

Latin Dictionary Tools in Internet

Dainis Zeps

Institute of Mathematics and Computer Science,
University of Latvia
Riga, Latvia
dainize@mii.lu.lv

Abstract

Latin Dictionary Tools Page <http://susurs.mii.lu.lv/dainize/lingua.htm> is produced for teachers and students of Latin to have an access to a large Latin morphological dictionary data base in Internet. Dictionary equipped with morphology recognizing and generating power becomes a quite new tool both in its inquiring and testability features.

1. Introduction

Latin Dictionary Tools Page (LDTP) is an Internet development¹ that is produced for teachers and students of Latin to have an access to a large Latin dictionary that actually is an integrated comprehensive database providing multifunctional access with aim to serve for users eventually inexperienced in computer science. Further, our aims are

- 1) to investigate possibilities and ways to implement set of tools that may be called *dictionary* in a sense that they do the labor of dictionaries and more;
- 2) to develop Latin morphology maintaining database for morphological form recognizing and generating functionality;

In future, these tools would be aimed, firstly, to become a morphological block of some more general Latin language recognition entity, and, secondly, to give rise to possibility to develop morphological tools for, say, Greek in Internet.

Among functions of LDTP are

- 1) recognition of Latin morphological forms enabling to recognize most of language forms in classic and ecclesiastic Latin (more than 45 thousand primitive stems),
- 2) searching full dictionary and four small (educational) subdictionaries with some most useful built in queries [scheduled for teachers],
- 3) inflection's table generator allowing to inflect words of all inflective classes supported with possibility to change grade of comparison

of adjective and genus of numeral and grammatical category of verb.

LDTP is replenished with Latin Words quiz with five levels of hardness (five subdictionaries) and Latin multiplication table quiz facilitating to make fun along with learning. Thus, among aims of LDTP is to investigate possible implementing of new necessary, untraditional tools for language teachers.

Author used William Whitaker's² Latin morphological dictionary not only in its lexical part but, at least initially, in its morphological part too.

2. Insight in general

Since computers are widely in use, a dictionary as a tool to provide lookup of an uncertain vocabular entity is not changed as dramatically as it would be expected if compared with general development of information technologies, i. e. electronic dictionary that do more or less the same thing what it did when it was in a printed and bound book form where it was consulted by browsing pages of it, are in wide use and new paradigmata in the field are not as much welcomed as it would be expected from investigators. How to overcome it is one of the problems touched in this article.

The history of development of electronic lookup tools reveals natural way of development of the idea. Firstly printed book sample is mirrored in an informational environment by one-one matching, i. e. it is simply made the same book with the only

¹ <http://susurs.mii.lu.lv/dainize/lingua.htm>

² <http://www.erols.com/whitaker/words.htm>

difference that it is in the electronic format. With little improvements this type of dictionary we use mostly nowadays. Integrating this type of electronic dictionary in a database gives improvement in swiftness of data retrieval, maybe attaining by the way some new relational features provided by database data organization, but generally the paradigm has remained the same. Next step would be to call dictionary a set of some lookup functions which are provided in some more general equipment either of translating or learning or both integrated together nature (Hanks, 2003).

2.1. Morphological dictionaries

Next step in the principal development of dictionary tool is the morphological form (or inflections (Bickel, 2001; Trost, 2003; Hausser, 1999)) recognizing dictionary, in a narrower application called also according functionality part-of-speech tagger (if it returns tagged initial text) or simply speller (if returns unrecognized morphological forms) [see references and links about this in (Hammarstrom, 2002; Voutilainen, 2003)]. Morphological dictionaries, as they simpler may be called, are essentially important in languages which have a large morphological form distinction, e. g. in ancient languages, Greek, Latin, e. c., although they are principally actual in all languages without exception (even more in agglutinative languages, e. g. Hebrew, with its simply inflectional part as its subset and its clitical part principally agglutinative in nature), even in English where relatively small form distinction is present.

It is just to say that morphological form recognizing dictionary give a completely new paradigm what is not present in conventional dictionaries: morphological dictionaries do part of the work what was supposed to be of intellectual nature, i. e. the recognition of the form, providing thus new principal function, for any word from the text automatic return of the set of lemmata and meanings from dictionary. Because of this, it is pity that morphological dictionaries are not very popular and required by philologists who are not computer scientists who should be eventual customers and exploiters of the new paradigm.

3. Fundamentals of the development

3.1. Morphological database

Morphological database as its fundamental part has two main tables, stems and inflects, i.e. morphological form always is concatenation of two strings, the stem part and the inflect part. Whenever some morpheme has allomorphs of different stems, morpheme is divided as if in two submorphemes, e. g. present stem submorpheme and perfect stem submorpheme. Consequently applying this method, maximal count of submorphemes necessary to enter to support two-partiality of a morphological form is four, where verbs, for example, require four submorphemes. As a consequence, a new Latin word, by entering it in the database as a new morpheme, is consisting from possibly several submorphemes, where each of them represents a distinct stem. Each part of speech or rather subclass of part of speech has its own representation pattern of its morpheme in submorphemes, e.g. noun as a morpheme is represented in the pattern consisting of two submorphemes, e.g. 'homo' and 'homin', verbs --four, e.g. 'capi' for present stem finite forms, 'cap' for infinitive, 'cep' for perfect stem, 'capt' for supine stem, adjectives are divided in subclasses of four, three and two stems.

Different pattern of morpheme's map in submorphemes is necessary only for some clitic parts' accepting pronominals, e.g. 'quiscumque', where pronominal 'quis' accepts enclitics 'cum' and 'que', which are distinct morphemes by their own.

3.2. Flexion table generator

Morphological forms are generated using a function called 'flexion table generator' FTG. FTG gives some fixed subsets of allomorphs of a morpheme, where allomorphs are varied in a fixed way, i. e. nouns are varied by case and number, and verbs are varied by person and number, i.e. similarly as in traditional flexion tables in book case grammars. Other grammatical categories as inflects in FTG are changeable rather arbitrary, e.g. even grades of comparison and genera for numerals. In order to teach teachers new templates of FTG, nonfixed variable parameters would be useful for teachers, e.g. varied tense and mood (or stem system) for verb, or, say, genus and grade of comparison for adjectives. Further, FTG

might be looked upon and correspondingly used as giving set of allomorphs with varied grammatically categorial inflection as uncertain parameters, which could be determined later, say, in the stage of the syntactical generation or analysis or both.

3.3. Dictionary search function

In *dictionary search function* an attempt is revealed to attribute to the dictionary as traditional lookup tool some new features, which would give dictionary rather database querying functionality than simple word lookup function. Parallel to this *subdictionary* and the dictionary itself as the set of them as an alternative to the dictionary as a closed entity is suggested. For teachers the possibility to find all words of a qualified grammatically categorial content is very useful and instructive. For example, to find all feminine nouns of 4th declination, or all -io verbs in full dictionary or some of its subdictionaries. One who learns such potentialities in a short time would feel them mostly necessary both in teaching and learning of the language. In this context we think this function deserves that it is carefully investigated and developed. Of course, mostly useful search term for philologist would be the stem as a set of all morphemes based on a common stem as a lexical entity with one common meaning. To some reasonable extent it is an easy attainable goal if only morphemes with common meaning stems may be uniquely indexed as belonging to the common class. Otherwise, if we would be interested not only in identical but similar, i.e. synonymous stems, where stems' classes overlap easily arise, we come to a more complex task where some universal solution hardly be expectable.

3.4. Self-testability

A completely new functionality we get from a dictionary as the set of functions as characterized here, if we question it for self-testability, i.e. if we try to evaluate its correctness by the tools which are integrated in itself. It is easy to search our database for its comprehensivity what concerns Latin morphology by taking any Latin grammar book and showing its correspondence with our morphological dictionary. Author did it by checking all morphology from a chosen Latin grammar book and it took time only few hours. In contrary, such quick possibility one

would completely lack either in a book case lexicon or even in electronic traditional book's equivalent. Thus we are as if solving an independent task, i.e. to try a computational morphological system for its comprehensivity and correspondence with the given language morphology, giving a simplest solution to the problem, i.e. integrating in the system itself sufficient amount of functionality which would provide its testability in a sufficiently natural way. As a side effect of the testability of this database, all errors (from most critical evaluators' point of view) and deficiencies and wants are easy discernable. Because of this, some parts are marked pro tempore as 'test version'.

3.5. Programming tools used

The site is produced using active server pages (asp) [with Basic scripts], database tables and queries are built in MS Access. To acquire fast functionality of site and by building dictionaries, throughout linear algorithms are used [Acho 1974].

4. Hale's machine

In the end of 19th century teaching of Latin reached its apogee. One of representatives of this time William Gardner Hale in (Hale 1887) taught that by reading the sentence should be understood as a sequence of augmenting word by word subsentences where before each new coming word correct prediction of all(!) possibilities of syntactical constructions that would follow should be explicitly named. To reach such extraordinary knowledge of Latin that such quite reasonably and precise prediction always could be given, could be possible only if Latin grammar could be taught correspondingly. Is it possible at all? The hardness of this approach is because of the too many possibilities that should arise if text's morphology and syntax are separated from other parts of Latin, i.e. phraseology, idiomatic, lexical peculiarities of the particular author. Could it all manage a single reader? Hale argued that it is possible and taught correspondingly his students of Latin and Greek and asserted that it is the only possible way to read ancient authors.

Today W. G. Hale's approach is significant because computer can model this extreme knowledge of a language in Hale's time required from a void-of-computer human

being. Hale's Latin reading approach justly may be called *Hale's machine* because his precise definition of the functionality of the reader of Latin and his appeal for its comprehensivity and inevitable necessity.

Hale's machine may be mentioned in another sense, i. e. it may be that just contemporary student may hope to reach that level of the knowledge of Latin for what argued Hale in 1887 if Hale's machine would come in use.

5. Resume

More and more new morphosyntactical tools exploiting new paradigmata in widening our understanding of what lexical lookup tool should look like to help us in learning and teaching languages and ancient languages in particular are highly necessary.

At the end author would like to raise a question that may be addressed to mathematical linguists in general what is the highest goal in AI: to produce only reference tools for user or rather learning tools, to provide self-learning machines with new learning abilities to give us intellectual machines as resources of knowledge, or to give us AI tools to learn ourselves. It seems, if questioned directly, mostly both necessities would be accepted, but, objectively, I think, the ratio of both opinions more or less could be found estimating proportion of [teaching in high schools] linguists who teach languages against those who teach mathematical linguistics.

Bibliographical References

- Acho A., Hopcroft J., Ullman J. 1974. *The Design and Analysis of Computer Algorithms*. Addison Wesley.
- Aronoff M. 1994. *Morphology by Itself: Stems and Inflectional Classes*. MIT Press, Cambridge, Mass.
- Bickel B., Nichols J. 2001. Inflectional morphology. In: *Language typology and syntactic description*, ed. Timothy Shopen. Cambridge University Press.
- Hale W.G. 1887. *The Art of Reading Latin: How to Teach It*. Cornell University, Boston, Ginn & Co., pp. 31.
- Hammarstrom H. 2002. *Overview of IT-based tools for learning and training grammar*, pp. 79.
- Hanks P. 2003. Lexicography, In: *The Oxford Handbook of Computational Linguistics*, R.Mitkov, ed., University Press, Oxford, pp.48-69.
- Hausser R. 1999. *Foundations of Computational Linguistics: Man-Machine Communication in Natural Language*, Springer Verlag.
- Khoja Sh., Garside R., Knowles G. *A tagset for the morphosyntactic tagging of Arabic*, Lancaster University.
- Trost H. 2003. Morphology. In: *The Oxford Handbook of Computational Linguistics*, R.Mitkov, ed., University Press, Oxford, pp. 25-47
- Voutilainen A. 2003. Part-of-Speech Tagging. In: *The Oxford Handbook of Computational Linguistics*, R.Mitkov, ed., University Press, Oxford, pp.219-232.