from FHW. The group involved archaeologists that provided the initial information in the form of a collection of documents; this collection had been used before to produce paper leaflets and other promotional material. The same group extracted the most important paragraphs from the collection. The information of the paragraphs was then entered in ELEON, with additional information obtained from an architectural plan. Furthermore, FHW edited the contents of the canned text database, by using a simple editor developed for this purpose.

User Types

User types such as adult, child, expert, are defined with the aid of the ELEON authoring tool. The user types are defined when creating the domain ontology, and do not change afterwards. User types are quite useful, since they permit extensive personalisation of the information that users receive (Androutsopoulos et al., 2007). Thus, user types determine the interest of the ontology facts (e.g., some facts about painting techniques may be too elementary for experts), as well as how many times a fact has to be repeated before the system can assume that a user of a certain type has assimilated it (e.g., how many times we need to repeat the duration of the Roman period). In addition, user types specify the appropriateness of linguistic elements (e.g., lexicon and microplans, see below), as well as parameters that control the maximum desired length of an exhibit description. Finally, different synthesiser voices can be chosen for different user types. Orthogonal to the user types is the multilingual support (English and Greek) for the lexical items and the microplans.

Interaction History and Application Databases

The interaction history records information showing the exhibits each user has seen, the information the system has conveyed, as well as the natural language expressions the system has used to convey the information. This allows the system to avoid repeating the same information and expressions. The interaction history is kept in a database that is controlled by the Personalisation Server, which can also produce statistical data. Moreover, it can act as a recommender system (based on past usage data) upon the arrival of a new user (Paliouras et al., 2006). There is also a *Canned Text Database*, which contains fixed text that will be spoken at the commencement, at the end, or at an intermediate stage of the visitor's interaction with the dialogue system. Canned texts also contain some string variables that are instantiated during the dialogue session. Finally, there is a *Domain Specific Database*, which in effect contains instances of the ontology, for example, particular buildings, programs and rooms. This information is extracted from the ontology that the NLG module uses (Galanis and Androutsopoulos, 2007).

4. The Natural Language Generation engine

The natural language generation engine of Xenios, called NaturalOWL, is heavily based on ideas from ILEX and M-

[8]http://www.w3.org/TR/owl-features/

PIRO.[9] Unlike its predecessors, NaturalOWL is simpler (e.g., it is entirely template-based) and it provides native support for OWL ontologies. Currently, the system supports English and Greek.

NaturalOWL adopts the typical pipeline architecture of NLG systems (Reiter and Dale, 2000). It produces texts in three sequential stages: *document planning*, *micro-planning*, and *surface realisation*. In document planning, the system first selects the logical facts (OWL triples) which will be conveyed to the user and it specifies the document structure. In micro-planning, it constructs abstract forms of sentences, then it aggregates them into more complex periods, and finally it selects appropriate referring expressions. In surface realization, the abstract forms of sentences are transformed into real text, and appropriate syntactic and semantic annotations can be added, for example to help the TTS produce more natural prosody. The system is also able to compare the described entity to other entities of the same collection (e.g., "Unlike all the vessels that you saw, which were decorated with the black-figure technique, this amphora was decorated with the red-figure technique.").

4.1. Document planning

Content Selection

When NaturalOWL is instructed to produce a description of a class or an entity, it attempts to locate all the logical facts of the OWL ontology (OWL triples of the form `<S, Property, O>`, where `S` and `O` are the semantic subject and object of the fact, respectively) that are relevant to that entity or class. First, it selects all the facts that are directly relevant. For example, in the case of the entity whose identifier is `exhibit24`, it first selects, among others, the following facts, which associate `exhibit24` with class `aryballos` and the entities `archaelogical-delos`, `iraion-delos` and `archaic-period`.

```
<exhibit24,rdf:type,aryballos>
<exhibit24,current-location,
    archaelogical-delos>
<exhibit24,location-found,iraion-delos>
<exhibit24,creation-period,archaic-period>
```

Then, the system selects iteratively facts that are indirectly relevant to the described entity or class. In our example, the second iteration would add facts like the following:

```
<archaic-period,covers,
    archaic-period-duration>
<aryballos,rdfs:subclassOf, vessel>
```

The final set of selected facts results by removing the already assimilated facts and then the facts with the lowest interest. The interest of each fact and the number of times a fact has to be mentioned in order to be considered as assimilated is specified in the user types (Androutsopoulos et al., 2007).

[9] See http://www.ltg.ed.ac.uk/methodius/ for information on METHODIUS, another descendant of M-PIRO's generator.

Text Planning

The selected facts of the first iteration are ordered based on a domain-specific partial order of their properties (e.g., `current-location` must be mentioned after `creation-period`). More specifically, the partial order is created with the help of the ELEON tool, which allows the users to assign to each property an order score, an integer from 1 to 10. Smaller order scores indicate that facts with those properties should be mentioned earlier in the generated description. The order is stored in the RDF annotations of the ontology. The selected facts of the second or subsequent iterations are placed immediately after the corresponding facts of the first iteration.

4.2. Micro-planning

Abstract sentence forms

The RDF annotations of the ontology map each OWL property to one or more natural language templates. The templates consist of a list of slots and instructions showing how to fill them in. Each slot can be filled in with any of the following:

- A referring expression pointing to the semantic subject of the fact.
- A fixed string. If the string is a verb, it is specially tagged along with tense and voice. Prepositions are also tagged. These tags are used in aggregation.
- A referring expression pointing to the semantic object of the fact.

The final sentences are constructed by concatenating the slot values. For example, for the `current-location` property, the following microplan is defined:

```
Microplan(MicroName: "templ1",
    ForProperty="current-location") ->
Slot(Type:String, Value: "today")
Slot(Type:Owner, Case: nominative)
Slot(Type:Verb, Value: "is exhibited",
  Voice: Passive, Tense: present)
Slot(Type:Prep, Value: "in")
Slot(Type:Filler, Case: accusative)
```

If the above microplan is applied to the fact `<exhibit24, current-location, archeological-delos>`, the abstract sentence "*today* `Ref(Entity: exhibit48, Case: nominative)` *is exhibited in* `Ref(Entity: archaelogical-delos, Case: accusative)`" is produced. The `Ref` tags represent the referring expressions which will be produced in a following stage.

Aggregation

The produced abstract sentences are then aggregated into longer sentences using a set of aggregation rules similar to M-PIRO's (Melengoglou, 2002). For example, NaturalOWL's rules aggregate phrases that share the same semantic subject (e.g., by adding the connective "and"), or pairs of phrases where the semantic object of the first one is the same as the semantic subject of the second one and the second phrase's verb is "to be", as illustrated below.

```
This aryballos was found in the Heraion of
Delos. It is currently exhibited in the
Archaeological Museum of Delos.
->
This aryballos was found in the Heraion of
Delos and  it is currently exhibited in the
 Archaeological Museum of Delos.

This is an aryballos. An aryballos is a kind
of vessel.
->
This is an aryballos, a kind of vessel.
```

Referring expressions generation

In this stage, the system generates the appropriate referring expressions. When an entity or class is introduced for the first time, its name (e.g., "*Bancroft Chardonnay is a . . .* ") or the name of its class is used (e.g., "*this is an aryballos*"). In all subsequent references, an appropriate pronoun is used (e.g., "*it* was found in the Heraion of Delos"). If the focus (semantic subject of facts) shifts to another entity or class, again the name or the class of the new focus is first used, followed by pronoun references. When the focus returns to the original entity, a demonstrative (e.g., "*this aryballos* is made of . . . ") or the name of the original entity is used.

4.3. Surface Realization

Finally, in surface realization the system simply concatenates the slot values of the sentence specifications. In addition, syntactic or sematic markup can be added, as already mentioned.

5. Sample Interaction

Below we present an excerpt from a dialogue between the robot and a visitor. In the robot's utterances, the words that appear in italics are generated by the NLG module, whereas the other words are canned texts, with some intervening values of variables. As it can be seen, each robotic utterance is a mixture of dynamically generated and canned text, and as the dialogue progresses, the robot traverses the states of the dialogue model.

Robot Ok. Let us talk about the programmes of the Hellenic World. There are *6* programmes. The following: *a programme about the Ancient Agora of Athens, a documentary about the Olympic Games, ...* Please tell me which programme you are interested in.

Human I am interested in the Ancient Agora of Athens.

R Right. *This is an educational programme. Its title is: "In the Ancient Agora of Athens". It uses multimedia technologies and it is hosted in building 1. Its subject is history and it was established in 2007. It is addressed to children from 9 to 15 years old.* Would you like me to guide you to the place of the programme?

The following excerpt illustrates a case where the robot does not understand the user's utterance:

R Which building would you like to visit? I remind you that the Hellenic World has the following buildings: *building 1, building 5, building 7, Tholos, Research Centre.*

H The first one.

R I am sorry! I did not understand. I asked you which building you would like to visit. The answers I expected were: *building 1, building 5, building 7, Tholos and Research Centre*. Please select one or use the touch screen.

6. Conclusions and Future Work

We have described the dialogue system of the Xenios robotic guide. The dialogue system comprises many modules and resources. The central module is the dialogue manager that supports robot initiative dialogues. During human-robot interaction, the robot's utterances are a mixture of canned and dynamically generated text. Dynamic text generation proceeds in three stages: document planning, microplanning, and surface realisation. One of the most important resources of the system is the enriched domain ontology, which is authored by using the ELEON ontology editor. The ontology comprises descriptions of the FHW exhibits in OWL, as well as linguistic and user type information in RDF. The system supports both English and Greek.

Although an evaluation of the robotic guide has not been performed yet in real conditions, an initial version of the dialogue system has already been examined by FHW. In particular, the dialogue model, the richness of the generated text by the NLG, and the ontology are currently under revision based on user interaction data.

In the future we envisage the integration of an affective unit into the dialogue system. There is considerable research that suggests that user emotion recognition and robotic emotion expression may lead to more natural forms of communication (André et al., 2004).

Acknowledgements

This work was carried out in the context of project XENIOS, which was co-funded by the Greek General Secretariat and the European Union.

7. References

E. André, L. Dybkjær, W. Minker, and P. Heisterkamp, editors. 2004. *Affective Dialogue Systems*. Springer.

I. Androutsopoulos, J. Oberlander, and V. Karkaletsis. 2007. Source authoring for multilingual generation of personalised object descriptions. *Natural Language Engineering*, 13(3):191–233.

M. Bennewitz, F. Faber, M. Schreiber, and S. Behnke. 2005. Towards a humanoid museum guide robot that interacts with multiple persons. In *Proceedings of the 5th IEEE-RAS International Conference on Humanoid Robots*.

D. Bilidas, M. Theologou, and V. Karkaletsis. 2007. Enriching OWL Ontologies with Linguistic and User-related Annotations: the ELEON system. In *Proceeding of International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE Computer Society Press.

K. Bontcheva and Y. Wilks. 2004. Automatic report generation from ontologies: the MIAKT approach. In *Proceedings of the 9th International Conference on Applications* *of Natural Language to Information Systems*, pages 324–335, Manchester, UK.

K. Bontcheva, V. Tablan, D. Maynard, and H. Cunningham. 2004. Evolving GATE to meet new challenges in language engineering. *Natural Language Engineering*, 10(3/4):349–373.

C. Chiu. 2004. *The Bryn Mawr Tour Guide Robot*. Ph.D. thesis, Bryn Mawr College.

D. Galanis and I. Androutsopoulos. 2007. Generating multilingual descriptions from linguistically annotated OWL ontologies: the NATURALOWL system. In *Proceedings of the 11th European Workshop on Natural Language Generation*, Schloss Dagstuhl, Germany.

A. Isard, J. Oberlander, I. Androutsopoulos, and C. Matheson. 2003. Speaking the users' languages. IEEE *Intelligent Systems*, 18(1):40–45.

M. F. McTear. 2004. *Spoken Dialogue Technology. Toward the Conversational User Interface*. Springer.

A. Melengoglou. 2002. Multilingual aggregation in the M-PIRO system. Master's thesis, School of Informatics, University of Edinburgh, UK.

D. Milward and M. Beveridge. 2003. Ontology-based dialogue systems. In *3rd Workshop on Knowledge and Reasoning in Practical Dialog Systems*.

M. O'Donnell, C. Mellish, J. Oberlander, and A. Knott. 2001. ILEX: an architecture for a dynamic hypertext generation system. *Natural Language Engineering*, 7(3):225–250.

G. Paliouras, A. Mouzakidis, C. Ntoutsis, A. Alexopoulos, and C. Skourlas. 2006. PNS: Personalized Multi-Source News Delivery. In *Proceedings of the 10th International Conference on Knowledge-Based & Intelligent Information & Engineering Systems (KES), UK*.

E. Reiter and R. Dale. 2000. *Building natural language generation systems*. Cambridge University Press.

S. Thrun, M. Bennewitz, W. Burgard, A.B. Cremers, F. Dellaert, D. Fox, D. Haehnel, C. Rosenberg, N. Roy, J. Schulte, and D. Schulz. 1999. MINERVA: A second generation mobile tour-guide robot. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*.

G. Xydas and G. Kouroupetroglou. 2001. The DEMOSTHENES Speech Composer. In *Proceedings of the 4th ISCA Tutorial and Workshop on Speech Synthesis*, Perthshire, Scotland.

A Semantic Wiki Approach to Cultural Heritage Data Management

René Witte, Thomas Gitzinger, Thomas Kappler, Ralf Krestel

Institut für Programmstrukturen und Datenorganisation (IPD)
Universität Karlsruhe (TH), Germany

Abstract

Providing access to cultural heritage data beyond book digitization and information retrieval projects is important for delivering advanced semantic support to end users, in order to address their specific needs. We introduce a *separation of concerns* for heritage data management by explicitly defining different *user groups* and analyzing their particular requirements. Based on this analysis, we developed a comprehensive system architecture for accessing, annotating, and querying textual historic data. Novel features are the deployment of a Wiki user interface, natural language processing services for end users, metadata generation in OWL ontology format, SPARQL queries on textual data, and the integration of external clients through Web Services. We illustrate these ideas with the management of a historic encyclopedia of architecture.

1. Introduction

The amount of publicly available knowledge increases faster than we can imagine—hence the term "Information Explosion" used by several authors (Lyman and Varian, 2003; Sweeney, 2001). With the barrage of newly created content—news, blogs, web pages, research papers—existing, "analog" documents and their users often receive less attention than the quality of the content deserves.

In this paper, we present the results from a project aimed at developing enhanced semantic support for users of textual cultural heritage data. A particular feature of our approach is the integration of different concerns into a single, cohesive system architecture that addresses requirements from end users, software engineering aspects, and knowledge discovery paradigms. The ideas were implemented and tested with a historic encyclopedia of architecture and a number of different user groups, including building historians, architects, and NLP system developers.

2. User Groups and Requirements

Nowadays, the baseline for cultural heritage data management of book-type publications is the production of a scanned (digitized) version that can be viewed and distributed online, typically with some kind of Web interface. Before we can deliver more advanced access methods, we have to be more precise about the targeted end users. Who needs access to heritage data, and for what purpose?

2.1. User Groups

Within our project, we had to consider the requirements from four different user groups; each of them having a different background and expectations concerning the management of historic textual data.

(1) Historians: Within this group, we target users that deal with historic material from a scientific motivation, namely, historians. They require an electronic presentation that provides for a direct mapping to the printed original, e.g., for citation purposes. Additionally, semantic analysis tools should support their work through the formulation and verification of hypotheses.

(2) Practitioners: Under this group, we are concerned with users that need access to the historic material for their contemporary work. In our example scenario, the handbook on architecture, these are today's architects that need information on the building processes and materials used, e.g., within a restoration project of an old building. Here, the historic material contains knowledge that is not readily accessible in modern sources. Another example for such a user group are musicians dealing with old music scores and their descriptions, or lexicographers analyzing documents for the development of dictionary entries.

(3) Laypersons: Historic materials are a fascinating source of knowledge, as they preserve information over centuries. Providing widespread online access to materials that are otherwise only available in a controlled environment to scientists due to their fragile nature is perhaps one of the greatest benefits of digitization projects.

(4) Computational Linguists: Similarly to practitioners, linguists are often interested in historic documents from a functional point of view. However, their domain focuses on the properties of the language and its development over time rather than the underlying domain of discourse. They also have particular requirements for corpus construction, access, and annotation to support automated NLP analysis workflows.

2.2. Detected Requirements

We can now derive a number of requirements a system needs to fulfill, based on the user groups defined above:

Web Interface. To make the historic data available over the Internet, and to provide easy access within a familiar metaphor, the system needs to support a Web interface. This concerns all user groups to various degrees, but in particular the historians and laypersons.

Annotation Support. Users working with the historic data from a scientific point of view—in particular group (1)—often need to comment, add, and collaborate on the historic data. This should be supported within the same interface as the primary (historic) data, to avoid unnecessary context and application switches for the end users. At the same time, these annotations must be maintained by the architecture on clearly separated layers, to keep the integrity of the historic data intact.

Corpus Generation. While a Web interface is helpful for a human user, automated analyses using NLP tools and frameworks (user group (4)) can be better supported with a corpus in a standard (XML-based) markup, since HTML pages generated through Web frameworks typically mix content and layout information (menus, navigation bars, etc.). Thus, the architecture should provide a separate corpus that is automatically derived from the historic data and contains appropriate markup (for headlines, footnotes, figure captions, etc.). Ideally, it should allow to cross-link entities with the Web interface.

NLP Services. For large collections of (historic) documents, manual inspection of all content or even a subset obtained through information retrieval (IR) is not feasible. Here, NLP analyses can deliver additional benefit to end users, in particular groups (1)–(3), by integrating NLP analysis services (and their results) into the overall architecture. It should allow the execution of any service, developed by user group (4), and also deliver the results back to the clients. Examples for such NLP services are summarization, index generation, or named entity detection.

Metadata Generation. While NLP results can be useful for a human user, we also need to support further automated analysis workflows. User group (2) in particular requires access to the historic data, as well as its metadata, from external tools and applications relevant for their domain. To support external access to metadata from many different clients, the architecture should be capable of generating standards-compliant data formats, such as RDF and OWL.

Application Integration. As pointed out in the last requirement, external applications should be provided with automated access to the historic data and its metadata. Generally speaking, this requires the introduction of a client/server model, where the communication, like the metadata format, should use open, established standards.

3. Related Work

Before we describe our approach in detail, we discuss related work relevant for the detected requirements.

The Cultural Heritage Language Technologies (CHLT) project (Rydberg-Cox, 2002; Rydberg-Cox, 2005) describes the use of NLP methods to help students and scholars to work with classic Greek and Latin corpora. Similar to our approach, collaboration is an important goal of the project. Not only for sharing metadata about the text itself, but also to offer users the possibility to annotate, comment, or correct the results of automated analysis. This metadata can also contain hyperlinks to connect related texts with each other. The importance of correct morphological analysis is stressed as a baseline technology for users in the humanities, a statement which is also reflected in our work by integrating a self-learning lemmatizer for the German language (Perera and Witte, 2005) for accurate index generation. Further processing in the CHLT project includes information retrieval and data visualization. Identifying keywords, clustering subsets of the data, and visualizing the resulting groups supports the users in grasping concepts or performing search. In contrast, our approach uses open, standardized data formats like an automatically populated ontology to facilitate searching and browsing through the corpus and a Wiki system to share information between users.

As outlined in (Mavrikas et al., 2004), access to cultural heritage data available in natural language can be facilitated using various NLP techniques. In the context of the Semantic Web, the proposed system extracts CH data from different sources in the Internet and processes the data afterwards. An ontology (Doerr, 2003) is used to organize the mined data. Templates are used to extract relevant information, and the use of multi-document summarization is also proposed, as a way to present relevant information in a condensed way to the user. Here, we present an actual implementation of a system addressing these problems and extend the use of ontologies to allow easy browsing and querying of the document content for different user groups.

Another approach based on the CIDOC-CRM[1] ontology is presented in (Généreux, 2007). The system described there consists of two parts, one for extracting CH knowledge from natural language texts and saving the information in the ontology format, and one for using natural language to query the database. The natural language is reformatted to a SPARQL query using WordNet. This approach, in contrast to our system, stresses more the search aspect to find relevant data and offers no further possibilities for collaboration or processing of the data.

In (Sinclair et al., 2005), a system is presented that enables the user to explore, navigate, link, and annotate digitized cultural heritage artifacts like videos, photos, or documents. The system also supports user-generated descriptions and content. The focus in this project lies on the integration of the different metadata formats of the source content, whereas we additionally focus on the processing and collaboration part.

From a technical perspective, semantic extensions to Wiki systems based on *Semantic Web* technologies like OWL ontologies and RDF are similar in that they provide the means for content structuring beyond the syntactical level. In these systems, the properties of and relations between objects can be made explicit, with the Wiki system "knowing" about them. This allows for automated processing of Wiki content, e.g., through software agents. Current implementations of these ideas can be found in systems like Semantic MediaWiki (SMW) (Krötzsch et al., 2006) or IkeWiki (Schaffert, 2006). It is important to note that these tools are different from and complementary to our approach: While in our context, the content of a Wiki is subject to semantic analysis via NLP methods (with the Wiki engine itself not needing to have semantic capabilities), semantic Wikis like SMW have explicit notational and internal semantic capabilities. Using a semantic Wiki in our system in the future would allow the Wiki engine itself direct access to the facts derived from semantic text analysis.

4. Semantic Heritage Data Management

In this section, we present our approach to cultural heritage data management, which integrates a number of different technologies in order to satisfy the requirements of the various user groups: (i) A Wiki user interface, (ii) text mining support using an NLP framework, (iii) Semantic Web ontologies based on OWL and RDF for metadata management,

[1] CIDOC Conceptual Reference Model, http://cidoc.ics.forth.gr/

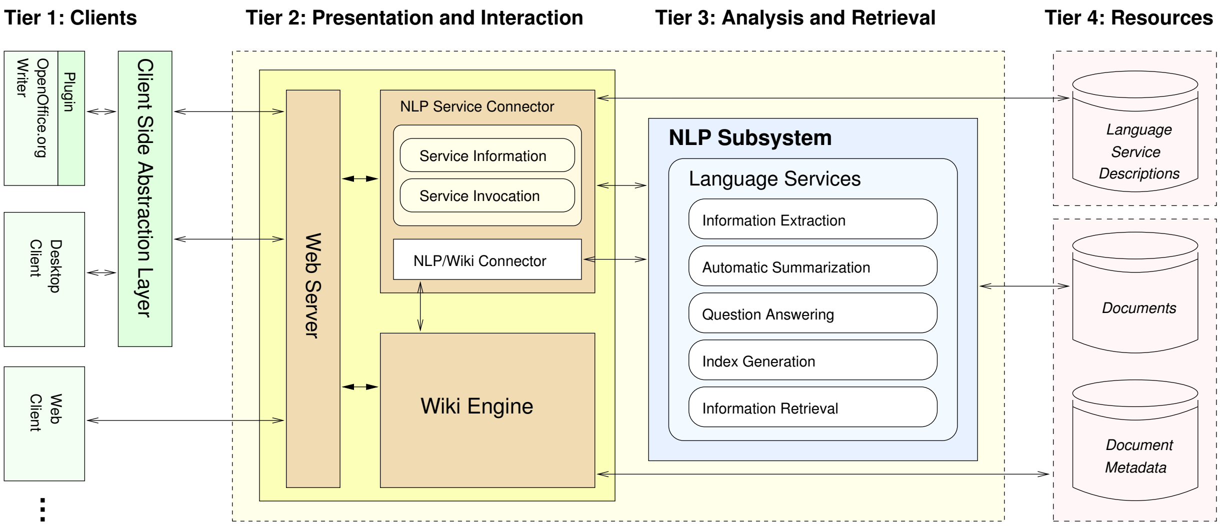


Figure 1: System architecture overview

and (iv) W3C Web Services for application integration. We first present an overview of our system in the next subsection. The various subsystems are illustrated using examples from a productive, freely accessible[2] Web resource built around the German *Handbuch der Architektur* (handbook on architecture) from the 19th century, described in detail in Section 4.2. The digitization process is described in Section 4.3. Necessary format conversions for the digital version are covered in Section 4.4. To support our user groups, we integrated several NLP analysis services, which are covered in Section 4.5. Finally, our semantic extensions for generating OWL/RDF metadata and application integration are covered in Section 4.6.

4.1. Architectural Overview

As stated above, our goal is the development of a unified architecture that fulfills the requirements (Section 2.2.) of the different user groups defined in Section 2.1., by integrating means for content access, analysis, and annotation.

One of the central pieces of our architecture is the introduction of a *Wiki* system (Leuf and Cunningham, 2001). Wiki systems provide the Web interface stipulated in our first requirement, while also allowing users to add meta-content in form of separate *discussion* or *annotation* pages. This capability directly addresses our second requirement, by allowing users to discuss and collaborate on heritage data, using an online tool and a single interface, while keeping the original data intact.[3]

Other clients, NLP services, and the actual content have to be integrated into this model. Figure 1 shows how these and the remaining components are systematically assembled to form the overall architecture of our system.

The architecture comprises four tiers. Tier 1 consists of clients that the users employ to access the system. Plug-in capable existing clients, like the OpenOffice.org application suite, can also be extended to be integrated with our architecture. New applications can have that functionality built in, like the "Desktop Client" depicted in the diagram. The "Client-Side Abstraction Layer" (CSAL) facilitates connecting clients by providing common communication and data converting functionality.

The clients communicate with a Web server on Tier 2, behind which we find the Wiki engine and a software module labeled "NLP Service Connector." The functionality of this module is offered as an XML Web service, as standardized by the W3C.[4] This means that there is a publicly accessible interface definition, written in the Web Service Description Language (WSDL), from which clients know how to use the offered functionality. The functionality itself is used through a Web service *endpoint*, to which the client sends and from where it receives messages. The main task of the NLP Service Connector is to receive input documents and have the NLP subsystem (Tier 3) perform various text analysis procedures on them. A sub-module of the NLP Service Connector, labeled "NLP/Wiki Connector," allows for the automatic retrieval, creation, and modification of Wiki content.

Finally, on Tier 4, we have metadata on the employed text analysis services (top), which the NLP Service Connector requires in order to operate these services. The bottom rectangle contains the documents maintained by the Wiki system as well as their metadata, which might have been provided by hand, or generated through automatic analysis methods.

4.2. Source Material

We implemented and evaluated the ideas described here for a particular set of historic documents: the German *Handbuch der Architektur*, a comprehensive multi-volume encyclopedia of architecture.[5] The full encyclopedia was written between the late 19th and early 20th century; It aimed to include all architectural knowledge at the time, both past and present, within the fields of architectural history, architectural styles, construction, statics, building equipment, physics, design, building conception, and town planning. The full encyclopedia comprises more than 140 individual

[2] See http://durm.semanticsoftware.info

[3] Assuming the Wiki has been properly configured for this scenario; the technical details depend on the concrete Wiki system.

[4] Web Services Architecture, http://www.w3.org/TR/ws-arch/

[5] Edited by Joseph Durm (⋆14.2.1837 Karlsruhe, Germany, †3.4.1919 ibidem) and three other architects since 1881.

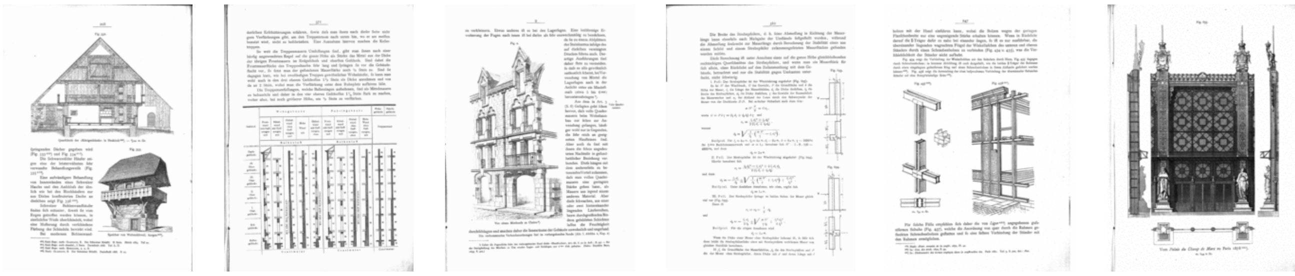


Figure 2: Source material examples: Scanned pages from *Handbuch der Architektur* (1900)

publications and contains at least 25 000 pages.

Due to the ambitious scope, the long publication process, and the limitations of the technologies available at that time, it is extremely difficult to gain an overview of a single topic. Information is typically distributed over several parts containing a number of volumes, which in turn are split into books. Most of these do not contain any kind of index. In addition, some of the volumes were edited and reprinted and a supplement part was added.

Due to funding limitations, we only dealt with a single volume[6] within the project described in this paper. However, the concepts and technologies have been developed with the complete dataset in mind.

4.3. Digitization and Error Correction

The source material is first digitized using specialized book scanners, producing a TIFF file for each physical page; in our case, with a grayscale resolution of 600dpi.

In a second step, the image files need to be converted to machine-readable text to support, amongst others, NLP analysis and metadata generation. We initially planned to automate this process using OCR software. However, due to the complex layout of the original material (see Figure 2), which contains an abundance of figures, graphs, photos, tables, diagrams, formulas, sketches, footnotes, margin notes, and mixed font sizes, as well as the varying quality of the 100-year old source material, this proved to be too unreliable. As the focus of this project was on developing enhanced semantic support for end users, not basic OCR research, we decided to manually convert the source material into an electronic document. This provided for not only a faster and more reliable conversion, but also accurately captured layout formation in explicit markup, such as footnotes, chapter titles, figure captions, and margin notes. This task was outsourced to a Chinese company for cost reasons; Manual conversion was performed twice to allow an automatic cross-check for error detection. The final, merged version contained only a very small amount of errors, which were eventually hand-corrected during the project.

4.4. Format Transformation and Wiki Upload

The digitized content was delivered in the *TUSTEP*[7] format. This content was first converted to XML, and finally to Wiki markup.

4.4.1. TUSTEP Format

TUSTEP is a toolkit for the "scientific work with textual data" (Uni, 2008), consisting of a document markup standard along with tools for text processing operations on TUSTEP documents. The markup is completely focused on layout, so that the visual structure of printed documents can be captured well. Structurally, it consists both of XML-like elements with an opening and closing tag, such as `<Z>` and `</Z>` for centered passages; and elements serving as control statements, such as `#H:` for starting text in superscript. The control statements remain in effect until another markup element cancels them out, such as `#G:` for adjusting the following text on the baseline.

TUSTEP predates XML, and while it is still in use at many universities, we found it makes automatic processing difficult. The control statements, for instance, make it hard to determine the range of text they affect, because their effect can be canceled by different elements. In addition, in the manual digitization process, markup was applied inconsistently. Therefore, we chose to first convert the data to a custom XML format, designed to closely match the given TUSTEP markup. This also enabled easier structural analysis and transformation of the text due to the uniform tree structure of XML and the availability of high-quality libraries for XML processing.

4.4.2. Custom XML

We developed a custom tool to transform TUSTEP data into XML. The generated XML data intends to be as semantically close to the original markup as possible; as such, it contains mostly layout information such as line and page breaks and font changes. Except for the exact placement of figures and tables, all such information from the original book is retained.

Parsing the XML into a DOM[8] representation provides for easy and flexible data transformation. The resulting XML format can be directly used for NLP corpus generation.

4.4.3. Wiki Markup

To make the historic data accessible via a Wiki, we have to further transform it into the data format used by a concrete Wiki engine. Since we were dealing with an encyclopedic original, we chose the *MediaWiki*[9] system, which is best known for its use within the *Wikipedia*[10] projects.

[6]E. Marx: *Wände und Wandöffnungen* (Walls and Wall Openings). In "Handbuch der Architektur," Part III, Volume 2, Number I, Second edition, Stuttgart, Germany, 1900. Contains 506 pages with 956 figures.

[7]TUebingen System of TExt processing Programs (TUSTEP), http://www.zdv.uni-tuebingen.de/tustep/tustep_eng.html

[8]Document Object Model (DOM), http://www.w3.org/DOM/

[9]MediaWiki, http://en.wikipedia.org/wiki/MediaWiki

[10]Wikipedia, http://www.wikipedia.org

A challenging question was how to perform the concrete conversion from content presented in physical book layout to Wiki pages. Obviously, translating a single book page does not translate well into a single web page. We first attempted to translate each book chapter into a single page (with its topic as the Wiki entry). However, with only 15 chapters in a 500-page book, the resulting Web pages were too long to be used comfortably in the MediaWiki interface. Together with our end users, we finally decided to convert each sub-chapter (section) into a single Wiki page, with additional internal structuring derived from the margin notes preserved by the manual conversion.
MediaWiki uses the markup language *Wikitext*, which was designed as a "simplified alternative to HTML,"[11] and as such offers both semantic markup, like headings with different levels, as well as visual markup, like italic or bold text. Its expressiveness is largely equal to that of HTML, despite the simplified approach, because it lets users insert HTML if Wikitext does not suffice.

Example: Footnote conversion. Footnotes were delivered in TUSTEP in the form #H:n#G:) for each footnote n. The markup indicates text being set to superscript (#H:), then back to the standard baseline (#G:). The footnote reference in the text and the anchor in the footnote section of a page have the same markup, as they look the same. The tool converting to XML locates footnotes using a regular expression, and creates <footnote to="n" /> resp. <footnote from="n">...</footnote> tags. Finally, the conversion to Wikitext transforms the references to <span id="fn8ref" />^{[[#fn8|8)]]}. The HTML sup tag sets the text as superscript, and its content is a link to the anchor "fn8" on the same page, with the link text simply being "8". The footnote itself is represented by <span id="fn8"/>''8)'' ... [[#fn8ref|^]]. We see the anchor linked to from the reference, and vice versa a link to jump back upwards to the reference.

4.4.4. Wiki Interface Features.

The conversion to Wikitext inserts further information for the Wiki users, such as links to scans of the original pages, and link/anchor combinations to emulate the page-based navigation of the book (see Figure 3). For instance, the beginning of page 211, which is indicated in TUSTEP by @@1@<S211><, looks as follows in the resulting Wikitext:

```
<span id="page10" />
'''Seite 211 ([[Media:S211_large.gif|Scan]])'''
[[Image:S211_large.gif|thumb|200px|Scan der
                   Originalseite 211]]
```

4.5. NLP Integration

One of the main goals of our work is to support the end users—groups (1) to (3)—with semantic analysis tools based on NLP. To make our architecture independent from the application domain (architecture, biology, music, ...) and their custom NLP analysis pipelines, we developed a general integration framework that allows us to deploy any kind of language service. The management, parametrization, and execution of these NLP services is handled in our framework

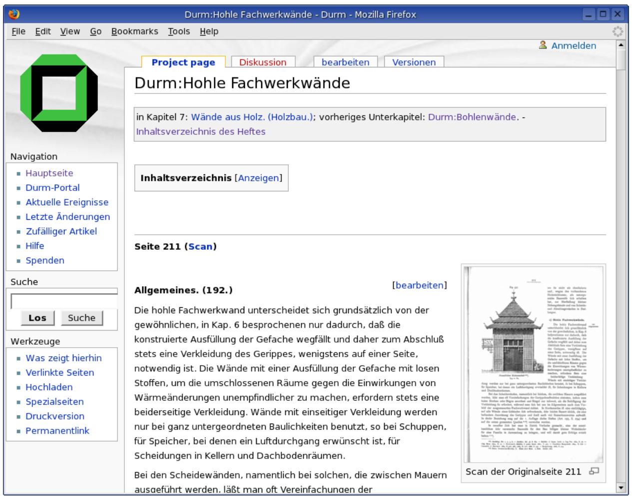


Figure 3: The Wiki interface integrating digitized text, scanned originals, and separate "Discussion" pages

(see Figure 1, Tier 3, "NLP Subsystem") by GATE, the *General Architecture for Text Engineering* (Cunningham et al., 2002). To allow a dynamic discovery of newly deployed language services, we added service descriptions written in OWL to our architecture (see Section 4.1.).
Language services should help the users to find, understand, relate, share, and analyze the stored historic documents. In the following subsections, we describe some of the services we deployed in our implementation to support users of the historic encyclopedia, including index generation, automatic summarization, and OWL metadata generation.

4.5.1. Index Generation

Many documents—like the discussed architectural encyclopedia—do not come with a classical back-of-the-book index. Of course, in the absence of an index, full-text search can help to locate the various occurrences of a single term, but only if the user already knows what he is looking for. An index listing all nouns with their modifiers (adjectives), with links to their locations of occurrence, can help the user find useful information he was not expecting, which is especially important for historical documents, which often contain terminology no longer in use.
For our index, we process all noun phrases found in the analyzed texts. For each noun phrase, we compute the lemma of the head noun and keep track of its modifiers, page number, and corresponding Wiki page. To deal with the problem of correctly lemmatizing historic terminology no longer in use, we developed a self-learning lemmatizer for German (Perera and Witte, 2005). Nouns that have the same lemma are merged together with all their information. Then, we create an inverted index with the lemma as the main column and their modifiers as sub-indexes, as shown in Figure 4. The generated index is then uploaded from the NLP subsystem into the Wiki through a connector ("NLP/Wiki Connector" in Figure 1).

4.5.2. Automatic Summarization

Large text corpora make it impossible for single users to deal with the whole documents in total. The sheer amount of information encoded in natural language in huge text collections poses a non-trivial challenge to information sys-

[11] Wikitext, http://en.wikipedia.org/wiki/Wikitext

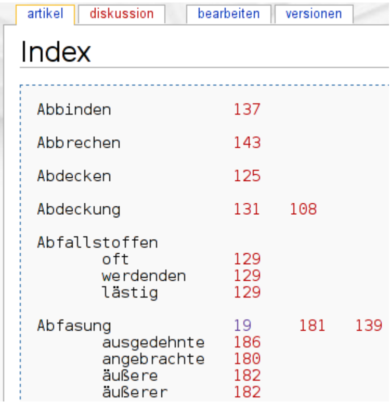


Figure 4: NLP-generated full text index, integrated into the Wiki interface (page numbers are hyperlinks to Wiki pages)

tems in order to adequately support the user. To find certain information, to get an overview of a document, or just to browse a text collection, automatic *summarization* (Mani, 2001) offers various methods of condensing texts.[12]

Short, headline-like summaries (around 10 words) that incorporate the most important concepts of a document or a Wiki page facilitate the search for particular information by giving a user an overview of the content at a glance. In addition, full-text summaries can be created for each page, e.g., with a length of 100 words or more. These summaries in free-text form can be read much more quickly than a full-length article, thereby helping a user to decide which Wiki pages he wants to read in full.

More advanced types of summaries can support users during both content creation and analysis. *Multi-document summaries* can combine knowledge from several pages within a Wiki or even across Wiki systems. *Update summaries* keep track of a user's reading history and only present information he has not read before, thereby further reducing the problem of information overload. And *focused summaries* enable the user to formulate a query (natural language questions) the generated summary focuses on. This is especially useful to get a first impression of the available information about a certain topic in a collection. In (Witte et al., 2005), we illustrate the usefulness of focused summaries for a particular architectural scenario.

4.5.3. Other NLP Services

The examples presented so far are by no means exhaustive. Depending on the type of data under investigation and the demands of the users concerned with their analysis (groups (1) and (2)), additional NLP services will need to be introduced. Due to our service-oriented approach (cf. Section 4.1.), new services can be added at any time, as they are automatically detected by all connected clients through the metadata repository, without any changes on the client side. Likewise, new user clients can be added dynamically to the architecture, without requiring any changes to the NLP server.

4.6. Semantic Extensions

The NLP analysis services introduced so far are aimed at supporting the user groups (1) and (3): Summaries, full-text

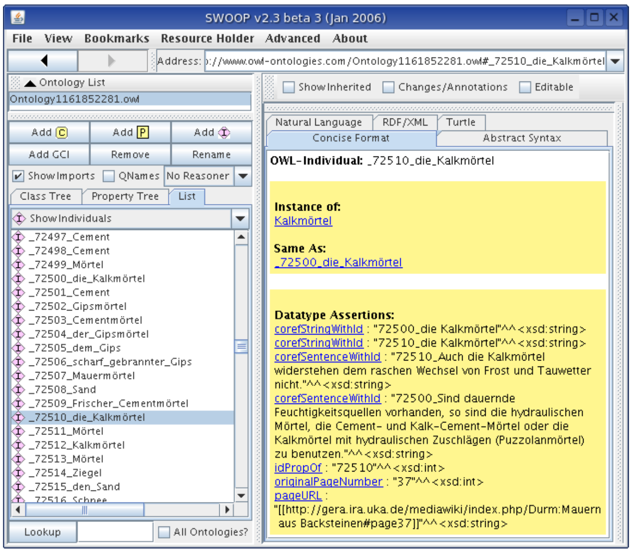


Figure 5: An ontology instance created through NLP

indices, and question-answering all produce new natural language texts, which are convenient for humans. But they are less useful for providing further automated access to the historic data, e.g., through desktop tools targeted at user group (2). In our example scenario, the architects need to integrate the historic knowledge "stored" in the encyclopedia within contemporary architectural design tools: While viewing a certain construction element, the relevant content from the handbook should be extracted and presented alongside other project information. This requires the generation of metadata in a machine-processable data format. In our architecture, this is provided through the NLP-driven population of formal ontologies. We discuss our ontology model in the next subsection, followed by a description of the automatic population process and the querying of the result format.

4.6.1. Ontology Model

Our NLP-generated metadata is formally represented using the *Web Ontology Language* (OWL),[13] which is a standard defined by the World Wide Web Consortium (W3C). Specifically, we use the sub-format OWL-DL, which is based on description logics (DL). OWL is also the foundation of the Semantic Web initiative, which allows us to immediately make use of a large variety of tools and resources developed for OWL-based information processing (editors, storage systems, query languages, reasoners, visualization tools, etc.).

Our ontology has two parts: a *document* ontology describing the domain of NLP (documents, sentences, NPs, coreference chains, etc.) and a *domain* ontology. While the document ontology is independent of the content in the historic documents, the domain ontology has to be developed specifically for their discourse domain. In our example, this ontology needs to contain architectural concepts, such as doors, walls, or windows. By combining both ontologies, we can run semantic *queries* against the ontology, e.g., asking for all sentences where a certain concept appears. The incorporation of CIDOC/CRM could extend our model in the future.

Document Ontology Model. Our document ontology models a number of concepts relevant for the domain of

[12] See, e.g., the *Document Understanding Conference* (DUC), http://duc.nist.gov

[13] OWL, http://www.w3.org/2004/OWL/

NLP. One of the main concepts is *document*, representing an individual text processed by an NLP pipeline, containing: the *title* of the document; its *source* address (typically a URL or URI); and a relation *containsSentence* between a document and all its *sentences*.
Likewise, sentences are also represented by an ontology class, with: the start and end position (*beginLocation*, *endLocation*) within the document, given as character offset; the sentence's *content*, stored as plain text, i.e., without additional markup; and a relation *contains* between a sentence and all *named entities* that have been detected in it.
Each of the named entities has, in addition to its ontology class, a number of additional properties: a unique id (*idPropOf*) generated for this instance; the page number (*originalPageNumber*), where the instance can be found in the (printed) source; and the full URL (*pageURL*) for direct access to the instance in the Wiki system.
Additionally, we can represent the result of the coreference resolution algorithm using the OWL language feature *sameAs:* If two instances appear in the same coreference chain, two separate ontology instances are created (containing different ids and possibly different page/URL numbers), but both instances are included in such a *sameAs* relation. This allows ontology reasoners to interpret the syntactically different instances as semantically equivalent. Additionally, a relation *corefStringWithId* is created for every entity in the coreference chain, refering to its unique id stored in the *idPropOf* property; and the content of the sentence containing the co-refering entity is stored in *corefSentenceWithId*.

Domain Ontology Model. In addition to the generic NLP ontology, a domain-specific ontology can be plugged into the system to allow further structuring of the NLP results. If such an ontology is developed, it can also be used to further facilitate named entity detection as described below.
In our approach, we rely on a hand-constructed ontology of the domain. This could be enhanced with (semi-)automatic *ontology enrichment* or *ontology learning*. In general, the design of the domain ontology needs to take the requirements of the downstream applications using the populated ontology into account.

4.6.2. Automatic Ontology Population

We developed an *ontology population* NLP pipeline to automatically create OWL instances (individuals, see Figure 5) for the ontology described above. An overview of the workflow is shown in Figure 6.
The pipeline runs on the XML-based corpus described in Section 4.4. After a number of standard preprocessing steps, including tokenization, POS tagging, and NP chunking, named entities (NEs) are detected using a two-step process. First, an *OntoGazetteer* (Bontcheva et al., 2004) labels each token in the text with all ontology classes it can belong to. And secondly, ontology-aware grammar rules written in the JAPE[14] language are used to find named entities (NEs). Evaluation of the correctness of the generated instances can be conducted using precision and recall measures (Maynard et al., 2006).

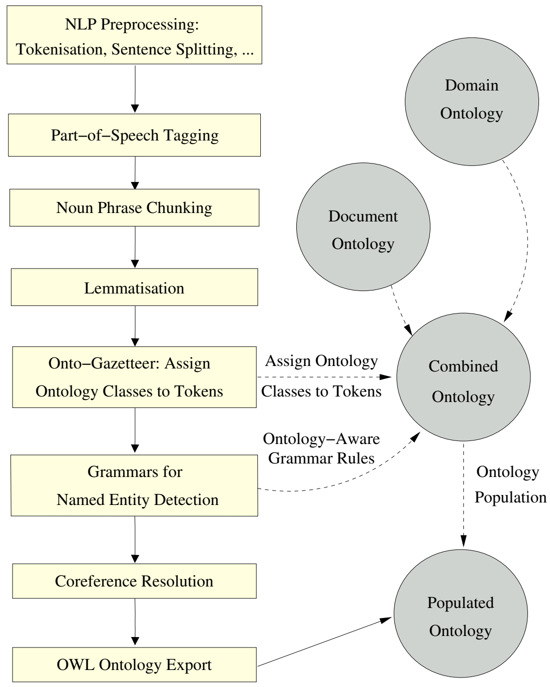


Figure 6: NLP pipeline for ontology population

Finally, the created instances are exported into the result ontology, combining a number of domain and document features. An example instance, of the ontology class *Kalkmörtel* (lime mortar), is shown in Figure 5.

4.6.3. Ontology Queries

The automatically populated ontology represents a machine-readable metadata format that can be *queried* through a number of standardized ontology query languages, such as SPARQL.[15] Queries are a much more expressive paradigm for analyzing text mining results than simple IR; in particular, if a domain model is available, they allow queries over the analyzed documents on a semantic level.
An example SPARQL query is shown in Figure 7. The query shown in the left box represents the question *"Which building materials are mentioned in the handbook together with the concept 'Mauer' (wall), and on which page?"* The result of this query (executed using Protégé[16]), is shown on the right. The first column ("type") shows what kind of entity (stone, plaster, concrete, . . .) was found, i.e., a sub-class of "material" in the domain ontology. The results can now be directly inspected by the user or used for further automatic processing by another application.
More abstractly speaking, ontology queries support automated problem-solving using a knowledge base. A user of our system, like a historian, might want to formulate hypotheses concerning the source material. Translated into an OWL query, the result can be used to confirm or refute the hypothesis. And as a standardized NLP result format, it also facilitates direct integration into an end-user application or a larger automated knowledge discovery workflow.

4.6.4. Application Integration

The populated ontology also serves as the basis for our final requirement, application integration. With "application" we

[14] Java Annotations Pattern Engine, a regular expression-based language for writing grammars over document annotation graphs.

[15] SPARQL, http://www.w3.org/TR/rdf-sparql-query/
[16] Protégé, http://protege.stanford.edu/

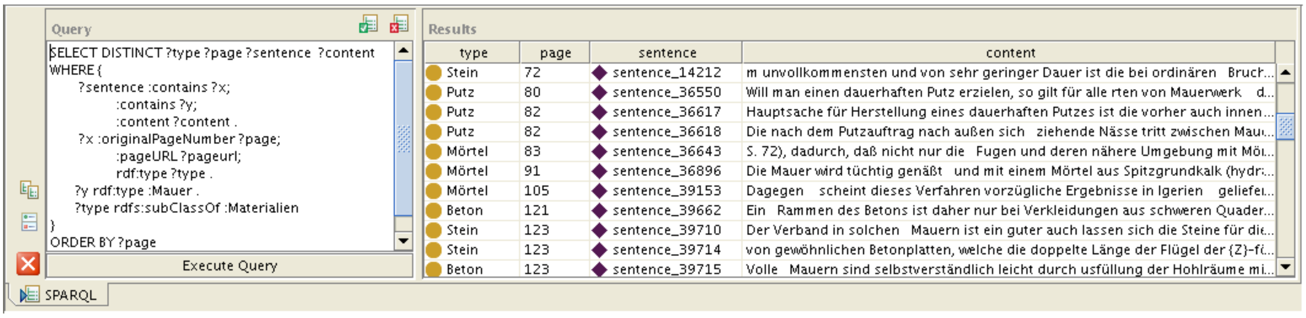


Figure 7: Posing a question to the historic knowledge base through a SPARQL query against the NLP-populated ontology

mean any end-user accessible system that wants to integrate the historic data within a different context. For example, in a museum setting, such an application might allow a visitor to access content directly relevant to an artifact. A lexicographer might want to query, navigate, and read content from historical documents while developing a lexical entry. And in our application example, an architect needs access to the knowledge stored in the handbook while planning a particular building restoration task. Here, construction elements displayed in a design tool (such as *window* or *window sill*) can be directly connected with the ontological entities contained in the NLP-populated knowledge. This allows an architect to view relevant content down to the level of an individual construction element using the named entities, while retaining the option to visit the full text through the provided Wiki link.

5. Summary and Conclusions

To support users in the cultural heritage domain, a precise analysis of the different user groups and their particular requirements is essential. In this paper, we present a holistic approach based on a unified system architecture that highlights the many inter-dependencies in supporting different groups with particular features, aimed at different use cases: *Historians* have the support of NLP analysis tools and a user-friendly Web-based access and collaboration tool build around a standard Wiki system. *Laypersons* also benefit from these user-friendly features, while *practitioners*—in our scenario building architects—can additionally use NLP-generated ontology metadata for direct application integration. Finally, our approach also supports computational linguists through corpus construction and querying tools.

The experience from the implemented system using the example of a historical encyclopedia of architecture demonstrates the usefulness of these ideas. Finally, providing a machine-readable knowledge base that integrates textual instances and domain-specific entities is consistent with the vision of the Semantic Web, which has the potential to further enhance knowledge discovery for cultural heritage data.

6. References

Kalina Bontcheva, Valentin Tablan, Diana Maynard, and Hamish Cunningham. 2004. Evolving GATE to Meet New Challenges in Language Engineering. *Natural Language Engineering*.

H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. 2002. GATE: A framework and graphical development environment for robust NLP tools and applications. In *Proc. of the 40th Anniversary Meeting of the ACL*. http://gate.ac.uk.

Martin Doerr. 2003. The CIDOC Conceptual Reference Module: An Ontological Approach to Semantic Interoperability of Metadata. *AI Mag.*, 24(3):75–92.

Michel Généreux. 2007. Cultural Heritage Digital Resources: From Extraction to Querying. In *Proceedings of the Workshop on Language Technology for Cultural Heritage Data (LaTeCH 2007)*, pages 41–48, Prague, Czech Republic, June. ACL.

Markus Krötzsch, Denny Vrandečić, and Max Völkel. 2006. Semantic MediaWiki. In Isabel Cruz, Stefan Decker, Dean Allemang, Chris Preist, Daniel Schwabe, Peter Mika, Mike Uschold, and Lora Aroyo, editors, *The Semantic Web – ISWC 2006*, volume 4273 of *LNCS*, pages 935–942. Springer.

Bo Leuf and Ward Cunningham. 2001. *The Wiki Way, Quick Collaboration on the Web*. Addison-Wesley.

Peter Lyman and Hal R. Varian. 2003. How Much Information?

I. Mani. 2001. *Automatic Summarization*. John Benjamins B.V.

Efthimios C. Mavrikas, Nicolas Nicoloyannis, and Evangelia Kavakli. 2004. Cultural Heritage Information on the Semantic Web. In Enrico Motta, Nigel Shadbolt, Arthur Stutt, and Nicholas Gibbins, editors, *EKAW*, volume 3257 of *Lecture Notes in Computer Science*, pages 477–478. Springer.

D. Maynard, W. Peters, and Y. Li. 2006. Metrics for Evaluation of Ontology-based Information Extraction. In *Proceedings of the 4th International Workshop on Evaluation of Ontologies on the Web (EON 2006)*, Edingburgh, UK, May.

Praharshana Perera and René Witte. 2005. A Self-Learning Context-Aware Lemmatizer for German. In *Proc. of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP 2005)*, pages 636–643, Vancouver, BC, Canada, October 6–8.

Jeffrey A. Rydberg-Cox. 2002. Cultural Heritage Language Technologies: Building an Infrastructure for Collaborative Digital Libraries in the Humanities. *Ariadne*, 34, December.

Jeffrey A. Rydberg-Cox. 2005. The Cultural Heritage Language Technologies Consortium. *D-Lib Magazine*, 11(5), May.

Sebastian Schaffert. 2006. IkeWiki: A Semantic Wiki for Collaborative Knowledge Management. In *WETICE*, pages 388–396.

Patrick Sinclair, Paul Lewis, Kirk Martinez, Matthew Addis, Adrian Pillinger, and Daniel Prideaux. 2005. eCHASE: Exploiting Cultural Heritage using the Semantic Web. In *4th International Semantic Web Conference (ISWC 2005)*, Galway, Ireland, November 6–10.

L. Sweeney. 2001. Information Explosion. In L. Zayatz, P. Doyle, J Theeuwes, and J. Lane, editors, *Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies*. Urban Institute, Washington, DC.

Universität Tübingen – Zentrum für Datenverarbeitung, 2008. *TUSTEP: Handbuch und Referenz*. Version 2008.

René Witte, Petra Gerlach, Markus Joachim, Thomas Kappler, Ralf Krestel, and Praharshana Perera. 2005. Engineering a Semantic Desktop for Building Historians and Architects. In *Proc. of the Semantic Desktop Workshop at the ISWC 2005*, volume 175 of *CEUR*, pages 138–152, Galway, Ireland, November 6.