The Lemlat 3.0 Package for Morphological Analysis of Latin

Marco Passarotti
CIRCSE Research Centre
Università Cattolica del Sacro Cuore
marco.passarotti@unicatt.it

Marco Budassi
Università degli Studi di Pavia
marcobudassi@hotmail.it

Eleonora Litta
CIRCSE Research Centre
Università Cattolica del Sacro Cuore
eleonoramaria.litta@unicatt.it

Paolo Ruffolo
CIRCSE Research Centre
Università Cattolica del Sacro Cuore
paolo.ruffolo@posteo.net

Abstract

This paper introduces the main components of the downloadable package of the 3.0 version of the morphological analyser for Latin Lemlat. The processes of word form analysis and treatment of spelling variation performed by the tool are detailed, as well as the different output formats and the connection of the results with a recently built resource for derivational morphology of Latin. A light evaluation of the tool's lexical coverage against a diachronic vocabulary of the entire Latin world is also provided.

1 Introduction

A sector of the research area dealing with linguistic resources and Natural Language Processing (NLP) tools that has seen a large growth across the last decade is the one dedicated to building, sharing and exploiting linguistic resources and NLP tools for ancient languages. This has particularly concerned Latin and Ancient Greek as essential means for accessing and understanding the so-called Classical tradition.

Although Latin was among the first languages to be automatically processed with computers (thanks to the pioneering work done by the Italian Jesuit Roberto Busa since the late '40s), throughout history, computational linguistics has mainly focused on living languages. However, the start, in 2006, of the first two syntactically annotated corpora (*treebanks*) for Latin[1] gave rise to a kind of renaissance for linguistic resources and NLP tools for ancient languages.

Several textual and lexical resources, as well as NLP tools, are currently available for Latin. Given that out-of-context lemmatisation and morphological analysis of word forms are generally considered basic layers of linguistic analysis - in some way, feeding the subsequent ones - different morphological analysers were developed for Latin across the years. These are: *Words* (http://archives.nd.edu/words.html), Lemlat (www.lemlat3.eu), *Morpheus* (https://github.com/tmallon/morpheus), reimplemented in 2013 as *Parsley* (https://github.com/goldibex/parsley-core), the PROIEL Latin morphology system (https://github.com/mlj/proiel-webapp/tree/master/lib/morphology) and *LatMor* (http://cistern.cis.lmu.de). *Morpheus*, *Parsley* and *LatMor* are all capable of analysing word forms into their morphological representations including vowel quantity.

Although Lemlat has proved to be the best performing morphological analyser for Latin together with LatMor[2] and the one provided with the largest lexical basis (in terms of both selection of the lexicographic sources and processing of attested graphical variants), its impact on the research community has been narrowed for years by its limited accessibility. Only recently, the tool was made freely available, in its 3.0 version,

[1] These were the *Index Thomisticus* Treebank, based on texts of Thomas Aquinas (IT-TB; Passarotti, 2009) and the Latin Dependency Treebank (LDT; Bamman and Crane, 2006), on texts of the Classical era. Later on, a third Latin treebank was made available in the PROIEL corpus (Haug and Jøndal, 2008), which includes the oldest extant versions of the New Testament in Indo-European languages and Latin texts from both the Classical and Late eras. All the three Latin treebanks are dependency-based.

[2] For the results of a comparison between the morphological analysers for Latin see Springmann et al. (2016, p. 389).

thanks to the collaboration between the CIRCSE Research Centre in Milan and the Istituto di Linguistica Computazionale of CNR in Pisa (ILC-CNR). This paper introduces the main components of the downloadable package of Lemlat 3.0.

2 Lemlat

First released as a morphological lemmatiser at the end of the 1980s at ILC-CNR (v 1.0; Bozzi and Cappelli, 1990; Marinone, 1990) and there enhanced with morphological features between 2002 and 2005 (v 2.0; Passarotti, 2004), Lemlat relies on a lexical basis resulting from the collation of three Latin dictionaries (GGG: Georges and Georges, 1913-1918; Glare, 1982; Gradenwitz, 1904) for a total of 40,014 lexical entries and 43,432 lemmas, as more than one lemma can be included in one lexical entry.

Lemlat was originally built for performing the automatic lemmatisation of the texts in the collection of Latin grammarians by Heinrich Keil (1855-1880). Since the first version of Lemlat, one desideratum was pursuing a philological approach to lexical data, which was addressed by connecting the lexical basis of the tool with widely recognised reference dictionaries for Latin, whose contents where collated and recorded carefully. In the light of such an approach, Georges and Georges (1913-1918) was chosen instead of Forcellini's *Lexicon Totius Latinitatis* (1940). Indeed, although Forcellini is the Latin dictionary that comprises the highest number of lemmas, Lomanto (1980) demonstrates that Georges and Georges shows both a higher lexical richness and a better quality of the entries.

Given that Forcellini is the Latin dictionary providing the largest Onomasticon, in the context of the development of the 3.0 version of Lemlat, its lexical basis was further enlarged by adding semi-automatically most of the Onomasticon (26,415 lemmas out of 28,178) provided by the 5th edition of Forcellini (Budassi and Passarotti, 2016).[3]

2.1 Word Form Analysis

Given an input word form that is recognised by Lemlat, the tool produces in output the corresponding lemma(s) and a number of tags conveying (a) the inflectional paradigm of the lemma(s) (e.g. first declension noun) and (b) the morphological features of the input word form (e.g. singular nominative), as well as the identification number (N_ID) of the lemma(s) in the lexical basis of Lemlat.[4] No contextual disambiguation is performed.

For instance, receiving in input the word form *acrimoniae* 'pungency', Lemlat outputs the corresponding lemma (*acrimonia*, N_ID: a0417), the tags for its inflectional paradigm (N1: first declension noun) and those for the morphological features (feminine singular genitive and dative; feminine plural nominative and vocative).

Lemlat is based on a database that includes several tables recording the different formative elements (segments) of word forms. The most important table is the "lexical look-up table", whose basic component is the so-called LES ("LExical Segment"). The LES is defined as the invariable part of the inflected form (e.g. *acrimoni* for *acrimoni-ae*). In other words, the LES is the sequence (or one of the sequences) of characters that remains the same in the inflectional paradigm of a lemma (hence, the LES does not necessarily correspond either to the word stem or to the root).

Lemlat includes a LES archive, in which LES are assigned an N_ID and a number of inflectional features among which are a tag for the gender of the lemma (for nouns only) and a code (called CODLES) for its inflectional category. According to the CODLES, the LES is compatible with the endings (called SF, "Final Segment") of its inflectional paradigm, which are collected in a separate table in the database of Lemlat. For example, the CODLES for the LES *acrimoni* is N1 (first declension nouns) and its gender is F (feminine). The word form *acrimoniae* is thus analysed as belonging to the LES *acrimoni* because the segment *–ae* is recognised as an ending compatible with a LES with CODLES N1.

Segmenting a word form into the structure LES + SF (*acrimoni-ae*) is just one of the possible options provided by Lemlat. Indeed, on one side, word forms can be analysed without any segmentation like in the case of uninflected words (e.g. *semper* 'always'). On the other side, more complex segmentation structures can be at work, including several different segments. This is the case, for instance, of the word form *castigatissimusque* 'and the most punished' (literal translation), which is segmented by Lemlat into *castig-at-issim-us-que*, where *castig* is a LES (for the

[3] For details about credits of the different versions of Lemlat see http://www.lemlat3.eu/about/credits/.

[4] The tagset of Lemlat is compliant with EAGLES (http://www.ilc.cnr.it/EAGLES/browse.html).

verb *castigo*, 'to punish'), *at* and *issim* are two SM ("Middle Segments") representing the infix respectively for perfect participle (*–at–*) and superlative degree (*–issim–*), *us* is a SF (singular masculine nominative) and the enclitics *que* is a SPF ("Post Final Segment"). Overall, the segmentation of *castigatissimusque* has the structure LES+SM+SM+SF+SPF. Each kind of segment is stored in a specific table in the database of Lemlat.

Finally, if the analysed word is morphologically derived or if it is the basis of one or more morphologically derived word(s), its derivation cluster is provided (see Section 3). For instance, the input word form *amabilem* is analysed by Lemlat as singular masculine/feminine accusative of the adjective *amabilis* 'lovable'. This lemma is part of a derivation cluster: *amabilis* is derived from the verb *amo* 'to love' and it is the basis for two derived words, namely the noun *amabilitas* 'loveliness' and the adjective *inamabilis* 'unlovely'. Relations are connected with the specific word formation rule they instantiate. For instance, *amabilis* is stored as a second class deverbal adjective with suffix *-a-bil-is*.

2.2 Spelling Variation

Textual material written in Latin is spread across a diachronic span wider than two millennia. Furthermore, Latin texts are distributed all over Europe and cover various kinds of genres.

Such a situation makes Latin a language featuring a large amount of spelling variations, due to several reasons, among which are the influence of local dialects, the writing conventions (which are subject to changes across time and place), as well as the style and the level of education of the authors.

Since its first version, Lemlat was designed to address the question of spelling variation. As mentioned above, one distinctive feature of Lemlat is its strict connection with the reference lexicographic sources. Such a connection motivates also the treatment of graphical variants in Lemlat. Indeed, the lexical look-up table featuring the list of LES includes also those that are used by the tool for processing spelling variations.

In the lexical look-up table, each lexical entry in dictionaries corresponds to as many lines as are the different LES required by Lemlat to process its full inflectional paradigm, spelling variations included. All lines belonging to the same lexical entry are assigned the same N_ID.

For instance, Glare (1982) records the Faliscan spelling variation *haba* for the first declension noun *faba* 'horse-bean'. In the lexical look-up table of Lemlat, this results into two separate lines with the same N_ID. One line reports the LES *fab* (for *faba*). The other has the LES *hab* (for *haba*). Both the LES are assigned a code for gender (feminine) and the same CODLES (N1). A specific field in the table is reserved for selecting the LES to use for building the lemma in the case of lexical entries featuring more than one LES. For *faba*, the LES *fab* is the one used, as the lemma in Glare (1982) is *faba* (and not *haba*).

Along with recording different LES for the same lexical entry, there is also another strategy used by Lemlat to process spelling variations. In the case of variations that apply to sets of words sharing some graphical properties, a field in the look-up table records a code that permits to alter the LES while processing the data. For instance, a large number of words including the prefix *trans–* have forms featuring graphical variations of *trans–*, namely *tra–* and *tras–* (*trans–* is the citation form of the prefix reported by Glare, 1982). In Lemlat, there are 35 lexical entries showing this spelling variation. All their LES are assigned a specific code (`t02`) in the lexical look-up table, which permits the alternation between the graphical forms of *trans–*. An example is the lemma *transfero* 'to transport'. Although its LES is *transfer*, the presence of `t02` makes Lemlat able to process also the graphical variants *trafero* and *trasfero*.

Such an approach to spelling variation is at the same time a pro and a con. On one side, it makes Lemlat lexicologically motivated, as only those variations that are recorded in the reference dictionaries are processed by the tool. On the other, it makes Lemlat rigid, as it allows to process only those graphical variants that are explicitly recorded in the lexical look-up table.

3 Derivational Morphology

The analysis of inflectional morphology provided by Lemlat has been recently enhanced with information on derivational morphology. Built within the context of an ongoing project funded by the EU Horizon 2020 Research and Innovation Programme (under the Marie Skłodowska-Curie Individual Fellowship), *Word Formation Latin* (WFL) is a derivational morphology resource for Latin that can also work as an NLP tool, thanks to its strict relation with Lemlat (Litta et al., 2016).

In WFL the lemmas of Lemlat are connected by Word Formation Rules (WFRs). In WFL,

there are two main types of WFRs: (a) derivation and (b) compounding. Derivation rules are further organised into two subcategories: (a) affixal, in its turn split into prefixal and suffixal, and (b) conversion, a derivation process that changes the Part of Speech (PoS) of the input word without affixation.

WFL is built in two steps. First, WFRs are detected. Then, they are applied to lexical data. Affixal WFRs are found both according to previous literature on Latin derivational morphology (e.g. Fruyt, 2011; Jenks, 1911) and in a semi-automatic manner. The latter is performed by extracting from the list of lemmas of Lemlat the most frequent sequences of characters occurring on the left (prefixes) and on the right (suffixes) sides of lemmas. The PoS for WFRs input and output lemmas as well as their inflectional category are manually assigned. Further affixal WFRs are found by comparison with data. So far, 244 affixal WFRs have been detected: 94 prefixal and 150 suffixal.

Compounding and conversion WFRs are manually listed by considering all the possible combinations of main PoS (verbs, nouns, adjectives), regardless of their actual instantiations in the lexical basis. For instance, there are four possible types of conversion WFRs involving verbs: V-To-N (*claudo* → *clausa*; 'to close' → 'cell'), V-To-A (*eligo* → *elegans*; 'to pick out' → 'accustomed to select, tasteful'), N-To-V (*magister* → *magistro*; 'master' → 'to rule'), A-To-V (*celer* → *celero*; 'quick' → 'to quicken'). Each compounding and conversion WFR type is further filtered by the inflectional category of both input and output. For instance, A1-To-V1 is the conversion WFR that derives first conjugation verbs (V1) from first class adjectives (A1).

Applying WFRs to lexical data requires that each morphologically derived lemma is assigned a WFR and is paired with its base lemma. All those lemmas that share a common (not derived) ancestor belong to the same "morphological family". For instance, nouns *amator* 'lover' and *amor* 'love', and adjective *amabilis* all belong to the morphological family whose ancestor is the verb *amo*.

WFRs are modelled as one-to-may relations between lemmas. These relations are implemented by a table in the database where they are enhanced with their attributes (type, category, affix). So far, 299 WFRs have been applied, which build 5,348 morphological families and 23,340 input-output relations.

The contents of WFL can be accessed via a web application (available at `http://wfl.marginalia.it`; Culy et al., forthcoming), which features a positive balance between potential of data extraction and simplicity, dynamism and interactivity.

The web application represents the information stored in the tables of the database as a graph. In this graph, a node is a lemma, and an edge is the WFR used to derive the output lemma from the input one (or two, in the case of compounds), along with any affix used. The graph is represented as a collection of nodes and edges, and the set of morphological families is simply the set of connected subgraphs.

Four distinct perspectives to query WFL are available from the web application:

- by WFR – the primary interest is the WFR itself. This view enables research questions on the behaviour of a specific WFR. For example, it is possible to view and download the list of all verbs derived from a noun through a conversive derivation process (e.g. *radix* 'root' → *radicor* 'to grow roots');
- by affix – it acts similarly as above, but works more specifically on affixal behaviour. For example, this perspective enables to retrieve all masculine nouns featuring the suffix *–tor* and to verify how many of them correspond to a female equivalent ending in *–trix*;
- by PoS – the primary interest is in the PoS of input and output lemmas. This view is useful for studies on macro-categories of morphological transformation, like nominalisation and verbalisation;
- by lemma – it focuses on both derived and non-derived lemmas. It supports studies on the productivity of one specific morphological family or a set of morphological families.

The results of these browsing options are of three types:

- lists of lemmas matching a query;
- derivational clusters. This type of graph represents the derivational chain for a specific lemma, which includes all the lemmas derived from the lemma selected, as well as all those the lemma is derived from;
- summaries of the application of given WFRs to different PoS and the resulting lemmas.

4 Data Processing

The database of Lemlat 3.0 is available at `https://github.com/CIRCSE/LEMLAT3`, where also a Command Line Interface (CLI) implementation of the tool for Linux, OSX and Windows can be downloaded.

In particular, two versions are made available: (a) a client version, which requires a working MySQL server (`www.mysql.com`) containing the provided database and (b) a stand-alone version, which uses an embedded version of the database. Both the client and the stand-alone versions use the same CLI interface and can be run either in interactive or in batch mode. The interactive mode provides the user with the possibility of running Lemlat on one input word form at a time, selecting the lexical basis to use for analysis (GGG only; Onomasticon only; GGG + Onomasticon). The batch mode enables to process a bunch of word forms by entering either a file featuring the list of word forms to analyse or a full text. Three different formats are available for the output: plain text, XML and Comma-Separated Values file (CSV).

The output in the form of a plain text file reports exactly the same information displayed in the interactive mode. For each analysis of a processed word form, it provides (a) the segmentation of the word form into its formative elements, (b) its morphological features and (c) its lemma(s) with the corresponding PoS.

The XML output includes the complete analysis for each processed word form organised into explicitly named elements and attributes, and can be validated against the provided DTD.

The CSV file provides just basic lemmatisation, without morphological features. Each analysed word form is assigned its lemma and PoS (with gender for nouns). If a word form is assigned more than one lemma, these are provided on separate lines.

The list of the not analysed word forms is provided in a separate plain text file with the same name of the input file and the extension ".unk".

Both in the plain text and in the XML output files, each lemma is assigned a feature coming from WFL that informs if it is morphologically simple, i.e. not derived, or complex, i.e. resulting from the application of a WFR. Each morphologically complex lemma is matched with (a) the lemma which it derives from (two lemmas, in case of compounding), (b) the type of WFR involved, (c) the input and output PoS of the WFR and (d) the affix (prefix or suffix) if present in the derivation.

5 Evaluation

In order to evaluate the lexical coverage of Lemlat 3.0 on real texts, the full list of word forms extracted from *Thesaurus Formarum Totius Latinatis* have been lemmatised (TFTL; Tombeur, 1998). Widely recognised as the reference tool *par excellence* with regard to studies of Latin lexicon, TFTL is a large diachronic database collecting the vocabulary of the entire Latin world ranging from the ancient Latin literature to Neo-Latin works. Word forms are assigned their number of occurrences in the texts of the different eras.

400,886 out of the total 554,826 different forms of TFTL were analysed by Lemlat, for a total of 489,441 analyses, returning a coverage percentage of 72.254%.[5]

However, among the 153,447 forms not analysed by Lemlat, there are prominently sequences of letters (e.g. *aaa*), numbers (e.g. *CCC*), and extremely rare word forms (e.g. *aaliza*, 1 occurrence in TFTL). Results are more reliably evaluated by looking at the number of textual occurrences of the words analysed by Lemlat, compared to the total number of occurrences in TFTL. The sum of absolute frequencies of all word forms in TFTL is 62,922,781. The sum of absolute frequencies of those analysed by Lemlat is 61,881,702. Thus, Lemlat can analyse 98.345% of the occurrences in the TFTL texts.

6 Discussion and Future Work

Lemlat processes word forms by segmentation, finding compatible connections of formative elements, which are recorded in the tables of a database. Such rigid approach to morphological processing looks quite out-of-date if compared with the most widespread techniques currently used to perform automatic morphological analysis. In particular, several finite-state packages are today available[6], which feature both large lexical coverage and high flexibility, especially when they are connected to data driven techniques for

[5] The number of analyses is higher than the number of analysed forms, because a single word form can be assigned more than one lemma.

[6] See, for instance, the Helsinki Finite-State Transducer (Lindén et al. 2009), the Stuttgart Finite-State Transducer Tools (Schmid, 2005), the OpenFST library for weighted finite-state transducers (Allauzen et al., 2007) and the Foma finite-state library (Hulden, 2009).

statistical processing with weighted transducers (Pirinen, 2015) and for inflectional class inference (Dreyer et al. 2008). Moreover, the finite-state approach makes it possible to use the same code to handle both analysis and generation.

The segmentation-based approach pursued by Lemlat is due to two main reasons.

First, despite its recent availability, Lemlat is an old tool, being conceived in the early 1980s (Marinone, 1983), a time when the finite-state turn in computational morphology was still in its infancy.[7] Actually, the process of word form analysis performed by Lemlat is quite similar to that of finite-state morphology, as they share the basic assumption that natural language words are formed of concatenated "pieces" which are compatible to each other. The formative elements recognised by Lemlat in word forms can be seen as states and their relations as directed arcs controlled by rules. Basically, Lemlat formalises the lexicon as a finite-state transducer and analyses words as sequences of compatible segments. However, two (important) differences remain: (a) Lemlat is not meant to generate morphologically well-formed words; (b) Lemlat does not include rules for constraining lexical/surface correspondences, as it treats phonological alternations just like regular sequences of explicitly recorded segments.

Second, Lemlat was primarily built to address the needs of philologists, who are more interested in processing data according to reference lexicographic sources than in having a flexible and computationally efficient tool able to perform (also) lexical generation. Indeed, one distinctive feature of Lemlat is the quality of its lexical basis, which enables the tool to process all the graphical inflectional variants attested for the lexical entries in the reference dictionaries. Although such lexical basis allows for quite a broad textual coverage (see Section 5), several lemmas belonging to different phases of Medieval Latin are still missing. For this reason and to keep supporting Lemlat with quality lexicographic sources, we plan to expand its lexical basis with all the entries of Du Cange's (1883-1887) *Glossarium Mediae et Infimae Latinitatis*.

Furthermore, while still keeping the original philological approach Lemlat is built upon, in the near future we plan to enhance it with a statistical guesser, which might process those word forms that are not recognised by Lemlat.

As mentioned, Lemlat is an out-of-context morphological analyser. The structure of the running text is lost and no contextual disambiguation of multiple analyses is performed. The current availability of annotated corpora for Latin, like the three dependency treebanks (see Section 1), made it possible to train a number of probabilistic PoS taggers and lemmatisers. For instance, two parameter files for Latin are available for TreeTagger (Schmid, 1999). One file is based on the IT-TB, while the other is built upon data joint from the three Latin treebanks. Recently, pre-trained tagging models for Latin (based on the versions of the three Latin treebanks available in Universal Dependencies 1.3; `http://universaldependencies.org/`) were provided by RDRPOSTagger version 1.2.2 (Nguyen et al., 2014) with those for other 40 languages. Tagging accuracy ranges from 90.39 for the LDT to 98.24 for the IT-TB, PROIEL standing somewhere in the middle (95.78).[8]

The large lexical coverage and the high quality of analysis provided by Lemlat can be helpful for improving the performances of PoS taggers, by enhancing the tools with a morphological lexicon that provides all the possible pairs of lemma and morphological features for each word form. For instance, such a lexicon is used in popular PoS taggers like TreeTagger and MorphoDiTa (Straková et al. 2014). Although Lemlat was conceived to analyse input words and not to generate morphologically well-formed words, the result of the analysis performed on TFTL (see Section 5) is just a morphological lexicon for Latin providing large coverage of attested word forms.

Finally, a web application of Lemlat will be made available at `www.lemlat3.eu`, enabling users to process either single words or short texts. The web application of Lemlat will be linked and merged with that of WFL, thus providing one common environment for the online processing and visualisation of both inflectional and derivational morphology of Latin.

Acknowledgements

We thank three anonymous reviewers for their valuable comments and suggestions, which helped to improve the quality of the paper.

[7] The publication that mostly contributed to start such a turn is the 1983 dissertation by Kimmo Koskenniemi on a formalism to describe phonological alternations in finite-state terms, which he called "Two-Level Morphology" (Koskenniemi, 1983). An historical overview on finite-state morphology is given by Karttunen and Beesley (2005).

[8] A survey of the accuracy of several taggers based on a corpus of Medieval Church Latin is provided by Eger et al. (2015).

References

Cyril Allauzen, Michael Riley, Johan Schalkwyk, Wojciech Skut, and Mehryar Mohri. 2007. OpenFst: A General and Efficient Weighted Finite-State Transducer Library. In *Proceedings of the Twelfth International Conference on Implementation and Application of Automata, (CIAA 2007)*, volume 4783 of *Lecture Notes in Computer Science*, pages 11–23, Springer, Prague, Czech Republic.

David Bamman and Gregory Crane. 2006. The Design and Use of a Latin Dependency Treebank. In *TLT 2006: Proceedings of the Fifth International Treebanks and Linguistic Theories Conference*, pages 67–78.

Andrea Bozzi and Giuseppe Cappelli. 1990. A Project for Latin Lexicography: 2. A Latin Morphological Analyser. *Computer and the Humanities*, 24:421–426.

Marco Budassi and Marco Passarotti. 2016. Nomen Omen. Enhancing the Latin Morphological Analyser Lemlat with an Onomasticon. In *Proceedings of the 10th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH 2016)*, pages 90–94, The Association for Computational Linguistics, Berlin.

Chris Culy, Eleonora Litta, and Marco Passarotti. Forthcoming. Visual Exploration of Latin Derivational Morphology. In *Proceedings of the 30th International Conference of the Florida Artificial Intelligence Research Society (FLAIRS-30)*.

Markus Dreyer, Jason R. Smith, and Jason Eisner. 2008. Latent-variable modeling of string transductions with finite-state methods. In *Proceedings of the conference on empirical methods in natural language processing*, pages 1080–1089, Association for Computational Linguistics.

Charles du Fresne Du Cange et al. 1883-1887. *Glossarium Mediae et Infimae Latinitatis*. L. Favre, Niort.

Steffen Eger, Tim vor der Brück, and Alexander Mehler. 2015. Lexicon-assisted tagging and lemmatization in Latin: A comparison of six taggers and two lemmatization methods. In *Proceedings of the 9th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH 2015)*, pages 105–113.

Egidio Forcellini. 1940. *Lexicon Totius Latinitatis / ad Aeg. Forcellini lucubratum, dein a Jos. Furlanetto emendatum et auctum; nunc demum Fr. Corradini et Jos. Perin curantibus emendatius et auctius meloremque in formam redactum adjecto altera quasi parte Onomastico totius latinitatis opera et studio ejusdem Jos. Perin*. Typis Seminarii, Padova.

Michèle Fruyt. 2011. Word Formation in Classical Latin. In James Clackson (ed.), *A companion to the Latin language*. Vol. 132, pages 157–175, John Wiley & Sons, Chichester.

Karl E. Georges and Heinrich Georges. 1913-1918. *Ausführliches Lateinisch-Deutsches Handwörterbuch*. Hahn, Hannover.

Peter G.W. Glare. 1982. *Oxford Latin Dictionary*. Oxford University Press, Oxford.

Otto Gradenwitz. 1904. *Laterculi Vocum Latinarum*. Hirzel, Leipzig.

Dag T.T. Haug and Marius L. Jøhndal. 2008. Creating a Parallel Treebank of the Old Indo-European Bible Translations. In *Proceedings of the Second Workshop on Language Technology for Cultural Heritage Data (LaTeCH 2008)*, pages 27–34.

Mans Hulden. 2009. Foma: a finite-state compiler and library. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 29–32.

Paul Jenks. 1911. *A Manual of Latin Word Formation for Secondary Schools*. DC Heath & Company, Boston, New York etc.

Lauri Karttunen and Kenneth R. Beesley. 2005. Twenty-five years of finite-state morphology. In *Inquiries Into Words, a Festschrift for Kimmo Koskenniemi on his 60th Birthday*, pages 71–83.

Heinrich Keil. 1855-1880. *Grammatici Latini*. Teubner, Leipzig.

Kimmo Koskenniemi. 1983. *Two-level morphology: A general computational model for word-form recognition and production*. Publication 11, University of Helsinki, Department of General Linguistics, Helsinki.

Kristen Lindén, Miikka Silfverberg, and Tommi Pirinen. 2009. HFST Tools for Morphology - An Efficient Open-Source Package for Construction of Morphological Analyzers. In Cerstin Mahlow and Michael Piotrowski (eds.), *Proceedings of the Workshop on Systems and Frameworks for Computational Morphology*, volume 41 of *Lecture Notes in Computer Science*, pages 28–47, Springer, Zurich, Switzerland.

Eleonora Litta, Marco Passarotti, and Chris Culy. 2016. Formatio formosa est. Building a Word Formation Lexicon for Latin. In Anna Corazza, Simonetta Montemagni, and Giovanni Semeraro (eds.), *Proceedings of the Third Italian Conference on Computational Linguistics (CLiC–it 2016). 5-6 December 2016, Napoli, Italy*, pages 185–189, aAccademia university press, Collana dell'Associazione Italiana di Linguistica Computazionale, vol. 2.

Valeria Lomanto. 1980. Lessici latini e lessicografia automatica. *Memorie dell'Accademia delle Scienze di Torino*, 5.4.2:111–269.

Nino Marinone. 1983. A project for a Latin lexical data base. *Linguistica Computazionale*, 3:175–187.

Nino Marinone. 1990. A Project for Latin Lexicography: 1. Automatic Lemmatization and Word-List. *Computer and the Humanities*, 24:417–420.

Dat Quoc Nguyen, Dai Quoc Nguyen, Dang Duc Pham, and Son Bao Pham. 2014. RDRPOSTagger: A Ripple Down Rules-based Part-Of-Speech Tagger. In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 17–20.

Marco Passarotti. 2004. Development and perspectives of the Latin morphological analyser LEMLAT. *Linguistica Computazionale*, XX-XXI: 397–414.

Marco Passarotti. 2009. Theory and Practice of Corpus Annotation in the *Index Thomisticus* Treebank. *Lexis*, 27:5–23.

Tommi A. Pirinen. 2015. Using weighted finite state morphology with VISL CG-3 – Some experiments with free open source Finnish resources. In *Proceedings of the Workshop on "Constraint Grammar-methods, tools and applications" at NODALIDA 2015, May 11-13, 2015*, pages 29–33, Institute of the Lithuanian Language, Vilnius, Lithuania. No. 113. Linköping University Electronic Press.

Helmut Schmid. 1999. Improvements in part-of-speech tagging with an application to German. *Natural language processing using very large corpora*. pages 13–25, Springer.

Helmut Schmid. 2005. A Programming Language for Finite State Transducers. In *Proceedings of the 5th International Workshop on Finite State Methods in Natural Language Processing (FSMNLP 2005)*, Helsinki, Finland.

Uwe Springmann, Helmut Schmid, and Dietmar Najock. 2016. LatMor: A Latin Finite-State Morphology Encoding Vowel Quantity. In Giuseppe Celano and Gregory Crane (eds.), *Treebanking and Ancient Languages: Current and Prospective Research* (Topical Issue), *Open Linguistics* vol. 2, pages 386–392.

Jana Straková, Milan Straka, and Jan Hajič. 2014. Open-Source Tools for Morphology, Lemmatization, POS Tagging and Named Entity Recognition. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 13–18.

Paul Tombeur (ed.). 1998. *Thesaurus formarum totius latinitatis a Plauto usque ad saeculum XXum*. Brepols, Turnhout.