

A L^AT_EX and BibT_EX Tutorial for Linguists

Original author:

Jackson Lee, University of Chicago
jsllee@uchicago.edu

Updated and expanded by:

Adam Roth Singerman, University of Chicago
adamsingerman@uchicago.edu

Daniel Puthawala, The Ohio State University, University of Chicago
puthawala.1@osu.edu

Christian Clark, The Ohio State University
clark.3664@osu.edu

November 9, 2024

Abstract

This is a L^AT_EX template for linguists who are new to L^AT_EX and who would like to learn how it can be used to create professional documents. This tutorial goes over the basics of L^AT_EX document formatting and shows you some basic tools that are extremely important for linguistics scholarship, including automatically aligned interlinear glossing and fancy shmancy trees. Since this tutorial is always a work in progress, please feel free to send e-mails to any of the authors listed above. Compliments should be reserved for Jackson, who authored the original version; criticisms should go to Adam, Daniel, and Christian.

1 Introduction

Welcome to the world of L^AT_EX! The learning curve is fairly steep, but your efforts will pay off as you gradually get the hang of this amazing tool. As you move along, there may be moments when things don't quite work as intended and you get frustrated. Google usually has answers to the questions you have (search by "latex xxx").¹

¹Also: <http://en.wikibooks.org/wiki/LaTeX>. By the way, this is a footnote. To introduce it, just use `\footnote{}`, with whatever text your note is to contain in between the curly brackets.

2 The Five Commandments of L^AT_EX

1. Start with a template and adapt
2. Compile often (every other sentence, or more often for equations, tables, etc.)
3. Ask someone if you get stuck. Google, Stack Exchange, and Detexify are your friends!
4. If you have an error, check for balanced { }, [], \$ \$
5. If you have an error, comment out what you just wrote and see if it fixes itself; then you know where the problem is

3 Packages

3.1 What are packages?

Packages add new capabilities to your L^AT_EX system, so to speak. They define new commands that do something you need. Quite often, different packages can do the same thing.

3.2 How do we tell L^AT_EX which packages we are using?

In the .tex file, the code preceding `\begin{document}` is called the preamble. You call the packages using the `\usepackage` command in the preamble. Side track: By now, probably you've noticed that all commands begin with the backslash symbol `\`. (Then how do we actually type and show a backslash? See the L^AT_EX code for the answer. This will happen a lot during this tutorial.)

3.3 Where do packages come from?

There are packages that are useful for L^AT_EX users in general, as well as those more specific for linguists. Most of them are available on one of the repositories readily accessible by your L^AT_EX distribution. So if you are trying to use a new package that your computer hasn't downloaded yet, it should be downloaded automatically when you hit "Typeset" (assuming correct configurations).

3.4 How do we learn to use particular packages?

Packages come with their documentation. If you want to try out or learn how to use a package X, google "latex X documentation" and a .pdf file is usually located – it should be the documentation, or manual, of that package.

4 Formatting

4.1 Comments

One of the wonderful things about L^AT_EX is that you can include content in the .tex file that doesn't show up in the output .pdf. We call these COMMENTS. In your .tex file, anything that comes on a line of text after the % symbol is ignored in compiling the .pdf file. This is useful for making notes as well as making your .tex much more readable. For several examples, just take a look at the code of this file. You'll notice, for example, that each of the packages called up in the preamble has a helpful summary following the %. These summaries have absolutely zero impact on the .pdf that gets produced.

4.2 Indentation and spacing

This is a paragraph. Take note of its indentation.

This is another paragraph. Also take note of its indentation here. In the .tex document, you separate two paragraphs by a blank line.

This is a third paragraph. Here we used the `\noindent` command to get rid of the usual indentation!

This is a fourth paragraph. We added some vertical space before this one with `\vspace{}`. You can also use `\hspace{}` to add horizontal space like so.

4.3 Another subsection!

Basic stuff: **bold**, *italic*, SMALL CAPS, underlined

4.3.1 WOW – a subsubsection

Quotation marks. Compare the L^AT_EX code and the quotation marks displayed here very carefully: “hello”, ‘hello’, ’hello’, ”hello”.

5 Symbols

Accents. Vous êtes... Très bien !

IPA symbols? Package: `tipa` Examples: əabcdeɑβεð̥εiʌɜɸ

6 Tables

Bare minimum to create tables:

Table 1:

1 2 3
 4 5 6
 7 8 9

Table 2:

1	2	3
4	5	6
7	8	9

Check out the package `booktabs` for more professional tables.

7 Morpheme glossing and in-text referencing

We will use the `Linguex` package for several important tasks, such as inserting examples, and doing glosses. Look a how we can easily make multiply embedded examples below!

- (1) This is the first level of embedding
 - a. This is the second level
 - b. This is still the second level, but:
 - (i) This is the third level
 - (ii) This is not the end.
 - (iii) This is the end of the innermost embedding
 - c. and this is the next item in the second level
- (2) This is the next example/table/figure

We can also use the `Linguex` package to do easily do glosses, and we can use the regular label and refer to examples and glosses in the following manner. Check out the text in the .tex file here and in (3), (4), and (5), as well as in (6) and (7).

- (3) Dies ist eine Glosse
 This is a gloss
- (4) Dies ist nicht eine Glosse
 This is not a gloss
- (5) %*This English not is.
- (6) vajafav josef mitseʁajima
 return.PERF.3.PL.MASC Joseph Egypt.ward
 ‘Joseph returned to Egypt’ (Genesis 50:14)
- (7) vajmahεʁ aveʁaham haʔohela
 hurry.PERF.3.SG.MASC Abraham DEF.tent.ward
 ‘Abraham hurried (in)to the tent’ (Genesis 18:6)

As with regular labeling and referring, you can even drag labeled things around, and \LaTeX will automatically renumber the examples appropriately and keep the reference numbers connected to the same example as before. More on this below in Section 9.

8 Making glossing even easier

- (8) The students all adore me.
The student-s all adore- \emptyset me.
ART student-PL all adore-PL 1SG.ACC
The students all adore me.

It is a major waste of time to have to rewrite 1SG.ACC every time you need to gloss the pronoun ‘me.’ So we can just write a command that stores this gloss for us:

- `\newcommand{\megl}[0]{\textsc{1sg.acc}}`

This tells LaTeX to make a new command with the name `\megl` (for ‘me-gloss’, though you could call it anything you want). This command will take zero arguments. It will automatically produce the text 1SG.ACC, formatted in the small caps traditionally used for glossing.

You can make new commands to automate all kinds of glosses:

- plural: `\newcommand{\pl}[0]{\textsc{pl}}`
- singular: `\newcommand{\sg}[0]{\textsc{sg}}`
- dative: `\newcommand{\dat}[0]{\textsc{dat}}`
- nominalizer: `\newcommand{\nmlz}[0]{\textsc{nmlz}}`
- and so on and so forth.

Check the \LaTeX code here for one more super-useful custom command!

With \LaTeX , it becomes easy to implement industry standards like the Leipzig Glossing Rules (<http://www.eva.mpg.de/lingua/resources/glossing-rules.php>).

9 Labeling and referencing

One of the best things about \LaTeX is the ability to label and then refer back to examples, sections, subsections, and so on. The trick to making this happen are the following commands:

- `\label{}`
- `\ref{}`

- `\eqref{}`

Here’s how it works. When you write an example, you can add in a `\label{}` command, with the label of your choosing inside the brackets. Take a look at the code for the following examples (which also introduce the useful `\hfill` command):

- (9) Names of birds in Hungarian:
- | | | |
|----|---------|------------|
| a. | madár | ‘bird’ |
| b. | papagáj | ‘parrot’ |
| c. | sólyom | ‘hawk’ |
| d. | varjú | ‘crow’ |
| e. | fácán | ‘pheasant’ |

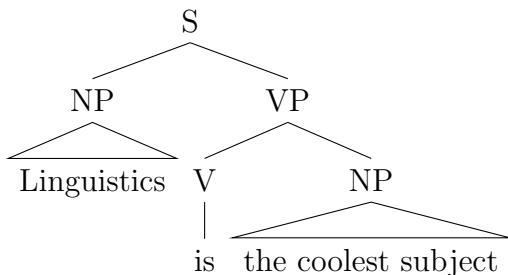
If you take a look at the code, you’ll notice that I have placed a `\label{}` command for each example. I can use `\eqref{}` or `\ref{}` to refer back to these labels.² L^AT_EX takes care of the numbering automatically. These labels can nest, as in (9) versus (9-a), (9-b), (9-c), (9-d), and (9-e).

I can also label and refer to sections of the document. Take another look at the code: there’s a `\label{labeling}` command right at the beginning of this section. I can refer back to the number of this section painlessly with `\eqref{}` or `\ref{}`: (9). Ta-da!

10 Syntax trees

The best and easiest package for making beautiful and (hopefully) convincing syntax trees is with the packages `tikz-qtrees` and `tikz`. (You will need *both* packages for tree-drawing!)

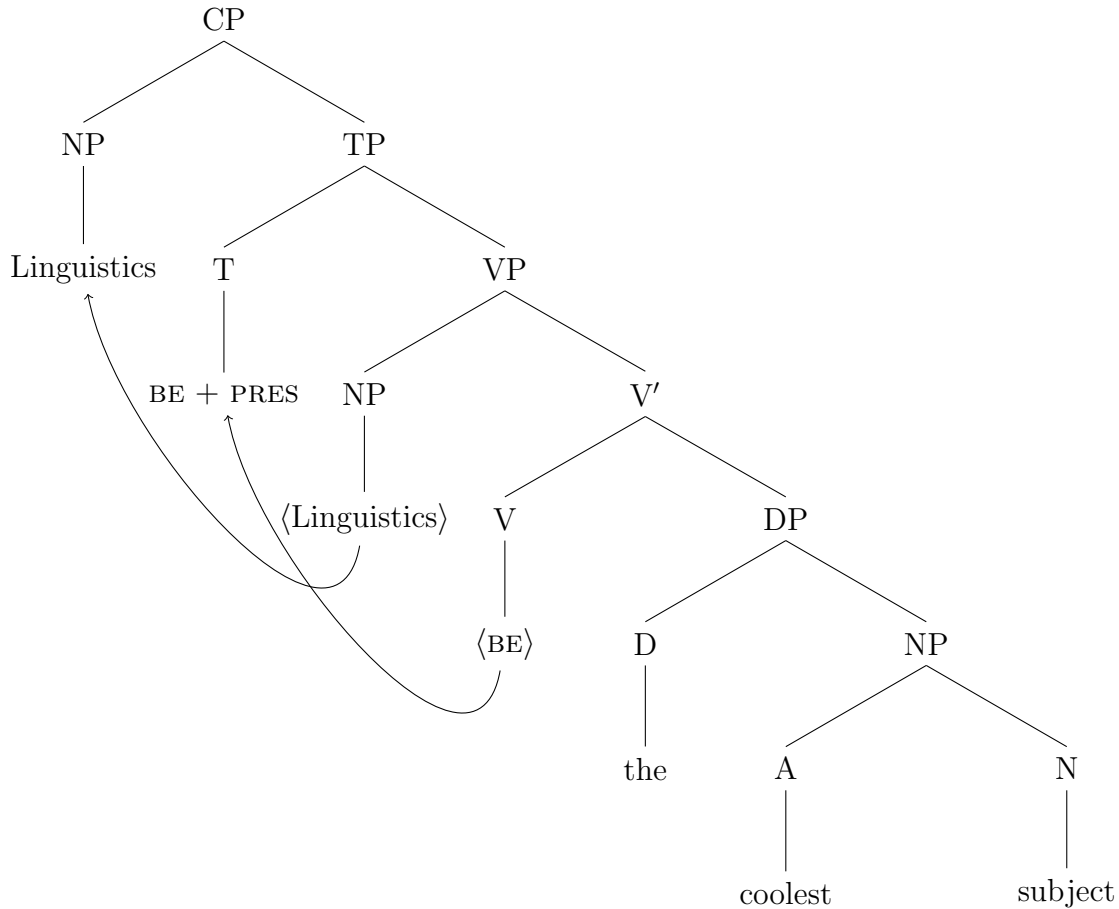
If you’re going to draw a tree that doesn’t have any movement arrows, the easiest means is to use the `\Tree` command within the `tikzpicture` environment. Here’s an example:



Make sure you put a space before every closing square bracket in your code when you’re working with the `\Tree` command. If you don’t do so, the tree won’t work!

Slightly more complicated (but considerably more powerful) is the `\node` command. Take a look at the following code to see how the magic happens.

²The difference is that `\eqref{}` adds a parentheses around the example number. However, `linguex` example numbers seem to already have parentheses by default, so `\ref{}` works fine for them.



With the `\node` command, you can draw trees by specifying their nodes and the children of those nodes and so on and so forth.

11 Logic Trees

A common and straightforward way to make logical proof trees is with the package `bussproofs`.

$$\frac{\frac{\Gamma \vdash A \quad A \rightarrow B \vdash A \rightarrow B}{\Gamma, A \rightarrow B \vdash B} \rightarrow E}{\Gamma \vdash (A \rightarrow B) \rightarrow B} \rightarrow I$$

12 Bibliographies and Citations

When, in the Course of human events, it becomes necessary for one person to use the academic bonds which have connected them with another, and to assume among the powers of the earth, the separate and equal station of reasoned argument to which the Laws of Nature and of the Academy entitle them, a decent respect to the opinions of mankind requires that they should declare where they got their ideas from. (Jefferson et. al. (Adapted), 1776)

In light of the above, we often need to cite our sources for papers and other projects. `BibTeX` is a plaintext format to make a database of our sources so `LATEX` can read in your sources and use them to create beautiful lists of references.

Generating a .bib File So how does one get a .bib file? While it is possible to simply type up your sources in a .bib file (take a look at ExampleBib.bib for an example), there is a better way. If you are using a citation manager like Zotero, you can easily export a your references in BibTeX format, which will automatically generate the appropriate .bib file. This will ensure that you don't make mistakes in the syntax, and it will let you quickly iteratively generate .bib files, or update it to incorporate new sources.

How to reference a .bib file There are two main commands you need to use in order to link a .bib file to your .tex file, and create a bibliography.

The command `\bibliographystyle{style}` is used to specify the citation style you want to use. If you google "bibtex citation styles", you can find many more! Often, journals and institutions will design their own style documents you can download and use. It is also possible (but slightly trickier) to define your own customized style format! Look in the code for the command here, which we specify the apalike style for the example citations in this document.

The second command, `\bibliography{FileName}`, is the command that will actually generate your bibliography from the file. If you move this command, the place your bibliography appears will also change. This allows you do do separate bibliographies for different chapters of a book, for instance, or to place the bibliography at the end. In the code of this tutorial, we use the command `\bibliography{ExampleBib}`. Since the file name is 'ExampleBib', this command will instruct L^AT_EX to look for a file called 'ExampleBib.bib' in the same folder as the .tex file. Look for this command to appear at the end of the document!

Compiling for BibTeX When you are connecting a .bib file to your .tex file, there are a lot of processes going on behind the scenes. There are several steps involved, such as reading the .tex document, looking through the .bib database for a matching source name, importing the source into the L^AT_EX auxiliary files, and then finally, figuring out the right format, and inserting the correct parenthetical citations into the text where they are referenced.

This means that in order to get citations to propogate through this process and appear correctly, we need to compile more than once, and in more than one way. In fact, here is the correct minimum compiles you need to do to go from inserting a new citation to having everything updated and the correct citation to appear:

(10) `pdflatex → bibtex → pdflatex → pdflatex`

Most Integrated Deveelopment Environments for L^AT_EX like TexMaker or Overleaf will do this for you, but you might need to tweak a setting first.

How to Cite! This tutorial makes use of the natbib package, which offers you a few different handy versions commands we can use to cite sources, as well as a greater list of style formats. When you cite something, you need to refer to the name of a source, which is the line of text that appears in the first line of each entry in the .bib file. So the name for the Fooley Barley's 2016 conference paper from the ExampleBib.bib file is fooly_barly_journal_2016.

One of Natbib’s options gives us `\citep{}`, which is useful for parenthetical citation when we want the entire citation to appear in parentheses: (Foo Bar, 2016)

One other useful Natbib command is `\citet{}`, which is used for textual citations, where we want to talk about the author in the text while citing them, as put forth in Foo Bar (2016).

Covertly Citing While you don’t usually want to vacuously or covertly cite works that you are not referencing in your project, it is often useful for troubleshooting purposes to cite something without any connection to the text. To do this, we use the command `\nocite{}` in the same manner as the above. Look in the code for the example citation

One more trick that’s useful when troubleshooting is to vacuously cite your entire bibliography. That way, if some citations are working and others not, you can more easily figure out what’s going wrong. For this, we use the same command as above, but instead of a title, we put in an `*` to mean “any and all.” So `\nocite{*}` should make our entire bibliography appear in the references section below.

13 Other Assorted Resources

- Anke Himmelreich’s \LaTeX for Linguists (very extensive!): https://www.ankehimmelreich.de/downloads/skript_latex.pdf
- Explanation about the difference between \LaTeX , MiKTeX, TeXWorks, etc.: <https://www.tug.org/levels.html>
- Overleaf, like Google Docs for \LaTeX : <https://www.overleaf.com/>
- LyX, a graphical \LaTeX editor: <https://www.lyx.org/>
- Full list of \LaTeX diacritics: https://en.wikibooks.org/wiki/LaTeX/Special_Characters#Escaped_codes
- IPA chart for textipa: https://jon.dehdari.org/tutorials/tipachart_mod.pdf
- Microsoft Excel-style \LaTeX table generator: <https://www.tablesgenerator.com/>
- Online TikZ figure editor: <https://tikzmaker.com/>
- Some general \LaTeX resources: <https://web02.gonzaga.edu/faculty/kearney/latex.html>
- Bonus: how to use \LaTeX to take your lecture notes: <https://castel.dev/post/lecture-notes-1/>

14 The End!

We hope that you've found this document helpful and easily understandable. Feel free to use it as a template for your own documents. An easy way to get started is to copy over the .tex file to a new location and rename it. Then open it as usual, and comment out the actual content (and delete the stuff you absolutely don't need). Use the old parts as patterns and examples to build your own fancy shmancy papers, homework assignments, squibs, QPs, and maybe even theses/dissertations! We sincerely wish you the best of luck and happy T_EXing!

References

Boo Far (2016). Manuscript Citation in BibTeX. Unpublished.

Foo Bar (2016). *Book Citations in BibTeX*, volume 1 of 1. Me, Midwest, 1 edition.

Fooy Barey (2016). Conference Paper Citation in BibTeX. In *Fooy*, volume 1 of 1, page 10, Midwest. Me.

Foolio Barlio (2016). Encyclopedia Article Citation in BibTeX.

Fooly Barly (2016). Journal Article Citation in BibTeX. *Applied BibTeX*, 1(1):10.

Foomian Barmian (2016). *Thesis/Dissertation Citation in BibTeX*. Dissertation, Some Midwestern University, Midwest.

Jefferson et. al. (Adapted) (1776). Declaration of Independence. In *Declaration of Independence*, pages 1 (vellum, large), Philidelphia. Colonial 'Merica.