R-based strategies for DH in English Linguistics: a case study

Nicolas Ballier, Paula Lissón

R-based strategies for DH in English Linguistics: a case study

Nicolas Ballier
Université Paris Diderot
UFR Études Anglophones
CLILLAC-ARP (EA 3967)
nicolas.ballier@univ-paris-diderot.fr

Paula Lissón
Université Paris Diderot
UFR Études Anglophones
CLILLAC-ARP (EA 3967)
paula.lisson@etu.univ-paris-diderot.fr

Abstract

This paper is a position statement advocating the implementation of the programming language R in a curriculum of English Linguistics. This is an illustration of a possible strategy for the requirements of Natural Language Processing (NLP) for Digital Humanities (DH) studies in an established curriculum. R plays the role of a Trojan Horse for NLP and statistics, while promoting the acquisition of a programming language. We report an overview of existing practices implemented in an MA and PhD programme at the University of Paris Diderot in the recent years. We emphasize completed aspects of the curriculum and detail existing teaching strategies rather than work in progress but our last section alludes to work still under way, such as getting PhD students to write their own R packages.

We describe our strategy, discuss better practices and teaching concepts, and present experiments in a curriculum. We express the needs of an initially limited NLP environment and provide directions for future DH curricular developments. We detail the challenges in teaching a non-CL audience, showing that some software suites can be integrated to a curriculum, outlining how some specific R packages contribute to the acquisition of NLP-based techniques and favour the awareness of the needs for statistical modelling.

1 Introduction

This paper deals with the development of an R-based culture of DH for students of English Linguistics at the university of Paris Diderot. We describe some aspects of a curriculum (MA and PhD) that aims at taking advantage of the flexibility and the adaptability of this programming language for research in linguistics, both in a quantitative and qualitative approach.

Developing a culture based on the programming language R (R Core Team 2016) for NLP among MA and PhD students doing English Linguistics is no easy task. Broadly speaking, most of the students have little background in mathematics, statistics, or programming, and usually feel reluctant to study any of these disciplines. While most PhD students in English linguistics are former students with a Baccalauréat in Sciences, some MA students pride themselves on having radically opted out of Maths. However, we believe that students should be made aware of the growing use of statistical and NLP methods in linguistics and to be able to interpret and implement these techniques.

We need to show our students how important the DH are for their research, enabling them to see that the use of NLP techniques provides them with a whole range of new possibilities for the treatment and the analysis of their data. In addition, with the growing tendency of corpus linguistics, students often need to work with large corpora or huge databases of images, and standard tutorials and introductory books do not cover these needs (Arnold and Tilton 2015).

Preparing students to work with NLP methods and with command lines also means to ask them to work with particular formats of data they need to get used to (e.g. tabular format, utf8, limited use of special characters unless necessary). However, this facilitates the interoperability of their data, as well as the replicability of their research.

The rest of the paper is organized as following: section 2 describes the context of an MA in English Linguistics and explains why the culture is traditionally limited in NLP and DH in this kind of curriculum. It also details the strategy used to develop an 'R-based culture' among MA

and PhD students, taking advantage of its flexibility and adaptability to the various requirements of linguistic data. Section 3 explains how collaboration with the Maths department has enabled the emergence of an R-based common culture for statistics to be taught to mostly 'mathless' students. Section 4 discusses the various strategies to teach R in recent textbooks of quantitative linguistics (from Baayen, 2008 to Levshina, 2015). Several teaching styles and aims are discussed. Section 5 discusses DH in a wider perspective of Machine Learning (ML) based analyses and show the benefits of R, reporting on the possibility of an interdisciplinary bridge for programs in data science. Section 6 proposes an epistemological interpretation of R as a possible medium for the 3rd revolution of grammatisation, expanding on Sylvain Auroux's notion. The conclusion reflects on the limitation of our strategy, when compared with other programming languages. We try to assess the relevance of this Esperanto-like, high-level programming language for digital humanities.

2 Developing an R-based culture for NLP, DH and beyond

2.1 Why is it necessary?

In France, the study of English Linguistics in English departments has traditionally been linked to the competitive exams to become teachers of English (*agrégation*), so that linguistics is only a sub-domain in relation to other domains of English studies such as translation and literature. As a consequence, the core of this curriculum can hardly be dedicated to corpus linguistics. There also is another structural (devastating) side effect for linguistic research: since there is not a single trace of NLP-driven questions for the *agrégation*, there is nothing in the curriculum of English studies about these issues (contrary to what the introduction of English phonology somehow triggered after 2000 in the *agrégation* and in the undergraduate programmes that are meant to prepare for this competitive exam). Since in most European Universities the rise of corpus linguistics and quantitative methods has become essential in Linguistics curricula, the gap in France between the *agrégation* and linguistic research is widening.

In corpus linguistics, the treatment of large corpora and complex datasets with many variables is getting increasingly frequent. However, the application of statistical models, as well as the use of NLP techniques such as parsers, Part of the Speech (POS) taggers or classifiers, among many other possibilities, require some familiarity with, at least, one programming language. Although most programs designed for corpus linguistics, such as AntConc (Anthony 2011), Cesax (Komen 2011), Sketch Engine (Kilgarriff et al. 2004), Wordsmith (Scott 2016); or within the French lexicometric tradition, Le Trameur (Fleury and Zimina 2014), or Le Gromoteur (Gerdes 2014) are presented in a graphical and more or less user-friendly interface (GUI) with built-in functions, the command line offers much more flexibility in terms of exploration and modelling of the data. For instance, R can be first used as a concordancer (see (S. Gries 2009) in order to explore a given corpus, and then, once the desired structures have been extracted, they can also be treated in R in terms of statistical analysis and/or modelling. Finally, a visual representation of the results of the analysis can be easily plotted.

Apart from all the statistical packages that can be used for a quantitative analysis of data, there are currently more than 50 packages for NLP available in the CRAN repository[1], some of them being particularly useful for linguists. Because our students have research questions on spoken or written data, they can find several R packages to suit their needs if they are taught how to use them. These are some of the specific packages our students are working with:

- Phonetics/Phonology. Students are presented with normalization issues and analyses of vowel systems drawing from packages such as phonTools (Barreda 2015) vowels (Kendall and Thomas 2010) emuR (Winkelmann, Jaensch, and Harrington 2017) and phonR (McCloy 2016), which allows for the treatment of data extracted with Praat (Boersma and Weenink, 2017), one of the most well-known pieces of software used in phonetics/phonology.
- Data mining/text processing: cleanNLP (Arnold 2017), koRpus (Michalke 2017) languageR (Baayen, 2011) tidytext (Silge and Robinson 2016) openNLP (Baldridge 2005) qdap (Rinker 2013). For the treatment and the exploration of written corpora, some of the functions that these packages offer are: automatic POS tagging, implementations of parsers (e.g. Stanford coreNLP and the SpaCy library implemented in cleanNLP),

[1] https://cran.r-project.org/web/views/NaturalLanguageProcessing.html

text trimming, mathematical modelling of vocabulary with LNRE models, automatic syllabification, measures of lexical diversity and readability…

2.2. Aspect of the current curriculum

The implementation of R-based modules in the BA and the MA has been taking place progressively. Here we detail the various courses and activities that are currently implemented in our curriculum.

Undergraduate module. We present R among other software used by linguists in a module centred on student's projects, following *Language and Computers* . This introductory module also aims at getting undergraduate students to be interested in pursuing our MA research program.

MA seminars. Although during the first year of the MA efforts are made to encourage students to start using R as a way to process their corpus-based data, it is during the second year of the MA that a seminar on R is offered. This seminar consists on 6 sessions of 120 minutes covering the use of R for phonetics and phonology, and 6 sessions of 120 minutes on the use of R for textual data. Over the two years of the MA, the introduction to R-based packages in the curriculum takes the following form, in the different relevant modules (cf. Table 1):

Table 1: R in the MA modules

Year	First semester	Second semester
M1	Corpus methodology: descriptive statistics	Phonetic Analysis 1 (normalisation, plots, visualisation)
M2	Language and Variation (inferential statistics)	Computational phonology (classifiers)/ Phonetic Analysis 2

PhD seminar. Currently, our graduate school offers 12 sessions of 90 min covering advanced statistical techniques for linguists. A basic command of R is presupposed, especially since MA seminars already cover the use of R for beginners. This seminar mainly focuses on the applicability of statistical methods to different types of data linguists deal with, but also on the mathematical formulae behind all these methods. Empirical cases (already published by linguists) are used as examples. This seminar covers probability distribution, linear regression models, ANOVAs, linear discriminant analysis, principal component analysis…

Data sessions. As a follow up to the Statistics seminar, PhD students can discuss their data during data sessions. In these sessions, our students have the opportunity to present their linguistic dataset to a statistician. Ideally, the student has already made some progress on a basic level and knows how to present the structure of the data and detail the kind of investigated variables. Together with a statistician, they explore all the different methods that can be applied according to what the student's project requires.

Individual work. Because we are conscious that our curriculum cannot cover all the formation on R that our students would need, some individual extra work is needed. In that sense, specific manuals for linguists such as *Analyzing Linguistic Data: a practical introduction using* R (Baayen 2008), *Quantitative Corpus Linguistics with R* (S. Gries 2009), *Quantitative Methods in Linguistics* (Johnson 2011), *Statistics for Linguists* (S. T. Gries 2013), *How to do Linguistics with R*, (Levshina 2015), *Data Humanities with R* (Arnold and Tilton 2015) are recommended knowing that they have different pre-requisites in mathematics. Although not all these manuals have the same approach in regards to the technicality and the progression path, all of them offer a comprehensive view of what linguists can do with R. For a more detailed summary and comparison between some of these manuals, see Ballier (Ballier, forthcoming).

Bootcamps. Bootcamps are regularly organized, both for complete beginners and for intermediate users who seek to improve their abilities in R and to discover new techniques. Generally, one basic bootcamp is proposed at the beginning of the year for all students who want to participate, and a second bootcamp is later offered for more advanced learners. Bootcamps are conceived as an intensive way to approach particular issues (e.g. regression models, visualisation, classifiers…) and consist of a week of 25-30 hours of instruction. Bootcamps are normally taught with examples of datasets taken from published papers or on-going research, but students may also explore their own data if applicable. These official bootcamps are normally instructed by visiting scholars, such as Stefan Gries or Taylor Arnold.

3 R-based strategies for English studies

In the previous section, we presented the R courses and activities that are currently taking place in our programme, but we think that it is insufficient, especially during the MA years. In this section, we present all the aspects that should be taken into account when considering an R-based curriculum for English Linguistics studies. We aim at establishing course modules that present statistical and NLP techniques currently used in linguistic research. We know that getting to use command lines and to work with a programming language takes a lot of time for non-specialists, but it also offers a lot of new possibilities for linguists. This section details some strategies to ease R's steep learning curve.

3.1 Mathematics for linguistics and interdisciplinarity

One key feature of our strategy is the collaboration between statisticians, mathematicians, programmers, and humanists. Although we want our students to be independent users of R and to understand what they do when they use and manipulate data with NLP techniques, we do not expect them to become professional programmers. However, we do expect linguists to become familiar enough with NLP techniques, programming and statistics, so that they can efficiently identify what they need to advance in their research in terms of technical treatment of their data. This interaction between experts from different fields promotes cross-fertilization between the domains, and it also allows mathematicians and programmers involved in DH and in NLP to understand linguists' needs.

However, one of the issues we face is the creation of a reasonable time schedule for a progressive acquisition of all these techniques without scaring students, knowing that most of them are rather averse to learning these methods. Another issue related to the pedagogy of these methods is the amount of maths that should be taught so that students get to know what they are doing with their data. On the one hand, if students are explained all the on-going mathematical methods at the same time as they are introduced to the technique, there is a risk that they do not understand the procedure and refuse to use it. On the other hand, if students do not receive the mathematical explanations of the method, they will never fully understand what they are doing with their data. The risk is that students may run theirs scripts too blindly.

3.2 Pedagogy of datasets and scripts

Datasets as such can be understood as a common aspect of the methodology that enables students to understand the logic behind all the possible statistical tests and NLP techniques. It is, in a way, a dialectic form of the transferability of knowledge: understand the mathematics used with something else than your specialty. Students then need to learn to adapt the code to their own needs (dataset and research questions), as well as getting familiarized with importing-exporting methods from the treatment of other datasets.

Part of the benefits of the R strategy can be cashed in with the on-line forums and scientific blogs. They heavily rely on few datasets when explaining complex methods, so that it makes sense to teach these recurrent datasets (mtcars, Titanic, iris) to our students so that they become more autonomous in understanding on-line explanations. The use of particular datasets such as the classical iris dataset is notable for classification problems or dimensionality reduction. Conversely, typical analyses based on linguistic data such as the Bresnan and Nikitina (2003)dataset (Johnson, 2008 [2011], Baayen, 2008) or even Jane Austen's novels (Arnold & Tilton, 2017) may also help by showing students the multiple applications of R from a linguistic perspective.

Graphs always help motivating students. Making datasets more attractive by showing visualisation techniques that catch students' attention, not only about the statistical procedure, but also about how this technique can be later visualized and presented in talks and vivas is definitely something to take into account.

Eventually, students may also rely on the easiness and the practicality of the script in order to reproduce, compare and share results. With RStudio, students can actually send their whole project (including the environment, datasets, graphs, objects created by them, and code) to their supervisors, for example.

The scripts actually represent a new model of exercise and teaching methodology: scripts replace the classical textbook, showing detailed and commented functions. Examples and datasets need to be previously adapted to this format. It is an example-based methodology that promotes autonomy: in many cases, the script is not only the example, but also includes the exercises that need to be executed. The difficulty of these exercises increases progressively. This is, in a way, the creation of a new didactic method: most textbooks rely on a specific R package, R

scripts and companion website. The companion website to Levshina's textbook is a good case in point[2].

3.3 Package-driven pedagogy

An alternative method for students who seek to discover the usefulness of some specific R packages to linguistic research is to present them an MA research project related to the functionalities of one particular package. For instance, the {koRpus} package implements 14 metrics of lexical diversity and 35 metrics of readability that can be automatically computed. Computing each one of these metrics individually for a corpus made up of hundreds of texts results time-consuming and unnecessary. Therefore, the student had to learn how to use R and to explore the functionalities of the package: starting from the POS tagging with TreeTagger (Schmid 1995), the interpretation of the (huge) numerical resulting dataset with all the results of the formulae, the correlations of the various metrics, ANOVAs to compare results between different groups… Getting students to work with a specific interest for an R package is another way to foster acquisition of the package itself but also of R.

3.4 We love you just the way you 'R': A typology of teaching styles

R is known for its steep learning curve, counter-intuitive to real programmers. How is it taught in the reference textbooks? In general, most textbooks for linguistics with R do not assume any prior experience with any programming language, or any knowledge in statistics. They start by giving basic notions on descriptive statistics, and complexity increases progressively. However, not all books present the same degree of mathematical complexity, and not all books detail the mathematical processes ongoing behind the various commands and functions used in R. Therefore, the choice of one book or another may depend on the interest of the students, their background in maths/programming, and their own attitude towards NLP and statistics. When it comes to courses, the instructor faces the same type of issues. Not all students follow the same progression curve, and this is mainly determined by their own background and profile. These approaches are not mutually exclusive. In an ideal curriculum, students should be confronted to several approaches and follow whatever suits their personality best. So far, we have identified four main teaching approaches:

a) Scaring literary/linguist students for their greatest benefit. Teaching the general benefit of learning R as an interface in itself. This is mostly relevant for future PhD students. Among the MA students, the strategy consists in insisting on the advantages of the existing libraries, and the developer's community at large. More specific packages {ZipfR} are presented and functions are detailed for their interest for linguists but also in relation to a research community representing a (professional) lifetime investment with possible job prospects outside the linguistic community proper, provided time is spent on learning enough statistics and other interdisciplinary skills. In other words, to make the most of their potential role as interface in the DH, linguists of English are advised to invest at the same in statistics, not to say Data Science, for longer term benefits.
b) R for statistics in disguise: this approach consists in presenting techniques from a mathematical point of view; showing, detailing and explaining formulae. Although it offers the possibility to really understand what is going on with the data, students normally get scared and lose interest on quantitative methods because they consider them to be too difficult. This methodology might be particularly useful for intermediate and advanced students, but not so much for beginners.
c) Motivating students. Contrary to the previous methodology, here students are introduced to the various techniques without considering the on-going mathematical processes. Although it motivates students because they can see 'quick results' and the applicability to their data, it does not cover a deep analysis of the processes the data is going through. The emphasis is on learning simple code, for instance, heavily relying on the recurrent functions of the tidyverse collection. Displaying fancy visualisations with simple code also helps catching students' attention.
d) Intermediate point: teaching how statistics are reported in the journals. This second approach gives students the basic maths to understand descriptive and inferential statistics, as well as reported results from linguistic journals. It prepares students to understand what they will be reading in quantitative lin-

[2] https://benjamins.com/sites/z.195/

guistics. However, it does not get into much detail when it comes to more complex techniques.

Lastly, one should not underestimate the importance of online forums. They are complementary to any teaching methodology. R has a big community of users that answer quickly on forums, mailing lists of R packages for doubts and bugs are usually very active, pages such as Stackoverflow[3], r-bloggers[4], and many other websites also offer a comprehensive amount of help and code that is relatively easy to understand. Being part of the R community is being part of a community of experts that speak the same language, in spite of their fields of expertise, and students may benefit from being part of this wider research community.

4 The bigger picture: NLP, DH and the quantitative turn.

A set of analytical practices is being established (while being at the same time constantly revised) in the industry and in the research community, but the field is evolving so quickly that this knowledge has not made it to the modules taught in Humanities yet. At best, competing textbooks about *How to do* are being produced, but they reflect on individual solutions more than on academic curricula. NLP teachers have introduced modules about Machine Learning, but the curriculum in most NLP departments is still being revised to take into account the constant changes in the perimeter of existing technologies.

One should distinguish between sets of tools and methods, text mining, data mining, and data science. These labels interact with NLP as such, but need to be nuanced. One of the reasons for the complexity of contemporary curriculum design for Humanities is that NLP in this framework is no more a means in itself, or an autonomous curriculum in the MA degree we describe. For linguists from English faculties, the NLP language game can be caricatured as getting the best F-score you can for a given task/dataset in a challenge. NLP as a field may well become NLP as set of tools in DH (or a step in processing chains).

While one may argue that academic political decisions are mostly designed by means of conflicting calls for projects in an indirect top-down approach, here is a bottom-up approach that tries to address what should be the real order of the day: what does it take to turn a motivated student from an English department into a DH specialist (if not a data scientist)?

DH is an emerging field where curricula still need to be designed. Felicitous encounters foster cross-disciplinary achievements, but how do we enforce these developments in our curricula? There is a historical responsibility in curriculum design to face the challenges of DH. More than a political agenda, there is an epistemological transition going on in terms of required skills (and knowledge) to process data and knowledge. Advocating a programming language with a strong background and tradition in maths rather than mere NLP modules is also a unique possibility to bootstrap Data Science in Humanities curriculum. Modules in Maths would be required, but having acquired an R culture may ease this transition to Maths. What follows builds on existing modules to outline possible developments: here we sum up putative modules around R that may help students of English to make a transition towards a postgraduate programme in data mining:

- **First semester:** mathematical bases of data mining. Pedagogical packages related to a particular manual (e.g. {languageR, car, caret}). Functions and Datasets to accompany *An R Companion to Applied Regression* (Fox et al. 2014) and *Implementing reproducible research* (Stodden, Leisch, and Peng 2014).
- **Second semester:** For data mining, *Data mining and analysis: fundamental concepts and algorithms* (Zaki, Meira Jr, and Meira 2014). For statistical modelling, *Applied predictive modelling* (Kuhn and Johnson 2013).

5 Philosophical Implications for R as a medium for the 3rd revolution of grammatisation

This section draws the bigger picture that explains why we promote the teaching of R for DH. The main revolution in this kind of faculties consists in convincing students to use the command line. RStudio is an interface that reassures students because it has windows and a GUI, it enables them to run scripts and learn how to comment script. It is a good compromise with a console and loading functions are actionable with the mouse as with any GUI. R commander (and similar plugins as the one proposed with the

[3] https://stackoverflow.com

[4] https://www.r-bloggers.com

{Rattle} package) were previous attempts at simplifying R (not to mention RExcel and other Excel-based interfaces) with the same click and play approach. We would like to suggest that as a programming language and as a programme giving access to thousands of packages, R has a special status for the DH turn under way (not to mention the fact that any epistemological angle on NLP tools should consider the programming side).

The DH turn is closely related to the quantitative turn and we consider that this piece of software takes part in what we call, after Sylvain Auroux, the third revolution of 'grammatisation'. According to Auroux (Auroux 1994) the first revolution of the 'grammatisation' was related to writing. With Gutenberg, speech enjoys some specific codification. Textual structures (paragraphs, books) play the role of technological innovation that preconditions linguistic analysis.

The second technological invention that revolutionized linguistics according to Auroux was the invention of dictionaries and grammars, especially in 18th century Europe. Again, codification of the language, and standardisation of spelling triggered some reflection about lemmatisation. Linguistic data was processed according to a certain format (e.g. dictionary entries).

Auroux (1994) suggests that the emergence of corpora and NLP techniques boils down to the emergence of a third revolution of grammatisation, which is still under way. We see R as a possible interface between corpus linguistics and the quantitative turn, our free access to statistical libraries and NLP tools. One of the benefits is that R can increasingly be used as a concordancer, mining corpora, especially with the tidyverse collection of packages. Adopting R facilitates a roadmap to data science, since corpus extractions end up as a dataset. The current 'tidy data' (Wickham 2014) philosophy favours the structure of the data frame where 'each type of observational unit is a table'.

We believe that the DH are a crossroad for the cross-fertilization of several disciplines: linguistics (where NLP is crucial), statistics, and the emerging domain of data science. In this respect, R is an excellent candidate as a tool to promote real pluridisciplinarity. Conceptions and boundaries vary as to the content of machine learning, data mining, and data science, but the recent evolutions within the tidyverse collection of R packages facilitate text mining. The common denominator to this emerging field is text mining: 'Text mining is an interdisciplinary field which involves modelling unstructured data to extract information and knowledge, leveraging numerous statistical, machine learning, and computational linguistic techniques.'. R packages such as {koRpus} (Michalke, 2017) and {tm} (Feinerer and Hornik 2015) make text mining with R much easier. We could even say that they favour hybrid uses between concordancers and the standard NLP blind processing of data.

Again, the uses of R are becoming increasingly more user-friendly than they used to be. Recent publications have heralded the emergence of accessible approaches to text mining, where NLP is in disguise. For example, Jockers (2014) distinguishes between micro-, meso- and macroanalysis. Micro-analysis deals with frequency analysis and correlation between the presence or absence of a word in a text with randomization techniques. Mesoanalysis consists in detailing lexical complexity, hapax analysis and teaching how to build KWIC-type concordancing with R. Macroanalysis presents unsupervised clustering and some initiation to support vector machines supervised learning models.

This more recent approach allows for more flexible analyses where individual texts in a corpus may be taken into account more flexibly than with other command line or corpus-based methods. The interaction with visualisation techniques and the flexibility offered by the tidyverse collection facilitates the focalisation on the mining of texts. As the 2017 rOpenSci Text Workshop puts it[5], working with these recent r packages encompasses 'text analysis, natural language processing, and other aspects of text mining and text data handling'. Because it can be used as an interface for these practical and theoretical issues, R is a good candidate for the development of hybrid approaches to text mining, mixing concordance-based approaches and more ambitious analyses of metadata and quantitative (textometric) aspects of texts in a single environment.

Data visualisation and all of the flexibility of the R environment posit R in a very favourable situation as a specific tool for the third revolution of grammatisation that is taking place around text mining. This can be seen with the flowcharts describing the interaction of R packages and tidyverse functions for topic modelling in Silge and Robinson (2016), or the R open science efforts for the interoperability of formats in some

[5] http://textworkshop17.ropensci.org/

packages (the text interchange format package, {tfi}[6]).

6 Conclusion: limits and difficulties

The conclusion reflects on the limitations of our curriculum and achievements as well as the drawbacks of our choice of R. The first part discusses actions that are still under way. The second part summarises some of the issues against R.

Alternative strategies still to be tested include reverse teaching for R packages, sessions where students would have to present an R package and what can be done with it, which is a way to get them acquainted both with the code and the bigger picture: the *why*? and the *what for*?

Another strategy to be tested consists in supervising an MA whose every step implies specific coding in R, to the point of designing the requirements of an R package along the MA, each cornerstone of the MA corresponding to an elaborate R function which needs to be implemented to process the data. Designing your own R package centred around your research question is an option, more likely to be realised at the end of a PhD. We are encouraging PhD students reaching the end of their thesis to design their own packages, encapsulating their codes and partial datasets, to join the github community in order to bypass the heavier requirements of the CRAN repository and propose their packages as prototypes or proofs of concepts through the devtools library.

What would make sense in a pluridisciplinary university would be to teach a general introductory course resembling a seminar centred on the versatility of R while teaching core linguistic concepts to non-specialists and presenting useful R packages for linguistically-related research questions. The challenge is to teach basic coding, linguistic notions and R packages in lecture halls. Nevertheless, a module like "Language, Texts and Data mining: An R-based introduction to Digital Humanities" should be offered to undergraduates to promote DH.

Suggesting that R should be essential to students contributes to developing a coding culture and foster on-line learning (as stackoverflow is your friend) but it makes sense in terms of lifelong learning. Our MA students get the extra-benefit of an initial training to data mining, however basic it may sound to a professional data scientist. The essential roadmap to emancipation still has to be designed. We are not to turn our students of English linguistics into programmers, but we want our PhD laureates to be as proficient as can be. Replicating and adapting a script to the needs of your data is one thing, being an expert is quite another.

As to the limitations of R, some of them are inherent to the language, some can be related to the misuses of R. The first limitation is that a programming language is not NLP as such. Becoming acquainted with some logic of packages gives access to specific resources, but with more limitations as to the languages under scrutiny than with Python-based tools.

Regarding the internal limitations of the programming language, known problems with R include issues related to the floating point calculations (which seem to play a role in the R word2vec implementation of the word embedding algorithms), a quirky syntax, issues with loops and the way everything is stored in the memory. With specific data processing (very often, phonetic data), matlab libraries and scripts have been developed so that R is superseded in these areas, not to mention the fact that the average documentation is usually more complete, but this service comes with a price, whereas R is free.

Last, we would like to report on some issues that have been encountered by students. On top of classic mismanagement of codes for models or serious issues with data coercion with R, we would like to report a typology of attitudes which may reflect more poorly on some of infelicitous uses of R. These attitudes are mainly related to what one could call 'the encyclopaedic ignorance of the self-taught linguists': becoming experts at secondary details but missing the basic maths. This means being able to execute relatively complex methods but not knowing exactly what the model does to the data. Another variant is linked to wanting to know everything about the R code, without considering the big picture, i.e., the underlying mathematical model required to address the data, practicing some sort of code-induced short-sightedness. It may well be the case that this package-based approach to R distorts the representation of a programming language. Full empowerment of students leading to the possibility of writing a programme should be the real of the day, whereas we probably endorse some empowerment limited to existing packages and scripts in a kind of solving problem based philosophy. Clearly, learning another programming

[6] https://github.com/ropensci/tif

language would help for this kind of learning profile.

Promoting the teaching of R is an important aspect of pluridisciplinarity (and an easy way to start building common background with the maths department) but should be seen as the first step. Beginning with R, students may move to Python (for example using jupyter or Anaconda) and then for example use the scikit-learn ML in Python (Pedregosa et al. 2011).

The new challenges posed by this new configuration of knowledge is mostly unheard of. Most learning paths for the different intersecting subdisciplines are still unchartered territories. For this complex language game ahead of us, linguists may lack computational linguistics, but other scientific partners will need more linguistics for true interdisciplinarity.

References

Laurence Anthony. 2011. *AntConc* (Version 3.2. 2)[Computer Software]. *Tokyo, Japan: Waseda University*.

Taylor Arnold. 2017. 'A Tidy Data Model for Natural Language Processing Using cleanNLP'. *arXiv Preprint arXiv:1703.09570*.

Taylor Arnold, and Lauren Tilton. 2015. *Humanities Data in R*. Springer.

Sylvain Auroux. 1994. *La Révolution Technologique de La Grammatisation. Introduction À L'histoire Des Sciences Du Langage*. Liège: Mardaga.

Harald R. Baayen. 2008. *Analyzing Linguistic Data: A Practical Introduction to Statistics Using R*. Cambridge: Cambridge University Press.

Jason Baldridge. 2005. 'The Opennlp Project'. *URL: Http://Opennlp. Apache. Org/Index*

Nicolas Ballier. forthcoming. 'R, Pour Un Écosystème Du Traitement Des Données? L'exemple de La Linguistique.' In *Données, Métadonnées Des Corpus et Catalogage Des Objets En Sciences Humaines et Sociales.*, edited by Ph Caron. Presses universitaires de Rennes.

Santiago Barreda. 2015. 'phonTools: Functions for Phonetics in R'. *R Package Version 0.2-2.1*.

Paul Boersma ,and David Weenink. 2017. *Praat.* (Version 6.0.29) [Computer Software].

Joan Bresnan, and Tatiana Nikitina. 2003. 'The Gradience of the Dative Alternation.' *Stanford University*. Retrieved from http://www-lfg.stanford.edu/bresnan/download.

Ingo Feinerer, and Kurt Hornik. 2015. *Tm: Text Mining Package* (version 0.6-6). https://cran.r-project.org/web/packages/zipfR/index.html.

Serge Fleury, and Maria Zimina. 2014. 'Trameur: A Framework for Annotated Text Corpora Exploration.' In *COLING* (demos), 57–61.

John Fox, Sanford Weisberg, Daniel Adler, Douglas Bates, Gabriel Baud-Bovy, Steve Ellison, David Firth, Michael Friendly, Gregor Gorjanc, and Spencer Graves. 2014. 'Companion to Applied Regression R Package.Version 2.0-20'.

Kim Gerdes. 2014. 'Corpus Collection and Analysis for the Linguistic Layman: The Gromoteur'. In *proceedings of the JADT*.

Stefan T. Gries. 2009. *Quantitative Corpus Linguistics with R. A Practical Introduction*. New York-London: Routledge.

Stefan T. Gries. 2013. *Statistics for Linguistics with R: A Practical Introduction*. Berlin: Walter de Gruyter.

Matthew L. Jockers. 2014. *Text Analysis with R for Students of Literature*. Springer.

Keith Johnson. 2011. *Quantitative Methods in Linguistics*. Chicester, UK: John Wiley & Sons.

Tyler Kendall, and Erik R Thomas. 2010. 'Vowels: Vowel Manipulation, Normalization, and Plotting in R. R Package, Version 1.1'. *Software Resource: Http://Ncslaap. Lib. Ncsu. Edu/Tools/Norm/*.

Adam Kilgarriff, Pavel Rychly, Pavel Smrz, and David Tugwell. 2004. 'The Sketch Engine'. In *Proceedings of Euralex*, 105–16.

Erwin R. Komen. 2011. 'Cesax: Coreference Editor for Syntactically Annotated XML Corpora'. *Reference Manual. Nijmegen, Netherlands: Radboud University Nijmegen*.

Max Kuhn, and Kjell Johnson. 2013. *Applied Predictive Modeling*. Vol. 810. Springer.

Natalia Levshina. 2015. *How to Do Linguistics with R: Data Exploration and Statistical Analysis*. John Benjamins Publishing Company.

Daniel R. McCloy. 2016. *phonR: Tools for Phoneticians and Phonologists.* (version 1.0-7). https://CRAN.R-project.org/package=phonR.

Meik Michalke. 2017. *Package koRpus: An R Package for Text Analysis* (version 0.10-2). http://reaktanz.de/?c=hacking&s=koRpus.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, and Vincent Dubourg. 2011. 'Scikit-Learn: Machine Learning in Python'. *Journal of Machine Learning Research* 12 (Oct): 2825–30.

R Core Team. 2016. *R: A Language and Environment for Statistical Computing.* (version 3.3.1 (2016-06-21)). English. Vienna, Austria.: R Foundation for Statistical Computing. https://www.R-project.org/.

Tyler W. Rinker. 2013. *Qdap: Quantitative Discourse Analyisis Package* (version 2.1.0.). University at Buffalo/SUNY, Buffalo, New York.

Helmut Schmid. 1995. ‘Treetagger: a Language Independent Part-of-Speech Tagger’. *Institut Für Maschinelle Sprachverarbeitung, Universität Stuttgart*.

Mike Scott. 2016. *WordSmith Tools, Stroud: Lexical Analysis Software.* (version 7).

Julia Silge, and David Robinson. 2016. *Tidytext: Text Mining and Analysis Using Tidy Data Principles in R.*

Julia Silge, and David Robinson. 2017. *Text Mining with R: A Tidy Approach*. O’Reilly Media, Inc.

Victoria Stodden, Friedrich Leisch, and Roger D. Peng. 2014. *Implementing Reproducible Research*. CRC Press.

Hadley Wickham. 2014. ‘Tidy Data’. *Journal of Statistical Software* 59 (10): 1–23.

Raphael Winkelmann, Klaus Jaensch, and Jonathan Harrington. 2017. *emuR: Main Package of the EMU Speech Database Management SystemR Package Version* (version 0.2.3.). https://CRAN.R-project.org/package=emuR.

Mohammed J. Zaki, and Wagner Meira. 2014. *Data Mining and Analysis: Fundamental Concepts and Algorithms*. Cambridge University Press.