Rendering IPA Symbols in R Markdown

I was thinking about promoting reproducible research in Linguistics, or more precisely, how to attract people with no programming skills to have incentives to learn at least a bit programming, so that they have the ability to make their research more reproducible.

I arrived at the solution: start by adopting R Markdown to write articles (see the last section for details), but making R Markdown more friendly to novices in a particular field of academia is crucial to enhance their incentives to learn programming.

Tasks Specific to Linguistics

I came out with some common tasks related to document writing in Linguistics (I will thank you if you tell me other tasks I missed):

1. Typing IPA symbols.
2. Drawing syntax trees.

To enhance R Markdown's ability to do these tasks without compromising one of it's great feature: render nicely to HTML and PDF with the same source, one need to consider the incompatiblity of LaTeX and HTML code.

Solving the first task (IPA symbol) is easy, draing syntax trees is hard and I haven't have a solution yet[1].

Typing IPA Symbols

There are two problems to be solved in order facilitate using IPA symbols in R Markdown:

1. Input method
2. Font support (only related to PDF output)

The first one is essentially about mapping some combination of keys to unicode strings. This post dementates how to solve the second, which is more fundamental.

[1] There are LaTeX packages supporting drawing syntax tree, but LaTeX package is not compatible with HTML output.

After doing a little research, I came out with a quick solution which stems from the combination of IPA Symbols in R, How do I use a particular font for a small section of text in my document?, and Conditional compilation of book chunks to ensure compatibility with both HTML and XeLaTeX.

The solution is very simple: define a new font family that supports IPA symbols in LaTeX and use conditional compilation to render the document: when compiled to HTML, use raw unicode string; when compiled to PDF, wrap LaTeX code around IPA unicode strings.

To define a new font family for IPA symbols, set `header.tex` and include it in the yaml header of R Markdown document:

```
% Set font size
\usepackage[fontsize=12pt]{scrextend}

% Set font family
\usepackage{xeCJK}
\usepackage{fontspec}

\setmainfont{Calibri}

\setCJKmainfont[
    BoldFont={HanWangHeiHeavy}
    ]{HanWangHeiLight}

% IPA font
\newfontfamily\ipa{Doulos SIL}
\DeclareTextFontCommand{\ipatext}{\ipa}
```

The font, Doulos SIL, which supports IPA symbols can be freely dowloaded.

The code chunk below is for conditional compilation.

```
ipa <- c('e\u026A', 'a\u026A', '\u0254\u026A')

if (knitr::opts_knit$get('rmarkdown.pandoc.to') == "latex") {
  ipa <- paste0("\\ipatext{", ipa, "}")
}
```

The IPA symbols are set in the variable `ipa` and can be access inline in R Markdown with, e.g., `r ipa` or `r  ipa[3]`, which renders to eɪ, aɪ, ɔɪ and ɔɪ, respectively.

Obstacles to Adopting a Reproducible Workflow

Skip this section if you're tired of stuff about reproducibility and R Markdown.

Reproducible research not only enhance scientific progress but also saves researchers a great deal of time, by automating repetitive and error-prone tasks in research. So if there are good reasons to adopt a reproducible workflow in research, saving time (in the long run) might be a good one.

Programming skill is fundamental to automating repetitive tasks, which saves one's time. However, learning programming to save time makes no sense to many people, since it is terrifying, hard, and time consumming[2]. So the problem now becomes:

> How to reinforce the incentive to learn programming?

Again, by showing people how to save time, but this time, programming skill is not required.

I think R Markdown is a very promising starting point, since writing is necessary for researcheres, and one can use RStudio without any knowledge of R. When becoming familiar with R Markdown, one begins to adopt a reproducible workflow and might notice the capability of R language, hence gaining more incentive to learn R.

Many people in academia uses Microsoft Word to write articles and papers. However, R Markdown has several advantages over MS Word:

- Easy to inserting images and tables in documents.
- Values of variables (e.g. values in tables or *p*-values) are automatically updated when raw data changes.
- Easy citation using citation keys (Zotero + Better BibTeX greatly facilitates this).
- Mutiple output format, e.g. LaTeX, PDF, Web Page, Book, etc.
- Template support for Journel articles, such as Elsevier, Sage, Springer, so no formatting is needed.

But I think all benefits about R Markdown mentioned above aren't enough to persuade people into giving up MS Word, since people are conservative in adoping new things.

[2] I actually stared and gave up learning programming languages for three times (C++, C, and then Python) before I successfully learned R.

If using R Markdown (or R) has benefits specific to the field related to the researcher, it greatly enhances the chance of adopting R Markdown. Hence, if I want to persuade people to use R Markdown, I can first build R packages that enhances the ability of R Markdown in that field.